Applied Soft Computing Journal 164 (2024) 111935

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Metaheuristics-guided active learning for optimizing reaction conditions of high-performance methane conversion

Gyoung S. Na [a,*], Hyun Woo Kim [b,*]

[a] Korea Research Institute of Chemical Technology (KRICT), Republic of Korea
[b] Gwangju Institute of Science and Technology (GIST), Republic of Korea

ARTICLE INFO

Dataset link: https://github.com/ngs00/alms

Keywords:
Artificial intelligence
Chemical engineering
Active learning
Non-oxidative coupling of methane

ABSTRACT

Converting greenhouse gases into value-added chemical compounds has been widely studied in chemical science and engineering for sustainable industry. In particular, nonoxidative coupling of methane (NOCM) that transforms methane into useful chemical compounds has received great interests because methane is a naturally abundant greenhouse gas. NOCM has received significant attention in sustainable industry for addressing the climate crisis. However, there are complex optimization problems in maximizing the efficiency of the nonoxidative methane conversion. Although conventional data-driven methods have been successfully applied to various engineering problems, a data-driven optimization of NOCM remains a challenging problem because expensive chemical experiments should be performed to collect prior data for model training. To avoid the expensive costs of chemical experiments, we propose an active learning method that performs training data augmentation and model re-training without pre-defined unlabeled experimental data. To this end, we combine active learning with metaheuristic algorithms to perform active learning with statistically augmented data. We applied the proposed method to a high-throughput screening task to discover new reaction conditions of high-performance NOCM, and the high-throughput screening error was significantly reduced by 69.11%.

1. Introduction

Clean and environment-friendly technologies are becoming increasingly important to achieve the carbon neutral society in the sustainable future [1]. In particular, chemical technologies to reduce methane in manufacturing systems have been widely studied because methane is a strong greenhouse gas released by many industries [2]. Recently, converting methane into value-added chemical compounds has received significant attention in various industrial fields due to the abundance of methane in natural gas and biomass [3,4], and some remarkable advances in chemical technologies were been made for an efficient and sustainable methane conversion [5–8].

Despite the benefits of the methane conversion methods, they should be carefully applied to industrial applications because transforming methane into other valuable chemical compounds may produce CO_2, which causes global warming [9]. Moreover, indirect methane conversion methods additionally consume water or oxygen to generate the syngas, and it has serious drawbacks in terms of carbon efficiency and high purity requirements for the feedstock [10]. For this reason, converting methane directly without making CO_2 [8,10,11] has received significant attention for sustainable and green manufacturing systems. In particular, nonoxidative coupling of methane (NOCM) or methane to olefins, aromatics, and hydrogen has been widely studied in chemistry-related fields because it converts methane directly into value-added compounds [8,12]. Recently, various methods of NOCM have shown remarkable methane conversion performances [13–15].

NOCM is usually performed by feeding methane and hydrogen molecules into a reactor in controllable thermodynamic conditions, as illustrated in Fig. 1. The reactor is typically in a cylindrical shape, and the catalytic materials can either be coated to the wall of the reactor or packed into the space of the reactor [13,16]. Thus, the conversion of methane into the desired chemical compounds depends on thermodynamic conditions such as temperature, pressure, ratio and flow rate of the reacting gases, as well as on the characteristics of the reactors, which are usually described by their length and diameter [15]. We refer to the engineering parameters of the reactor and thermodynamic conditions as *reaction conditions* throughout this work. Experimentally, these reaction conditions are controlled externally, and several elementary reactions take place in the reactor, the first few examples are listed in the figure. Since these elementary reactions are determined by the molecules involved in the NOCM process [17], uncontrollable factors should occur due to the intrinsic properties of the molecules. This makes the development of NOCM challenging, as it

* Corresponding authors.
E-mail addresses: ngs0@krict.re.kr (G.S. Na), hwk@gist.ac.kr (H.W. Kim).

https://doi.org/10.1016/j.asoc.2024.111935
Received 6 June 2023; Received in revised form 11 June 2024; Accepted 25 June 2024
Available online 3 July 2024

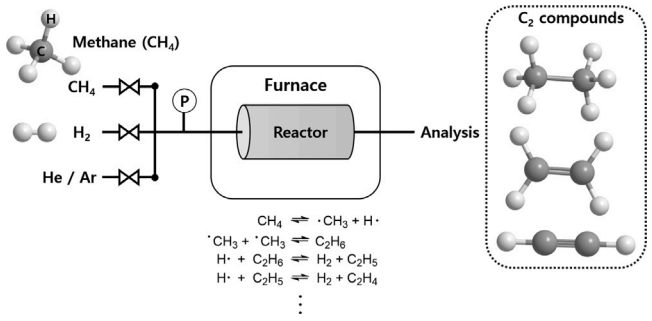

Fig. 1. Schematic illustration of the nonoxidative methane conversion. In a typical experimental setting, gases including methane (CH_4) and hydrogen (H_2) are fed into a cylindrical reactor of which pressure and temperatures are maintained externally. Important elementary reactions to produce C_2 compounds are also displayed. Final reaction outcomes are analyzed through gas chromatography. In this scheme, the NOCM reaction is mainly controlled by six reaction conditions such as pressure, temperature, flow rate of gases, H_2 content, reactor length, and reactor diameter.

requires harsh conditions such as high temperatures above 950 °C [14,17]. Although various manual and high-throughput approaches have been conducted to optimize the reaction conditions of NOCM [18], existing search methods still have practical limitations because the experimental evaluations based on real-world chemical experiments require extensive costs. Therefore, we need an efficient method to find optimal reaction conditions beyond the manual and high-throughput screening approach.

To this end, a number of computational approaches were developed in engineering disciplines related to optimization tasks [19–23]. One of the most prominent examples is the use of machine learning in various engineering applications to optimize engineering conditions in large search spaces [24–26]. In particular, machine learning showed great potentials in chemical engineering [21,22,27]. However, the prediction capabilities of the machine learning models fundamentally limited to the given training dataset because the model parameters are optimized to approximate the data distribution of the training dataset [28]. For this reason, a sufficient amount of training data is crucial for successful machine learning in real-world data-driven optimizations, which contain many unexplored and noisy data. However, constructing a large training dataset is not feasible in chemical engineering because chemical experiments to obtain the training data are usually time-consuming [29]. Therefore, an efficient method to selectively collect informative training data is crucial to build an accurate prediction model for the applications of chemical engineering.

Active learning is a machine learning strategy to overcome the lack of training data by selectively labeling new training data from a set of pre-defined unlabeled data [30]. In various scientific applications, active learning has been successfully applied to overcome insufficient and biased training datasets [31–33] and showed remarkable improvements in the prediction accuracy of the prediction models without labor-intensive experiments to collect training data. In the first step of active learning, a prediction model $f(\cdot;\theta)$ is trained on the initial training dataset $\mathcal{D}_{tr}^{(0)}$, where θ are model parameters of the prediction model. Then, an acquisition function $\psi(\theta, \mathcal{D}_{tr}^{(0)})$ selects a certain number of data from a pre-defined unlabeled dataset $\mathcal{D}_{ul}$. In active learning, the acquisition function can be implemented based on the model- and data-driven methods, such as uncertainty estimation [34], reinforcement learning [35], and meta-learning [36]. After the data selection on $\mathcal{D}_{ul}$, a human expert or labeling method makes target labels of the selected unlabeled data, and the prediction model is re-trained on a new training dataset $\mathcal{D}_{tr}^{(1)} = \mathcal{D}_{tr}^{(0)} \cup \psi(\theta, \mathcal{D}_{tr}^{(0)})$. This training-labeling cycle is repeated K times in active learning, where $K \in \mathbb{N}$ is a hyperparameter of active learning. The number of active learning cycles K is determined by budgets for data labeling [37]. After repeating the active learning cycle K times, we can get a final prediction model $f(\cdot;\theta^*)$ trained on $\mathcal{D}_{tr}^{(K)}$, which is expected to have better generalization and extrapolation capabilities than the initial prediction model trained on $\mathcal{D}_{tr}^{(0)}$.

Despite numerous successes of active learning in various engineering applications, the improvements by active learning are fundamentally limited to the volume and diversity of the pre-defined unlabeled datasets because the training data is selected within the given unlabeled dataset. For this reason, sufficient unlabeled data is crucial for successful active learning. However, preparing sufficient unlabeled data for active learning is not feasible in many applications of chemical engineering because systemic generation of training data through chemical experiments are time-consuming and labor-intensive [38]. To this end, Bayesian optimization [39] has been utilized for active learning in chemistry-related fields [40–43], and self-driving platforms based on the Bayesian optimization have attracted considerable interest [44,45].

We propose an efficient and data-generative active learning method without pre-defined unlabeled dataset to optimize the engineering conditions of chemical engineering. We call the proposed method the *active learning with metaheuristic search* (ALMS). ALMS aims to build an accurate prediction model without pre-define unlabeled dataset of active learning by adaptively generating new training data based on metaheuristic algorithms that can augment training data without model optimization. In this paper, we refer to the active learning without pre-defined unlabeled datasets as *generative active learning*. In the active learning phase, the metaheuristic algorithm of ALMS generates new unlabeled data by maximizing the predictive uncertainty [46] estimated by the current prediction model. After the data generation, a human expert or labeling method makes target labels of the generated unlabeled data. Then, ALMS optimizes the model parameters of the prediction model on a new training dataset $\mathcal{D}_{tr}^{(k)} = \mathcal{D}_{tr}^{(k-1)} \cup \mathcal{D}_{cd}^{(k)}$, where $\mathcal{D}_{cd}^{(k)}$ is a set of the generated data. The active learning cycle of the model training and data generation steps in ALMS is repeated K times, where $K \in \mathbb{N}$ is a pre-defined hyperparameter of active learning. Finally, we take the final prediction model trained on $\mathcal{D}_{tr}^{(K)}$.

In this paper, we applied ALMS to an engineering problem of optimizing reaction conditions for high-performance NOCM, which is one of the emerging problems in chemical engineering for sustainable and green manufacturing systems [47]. To evaluate the effectiveness of ALMS, we conducted a computational experiment based on a dataset of nonoxidative methane conversion containing 251 reaction conditions and their experimental methane conversion performances [15]. In the experiment, we trained a prediction model based on ALMS and predicted C_2 yields of the input reaction conditions of NOCM experiments. Note that the C_2 yield is a key property to determine the performance of NOCM to desired chemical products, such as ethylene and acetylene [8], which are used as a base compound to produce synthetic fibers and drugs.

In the task of predicting C_2 yields, we were able to significantly improve the prediction accuracy of the prediction models from 0.59 to 0.75 in the R^2-score [48] by applying ALMS in the training processes of the prediction models. Furthermore, we conducted a high-throughput screening using the prediction models trained with ALMS to discover new reaction conditions of high-performance NOCM, and the screening method based on ALMS successfully discovered promising reaction conditions. The contributions of this work can be summarized as follows.

- This paper proposes a new framework for active learning without pre-defined unlabeled datasets.
- ALMS does not require a pre-defined large dataset for active learning, and it significantly improves the practicality of ALMS in real-world chemical applications.
- We devised a new decision strategy and learning rate adaptation method for active learning on real-world chemical data.
- We applied ALMS to the engineering problem of optimizing the reaction conditions for high-performance NOCM, which is an emerging issue in sustainable and green manufacturing systems.

- We significantly improved the R^2-score of the prediction models from 0.59 to 0.75 in the problem of predicting the C_2 yields of the reaction conditions.
- The prediction models trained by ALMS successfully predicted the C_2 yields of the top-10 reaction conditions in a high-throughput screening to discover new high-performance reaction conditions.

2. Related work

2.1. Engineering problems on nonoxidative methane conversion

For sustainable and green manufacturing systems, various methods have been devised to develop high-performance NOCM engineering [13,14] and computational models for explaining the NOCM processes [17,49]. However, the conventional methods are not generally applicable to the studies on the nonoxidative methane conversion due to the following two main reasons as: (1) The applicability of conventional reaction optimization methods is limited to specific catalyst compositions constrained by the limited range of elements. (2) Conventional reaction optimization methods require a large amount of data to adjust manual parameters because the experimental results are sensitive to the reaction conditions [19].

Recently, machine learning has been widely studied in various engineering applications to efficiently explore large search spaces of chemical engineering [50–52]. Furthermore, metaheuristic-based machine learning was successfully applied to optimize the reaction conditions of high-performance NOCM [15]. However, collecting sufficient training data in chemical applications is a challenging problem in most chemical applications due to the expensive costs of chemical experiments to collect the training data, and existing machine learning methods in chemical engineering overlooked the fundamental limitations of machine learning in the generalization and extrapolation capabilities on insufficient training datasets [53,54]. Therefore, we need an efficient method for supplementing training data to build a prediction model with reliable generalization and extrapolation accuracy.

2.2. Metaheuristic algorithms for data augmentation

Data augmentation aims to generate new prior data based on statistical and data-driven approaches [55,56]. In machine learning, data augmentation has been widely studied to prevent the overfitting of the prediction models due to lack of training data [57]. Various data augmentation methods were proposed to generate additional training based on noise injection, kernel methods, and random deletion [56,58,59]. Existing data augmentation methods were successfully applied to image data [56,60], text data [58], and time-series data [59]. However, existing data augmentation methods have been hardly applied to vector-shaped data because the vector-shaped data does not inhere structured information. Although generative adversarial networks (GANs) have been proposed for generating synthetic data from unknown data distributions [61], the GAN-based data augmentation still has limitations in real-world applications because it requires sufficient training data to train the generative model.

Several metaheuristic algorithms were studied for data augmentation in scientific applications, such as statistical tests [62] and computer vision [63]. These metaheuristic-based data augmentation methods employed domain-specific objective functions to generate informative training and test datasets. Although several metaheuristic-based methods were successfully applied to several scientific applications by adapting the domain-specific objective functions [64], their practical applications to general problems are still limited due to the domain-specific problem definition and objective function. Furthermore, combining active learning and metaheuristic algorithms for the generative active learning has hardly been studied.

2.3. Active learning

Active learning has been widely studied to efficiently collect training data to improve the generalization capabilities of machine learning method [65,66]. Various active learning methods have been studied from the model- and data-driven approaches [37]. The model-driven approach contains heuristically-designed and statistical methods, such as divergence estimation [67] and uncertainty calculation [34]. On the other hand, the data-driven approach aims to estimate the usefulness of unlabeled data in the training process of the prediction models [68]. Several data-driven methods for active learning have been proposed based on various approaches, such as uncertainty learning [68], reinforcement learning [35], and meta learning [36]. Although active learning has been successfully applied to various applications in computer science, the effectiveness of active learning is fundamentally limited to pre-defined unlabeled dataset that defines the search space of the active learning methods. However, preparing a large and diverse unlabeled dataset for active learning is not feasible in chemical applications because extensive chemical experiments to collect sufficient number of data for active learning are labor-intensive and time-consuming [29]. To circumvent this issue, active learning in chemistry-related fields is employed to specific processes or reactions in order to reduce the effort required to obtain the necessary training data [40,41]. By focusing on specific reactions, the optimization of a chemical reaction was successfully conducted with seven optimization cycles [40]. To the best of our knowledge, most of active learning studies have been conducted using the Gaussian process [69], with the objective of optimizing either a single or multiple variables [40–43,70,71]. Since metaheuristic algorithms have advantages as mentioned above, we believe there is still room to develop a new method to explore data without pre-defined datasets in various fields including chemical engineering.

3. Active Learning with Metaheuristic Search (ALMS)

3.1. Problem Definition of Active Learning on NOCM

The purpose of active learning on NOCM is to build an accurate prediction model $f(\cdot;\boldsymbol{\theta})$ from a given initial training dataset $\mathcal{D}_{tr}^{(0)} = \{(\mathbf{x}_1^{(0)}, y_1^{(0)}), (\mathbf{x}_2^{(0)}, y_2^{(0)}), \dots, (\mathbf{x}_{N_0}^{(0)}, y_{N_0}^{(0)})\}$ by supplementing new training data into $\mathcal{D}_{tr}^{(0)}$ over the active learning cycle, where $\mathbf{x}_n^{(k)}$ and $y_n^{(k)}$ are the reaction conditions and the real-valued target C_2 yield of the nth observation in $\mathcal{D}_{tr}^{(k)}$, respectively. In our problem setting, as shown in Table 1, the reaction condition $\mathbf{x}_n^{(k)}$ is defined as a 6-dimensional vector containing six control variables of NOCM. The values of these six control variables were collected during the NOCM experiments, and the correlations between the control variables and the target C_2 yields are presented in Supplementary Material.

For a given current training dataset $\mathcal{D}^{(k)}$ and model parameters $\boldsymbol{\theta}^{(k)}$, active learning methods construct the next training dataset $\mathcal{D}^{(k+1)}$ and re-train the current prediction model $f(\cdot;\boldsymbol{\theta}^{(k)})$ on $\mathcal{D}_{tr}^{(k+1)}$. We expect to improve the prediction accuracy of $f(\cdot;\boldsymbol{\theta}^{(k)})$ on the test dataset $\mathcal{D}_{ts}$ by re-training $f(\cdot;\boldsymbol{\theta}^{(k)})$ on $\mathcal{D}_{tr}^{(k+1)}$. This expectation is formally given by:

$$\mathcal{D}_{tr}^{(k+1)} = \underset{\mathcal{D}_{tr}^{(k+1)} \in \mathcal{X}\times\mathcal{Y}}{\arg\min} \sum_{(\mathbf{x},y)\in\mathcal{D}_{ts}} L(y, f(\mathbf{x};\boldsymbol{\theta}^{(k+1)})), \tag{1}$$

where $\mathcal{X}$ is the domain of the input reaction conditions, $\mathcal{Y}$ is the codomain of the target C_2 yields, and $\boldsymbol{\theta}^{(k+1)}$ is the model parameters optimized on $\mathcal{D}_{tr}^{(k+1)}$. The active learning cycle is repeated $K \in \mathbb{N}$ times, and $f(\cdot;\boldsymbol{\theta}^{(K)})$ becomes the final prediction model.

3.2. Overall process of ALMS

To avoid the expensive costs of dataset construction in chemical engineering, ALMS aims to build an accurate prediction model based on the generative active learning that repeats the training-labeling cycle of active learning without pre-defined extensive unlabeled datasets. To

Table 1
Description of the six reaction conditions in NOCM.

Name	Unit	Range	Mean (±Std.)	Median	Description
Pressure	barg	[0.005, 0.888]	0.228 (±0.186)	0.158	Pressure in reactor
Temperature	°C	[890, 1,200]	1130.347 (±71.259)	1150	Temperature in reactor
Flow rate	sccm	[40, 320]	146.848 (±79.325)	120	Feeding rate of gas
H_2 content	–	[0, 1]	0.391 (±0.312)	0.2	Relative amount of H_2
Reactor length	cm	[15, 45]	20.199 (±10.830)	15	Length of reactor
Reactor diameter	mm	[4, 16]	10.711 (±5.064)	6.8	Diameter of reactor

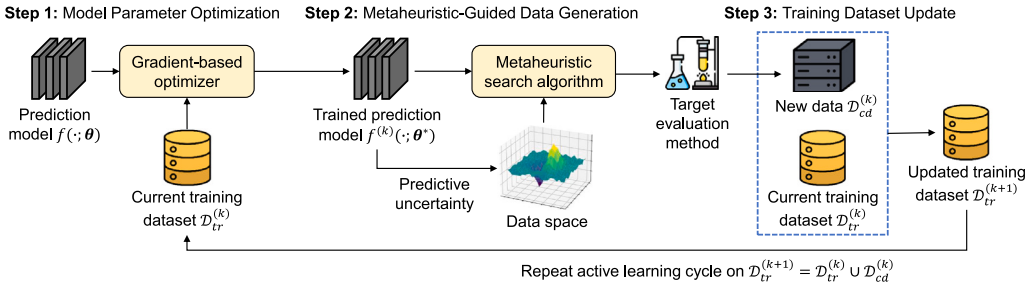

Fig. 2. Active learning cycle of ALMS to train a prediction model $f(\cdot;\theta)$. The ALMS-based active cycle consists of three steps: (1) model parameter optimization, (2) metaheuristic-guided data generation, and (3) training dataset update. This active learning cycle is repeated K times, where $K \in \mathbb{N}$ is a pre-defined hyperparameter of ALMS.

this end, the active learning cycle of ALMS consists of three steps: (1) model parameter optimization, (2) metaheuristic-guided data generation, and (3) training dataset update. Fig. 2 illustrates the active learning cycle of ALMS to build an accurate prediction model beyond the initial training dataset $\mathcal{D}_{tr}^{(0)}$. In each step, ALMS performs the following metaheuristic-based active learning to build a prediction model $f(\cdot;\theta)$ as:

- **(Step 1) Model parameter optimization:** ALMS optimizes the model parameters of the prediction model $f(\cdot;\theta)$ for a given current training dataset $\mathcal{D}_{tr}^{(k)}$.
- **(Step 2) Metaheuristic-guided data generation:** ALMS generates new dataset $\mathcal{D}_{cd}^{(k)}$ to be labeled, by maximizing the predictive uncertainty of $f(\cdot;\theta)$ based on the metaheuristic algorithm.
- **(Step 3) Training dataset update:** The target values of the generated dataset $\mathcal{D}_{cd}^{(k)}$ are experimentally measured (or calculated), and the new training data is merged into the next training dataset as $\mathcal{D}_{tr}^{(k+1)} = \mathcal{D}_{tr}^{(k)} \cup \mathcal{D}_{cd}^{(k)}$.

We define the above steps 1–3 as an active learning cycle of ALMS, and the active learning cycle is repeated K times in ALMS, where K is a pre-defined hyperparameter determined by the budgets for data labeling. Finally, we used the prediction model trained on $\mathcal{D}_{tr}^{(K)}$. In the following sections, we will explain formal definitions and implementation details of each step of ALMS.

3.3. Model parameter optimization

As a prediction model of active learning, ALMS employs Bayesian neural networks (BNNs) [46] to estimate the predictive uncertainty as well as predict the target values. A predictive distribution of BNNs is defined by:

$$p(y|\mathbf{x}, D) = \int p(y|\mathbf{x}, \theta)p(\theta|D)d\theta, \tag{2}$$

where θ are model parameters of BNNs, and D is a training dataset for optimizing the model parameters of BNNs. The posterior $P(\theta|D)$ of the Bayesian inference can be rewritten by:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{\int P(D|\theta)P(D)d\theta}. \tag{3}$$

The training algorithm for BNNs aims to optimize the model parameters θ by minimizing the Kullback–Leibler divergence (KL-divergence) based on the variational inference as:

$$\theta^* = \arg\min_{\theta} D_{KL}(q(\theta) \parallel p(\theta|D)). \tag{4}$$

where D_{KL} means the KL-divergence, and $q(\theta)$ is a variational distribution to approximate the prior of θ. The purpose of the model parameter optimization step in ALMS is to fit the model parameters θ of BNNs for a given current training dataset $\mathcal{D}_{tr}^{(k)}$ as:

$$\theta^* = \arg\min_{\theta} D_{KL}(q(\theta) \parallel p(\theta|\mathcal{D}_{tr}^{(k)})). \tag{5}$$

In the active learning cycle, ALMS uses the trained BNN $f(\cdot;\theta^*)$ as a prediction model to predict the target values of the input reaction conditions and estimate the uncertainty of the prediction results for the next metaheuristic-guided data generation step.

3.4. Metaheuristics-guided data generation

Since we should conduct time-consuming and labor-intensive chemical experiments to collect chemical data, preparing sufficient unlabeled data for active learning is not practical in most applications of chemical engineering. To overcome this limitation of active learning in chemical applications, ALMS adopts the metaheuristic algorithm to generate new training data for the given initial training dataset and current model parameters. There are many metaheuristic algorithms for implementing ALMS, such as genetic algorithm [72], particle swarm optimization [73], and equilibrium optimization [74]. To select an appropriate metaheurisitic algorithm for ALMS, we conducted an experimental evaluation in Section 4.6 to compare the active learning performances of ALMS for different metaheuristic algorithms. Based on the evaluation results, we implemented ALMS using the artificial bee colony (ABC) algorithm [75] that showed the best active learning performance in the NOCM problem. The ABC algorithm is an efficient randomized iterative search based on the intelligence foraging behavior of honey bee colonies. In addition to the NOCM problem, the ABC algorithm showed state-of-the-art optimization performances in various scientific applications [64,76]. By employing the ABC algorithm, ALMS can generate new candidate training data without the predefined unlabeled datasets, and it makes the active learning methods to be applicable to the optimization problems in chemical engineering applications.

In the initial stage, the ABC algorithm in ALMS generates a set of initial engineering conditions $\mathcal{X}^{(0)} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ based on the

randomized search of the input space as:

$$\mathbf{x}_{n,i} = \mathbf{x}_{min,i} + \texttt{rand}(0,1) \times (ub_i - lb_i), \tag{6}$$

where $N \in \mathbb{N}$ is a pre-defined hyperparameter to determine the number of candidate engineering conditions, $\mathbf{x}_{n,i}$ is the ith feature of the nth engineering condition, and $\texttt{rand}(0,1)$ is a random number within $[0, 1]$. Note that lb_i and ub_i are the lower and upper bonds of the ith feature, respectively. After generating the initial set $\mathcal{X}^{(0)}$, the ABC algorithm calculates new candidate engineering conditions by exploring the input space based on the neighborhood search around the initial engineering conditions as:

$$\mathbf{v}_{n,i} = \mathbf{x}_{n,i} + \texttt{rand}(-1,1) \times (\mathbf{x}_{n,i} - \mathbf{x}_{m,i}), \tag{7}$$

where $\mathbf{x}_m$ is a randomly selected engineering condition from $\mathcal{X}^{(0)} \setminus \{\mathbf{x}_n\}$, and $\texttt{rand}(-1,1)$ is a random number within $[-1, 1]$. Once the new candidate engineering condition $\mathbf{v}_n$ is calculated from $\mathcal{X}^{(0)}$, the ABC algorithm selects either $\mathbf{x}_n$ or $\mathbf{v}_n$ based on a greedy selection. The ABC algorithm of ALMS performs the greedy selection based on the predictive uncertainty of the engineering conditions. In other words, if the predictive uncertainty of $\mathbf{v}_n$ from the pre-trained BNN $f(\cdot;\theta^*)$ is greater than that of $\mathbf{x}_n$, then replace $\mathbf{x}_n$ with $\mathbf{v}_n$, and otherwise keep $\mathbf{x}_n$. As a result, an intermediate set $S = \{\zeta(\mathbf{x}_1, \mathbf{v}_1), \dots, \zeta(\mathbf{x}_N, \mathbf{v}_N)\}$ is generated by the greedy selection, where $\zeta(\cdot,\cdot)$ is the greedy selection method based on the predictive uncertainty.

For an intermediate set S, a set of new engineering conditions $\mathcal{X}^{(1)}$ of the next generation is constructed by a probabilistic data selection strategy based on the roulette wheel selection mechanism [77]. The data selection probability of the nth data in the roulette wheel method is defined by the predictive uncertainty on S as:

$$p_n = \frac{\sigma(\mathbf{x}_n;\theta^*)}{\sum_{\mathbf{x}\in S}\sigma(\mathbf{x};\theta^*)}, \tag{8}$$

where $\sigma(\mathbf{x}_n;\theta^*)$ is the predictive uncertainty of $\mathbf{x}_n$, which is calculated by entering $\mathbf{x}_n$ into the pre-trained BNN $f(\cdot;\theta^*)$. The data selection probability in Eq. (8) means that the engineering condition with the high predictive uncertainty have a high probability of the data selection on S. For a given hyperparameter $T \in \mathbb{N}$, the ABC algorithm in ALMS repeats the data generation and selection processes T times to generate the final set of the new engineering conditions $\mathcal{X}^{(T)}$, which will be merged to the current training dataset $\mathcal{D}_{tr}^{(k)}$ in the active learning cycle of ALMS.

3.5. Target data evaluation and training dataset update

After the metaheuristic-guided data generation, $\mathcal{D}_{cd}^{(k)} = \{(\mathbf{x}_1, \phi(\mathbf{x}_1)), (\mathbf{x}_2, \phi(\mathbf{x}_2)), \dots, (\mathbf{x}_N, \phi(\mathbf{x}_N))\}$ is generated by measuring or calculating the target values of the generated engineering conditions in $\mathcal{X}^{(T)}$, where $\phi(\mathbf{x})$ is a calculation or experimental method to obtain the target values. We refer to $\phi(\mathbf{x})$ as *target evaluation method*. In this work, $\phi(\mathbf{x})$ is defined as a nonoxidative methane coupling experiment to measure the C_2 yield for a given reaction condition $\mathbf{x}$.

In the active learning cycle of ALMS, $\phi(\mathbf{x})$ can be implemented by domain knowledge of human experts, computational simulations, or chemical experiments [30,37]. For a generated dataset $\mathcal{D}_{cd}^{(k)}$, the training dataset of the next active learning cycle is constructed by merging the previous training dataset $\mathcal{D}_{tr}^{(k)}$ with the generated training dataset $\mathcal{D}_{cd}^{(k)}$ as $\mathcal{D}_{tr}^{(k+1)} = \mathcal{D}_{tr}^{(k)} \cup \mathcal{D}_{cd}^{(k)}$. After updating the training dataset, ALMS re-trains the prediction model $f(\cdot;\theta)$ on the new training dataset $\mathcal{D}_{tr}^{(k+1)}$, as described in Fig. 2. In ALMS, the three active learning steps in Sections 3.3–3.5 are repeated K times, where $K \in \mathbb{N}$ is a pre-defined hyperparameter of ALMS. Finally, we use the prediction model $f(\cdot;\theta)$ trained on the final training dataset $\mathcal{D}_{tr}^{(K)}$.

Algorithm 1: The overall process of ALMS

Input : $\mathcal{D}_{tr}^{(0)}$: Initial training dataset;
$\phi(\mathbf{x})$: Target evaluation method;
K: Number of active learning cycles;
N: Number of generated data;
M: Number of selected data;
T: Iteration of metaheuristic search;
γ: Acceptance threshold

Output: $f(\cdot;\theta^*)$: Optimized prediction model

```
// Train a prediction model on the initial training dataset.
θ* = argmin_θ D_KL(q(θ)||P(θ|D_tr^(0)))
for k ∈ {0, 1, ..., K − 1} do
    // Metaheuristic search for data generation.
    for t ∈ {1, 2, ..., T} do
        // Generate new engineering conditions.
        {x_1, x_2, ..., x_N} = RandomizedSearch(X^(t−1), N)
        // Calculate data selection probability by Eq. (8).
        for n ∈ {1, 2, ..., N} do
            p_n = σ(x_n;θ) / Σ_{m=1}^{N} σ(x_m;θ)
        end
        X^(t) = RouletteWheel(p_1, p_2, ..., p_N, M)
    end
    // Measure target values of new data.
    for x ∈ X^(T) do
        y = φ(x)
    end
    // Construct a set of new training data.
    D_cd^(k) = {(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)}
    // Update the training dataset by Eq. (9).
    D_tr^(k+1) = ζ(D_tr^(k), D_tr^(k+1), D_cd^(k), γ)
    // Learning rate decay.
    η^(k) = max{(W_p^(k) / W_p^(k+1)) η^(k−1), 1e-5}
    // Re-train f(·;θ) on D_tr^(k+1).
    θ* = argmin_θ D_KL(q(θ)||P(θ|D_tr^(k+1)))
end
return f(·;θ*)
```

3.6. Acceptance threshold of active learning

Since ALMS generates new training data based on the randomized iterative search, some generated data may overlap with the data of the current training dataset. That is, the data imbalance problem [78] can occur during the active learning by ALMS, and it causes the overfitting problem of the prediction models. To prevent the data imbalance problem, we devised a decision rule of ALMS for accepting or rejecting the training data update based on the statistical similarity between the current and updated training datasets. To this end, we employ the Wasserstein distance that nonparametrically measures the statistical divergence between two data distributions [79]. Based on the Wasserstein distance, the decision rule for the training dataset update $\zeta(\mathcal{D}_{tr}^{(k)}, \mathcal{D}_{tr}^{(k+1)}, \mathcal{D}_{cd}^{(k)}, \gamma)$ is formally defined as:

$$\zeta = \begin{cases} \mathcal{D}_{tr}^{(k)} \cup \mathcal{D}_{cd}^{(k)}, & \text{if } W_p^{(k)} > \gamma \\ \mathcal{D}_{tr}^{(k)}, & \text{otherwise,} \end{cases} \tag{9}$$

where $\gamma \geq 0$ is a hyperparameter of ALMS, and $W_p^{(k)}$ is the p-Wasserstein distance between $\mathcal{D}_{tr}^{(k)}$ and $\mathcal{D}_{tr}^{(k+1)}$. The decision rule in Eq. (9) means that we merge the generated training datasets with the current training dataset only when the difference between the current and updated training datasets should be greater than the hyperparameter γ.

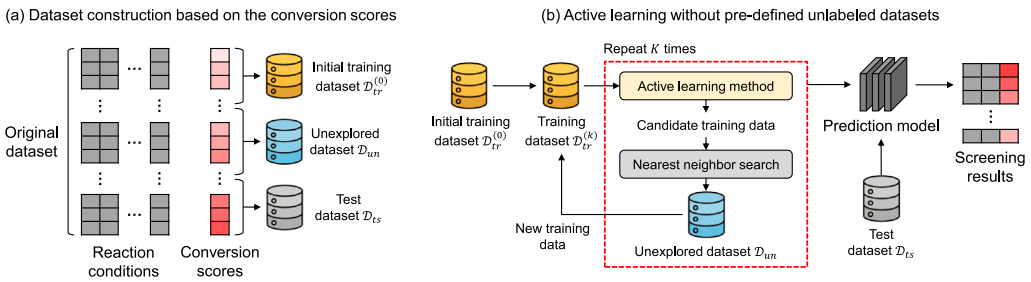

Fig. 3. Experiment settings of dataset construction and high-throughput screening to evaluate the effectiveness of active learning methods in discovering high-performance reaction conditions. (a) Dataset construction of the initial, unexplored, and test datasets based on conversion score. (b) Experiment process of the active learning without pre-defined unlabeled datasets.

3.7. Learning rate decay over the active learning cycle

Although the convergence of BNNs and the ABC algorithm have been investigated independently [80,81], the convergence of BNNs in the ABC algorithm is not guaranteed. ALMS employs a data-driven learning rate decay method to enforce the convergence of BNNs. To this end, we devised an algorithmic rule for determining the initial learning rate of BNNs based on Wasserstein distance [79] as:

$$\eta^{(k)} = \texttt{max}\left\{\left(\frac{W_p^{(k)}}{W_p^{(k-1)}}\right)\eta^{(k-1)}, \texttt{1e–5}\right\}, \tag{10}$$

where $\eta^{(k)}$ is the initial learning rate at kth active learning cycle of ALMS, $W_p^{(k)}$ is the Wasserstein distance between $\mathcal{D}_{tr}^{(k+1)}$ and $\mathcal{D}_{tr}^{(k)}$, and max is a binary operation that selects the maximum value among two given values. Note that $W_p^{(0)}$ is initialized to 5e–4.

The update rate $W_p^{(k)}/W_p^{(k-1)}$ determines the increase or decrease of the current initial learning rate by comparing the Wasserstein distances between the updated and current training datasets at k and $k-1$. If most informative data is already supplemented during active learning, $W_p^{(k)}$ will converge to zero because new training data generated by the ABC algorithm is in the current training dataset. As a result, the initial learning rate of BNNs will converge to small values, and it leads to the convergence of the model parameters of BNNs during the generative active learning. Algorithm 1 describes the overall process of ALMS to construct a prediction model from given initial training dataset $\mathcal{D}_{tr}^{(0)}$ and target evaluation method $\phi(\mathbf{x})$.

4. Experiments

We applied ALMS to an engineering problem to discover the reaction conditions of high-performance methane conversion, which is an emerging issue in various industrial applications for sustainable and green manufacturing systems [1,47]. In the experiments, we used ALMS to build an accurate prediction model that predicts C_2 yields of the reaction conditions beyond a given initial training dataset. We followed a previous study of artificial intelligence for the nonoxidative methane conversion [15] to set the experiment settings of active learning. We used BNN with four fully-connected hidden layers with 128 neurons and the model parameters of BNN using Adam optimizer [82]. The initial learning rate and L_2 regularization coefficient of the Adam optimizer were fixed to 5e–4 and 5e–6 during the active learning, respectively. In the active learning cycle, the ABC algorithm in ALMS was executed with 1000 population size ($N = 1,000$) and 500 search iteration to find new training reaction condition where the predictive uncertainty of the pre-trained BNN is maximized.

We implemented ALMS and experiment scripts based on Python 3.9. PyTorch[1] and MEALPY[2] libraries were used to implement the BNN-based prediction models and the ABC-based metaheuristic search, respectively. For the learning rate decay of ALMS in Section 3.7, we used the sliced Wasserstein distance [83] to calculate the Wasserstein distance of Eq. (10) in multi-dimensional data. All experiment scripts and source codes of ALMS are available in a public repository.[3] The experiments were conducted on a single machine of Intel i9-12900K CPU, 64G memory, and GeForce RTX 3090. In this machine, ALMS required about 0.08 s for each iteration of data generation and total 29.31 s for the entire active learning.

4.1. Experiment settings of active learning

Before the high-throughput screening of the reaction conditions, we evaluated the active learning performance of ALMS in the problem of nonoxidative methane conversion, where it is difficult to construct an extensive unlabeled dataset for active learning. For the experimental evaluation, we designed a computational experiment because experimentally measuring the C_2 yield of the given reaction conditions is time-consuming and labor-intensive. Note that the purpose of the experimental evaluation in this paper is to demonstrate the effectiveness of ALMS in the active learning task of chemical engineering (especially the methane conversion). In addition, although we focus on a specific manufacturing process of nonoxidative methane conversion, it should be noted that ALMS is applicable to general reaction engineering problems.

Fig. 3 illustrates the experiment setting of our computational experiment for the experimental evaluation of ALMS. In the experiment, we split the original dataset into initial, unexplored, and test datasets to make an experiment setting of real-world chemical engineering, as shown in Fig. 3-(a). We define a conversion score to split dataset as:

$$s_n = y_n + (10 - \texttt{min}(10, c_n)), \tag{11}$$

where y_n and c_n are experimentally measured C_2 yield (%) and coke selectivity (%) of the nth reaction condition, respectively. Since our goal is to discover new reaction conditions of high C_2 yield with a constraint for low coke selectivity, we defined the conversion score so that it is proportional to the C_2 yield under the constraint inversely proportional to the coke selectivity.

After splitting the dataset, we perform the active learning based on the generated data, as shown in Fig. 3-(b). In the active learning cycle, new training data is selected by a nearest neighbor search for the generated data on the unexplored dataset instead of collecting the training data by experimentally measuring the C_2 yields of the generated reaction conditions. We selected the most similar three neighbor data for each intermediate training data in the nearest neighbor search. All experimental evaluations were repeated by 10 times, and we report the means and standard deviations of R^2-scores and MAEs if their calculations are available. Based on our computational experiment,

[1] https://pytorch.org.
[2] https://mealpy.readthedocs.io/en/latest/.
[3] https://github.com/ngs00/alms.

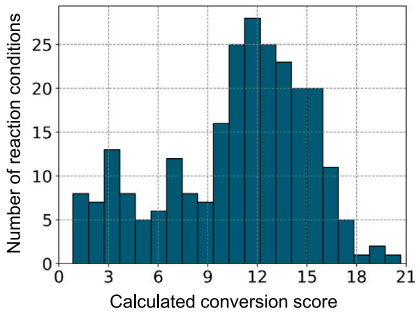

Fig. 4. Histogram of the calculated conversion scores of the 251 reaction conditions. The conversion score of the reaction condition is calculated by Eq. (11). X and Y axes mean the calculated conversion score and the number of reaction conditions, respectively.

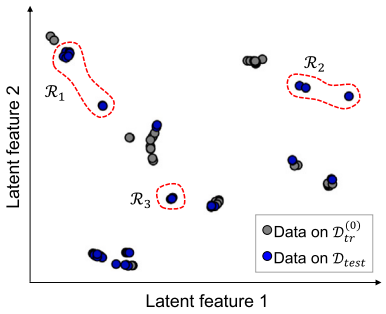

Fig. 5. Data distribution of the reaction conditions on $\mathcal{D}_{tr}^{(0)}$ and $\mathcal{D}_{ts}$. Gray and blue points indicate the reaction conditions embedded by t-SNE [85]. In the data distribution, $\mathcal{R}_1$, $\mathcal{R}_2$, and $\mathcal{R}_3$ denote unexplored data regions on the initial training dataset.

we can evaluate the effectiveness of ALMS in the active learning tasks of chemical engineering without the expensive costs of chemical experiments for the experimental valuation of the generated data [84].

4.2. Experimental NOCM dataset

Fig. 4 shows the histogram of the calculated conversion scores of the 251 reaction conditions. As shown in the histogram, most of the data is distributed in the conversion scores between 10 and 15. Based on the data distribution, we split the original dataset containing the 251 reaction conditions into the initial training, unexplored, and test datasets as follows.

- **Initial training dataset $D_{tr}^{(0)}$:** This dataset contains 77 reaction conditions whose conversion scores are less than 10. It will be used to train the prediction models in the initial active learning cycle of ALMS.
- **Unexplored dataset D_{un}:** It is a set of candidate data that will match to new training data discovered by the active learning methods. We constructed D_{un} by collecting 211 reaction conditions whose conversion scores are in [0, 15].
- **Test dataset D_{ts}:** It contains 40 reaction conditions whose conversion scores are greater than 15. This dataset will be used to evaluate the prediction accuracy of the prediction models trained by the active learning methods.

In the experiment, we trained the prediction models on $D_{tr}^{(0)}$ and D_{un} containing the relatively low-performance reaction conditions. Then, we evaluated the prediction capabilities of the prediction models on D^{ts} containing high-performance reaction conditions. This experiment verifies the practical potentials of ALMS in a real-world chemical application of discovering high-performance reaction conditions, even when it is difficult to accumulate sufficient amount of high-performance reaction conditions and outcomes in chemical engineering.

4.3. Prediction performances of machine learning prediction models

Before the high-throughput screening to discover high-performance reaction conditions, we experimentally validate that the prediction accuracy of the prediction models can be improved by ALMS in a task of predicting the C_2 yields of the reaction conditions. For the evaluation, we constructed the initial training, unexplored, and test datasets in Section 4.1. Fig. 5 shows the data distribution of the reaction conditions on the initial training and test datasets. The reaction conditions are projected into the 2-dimensional embedding space by t-SNE [85]. As shown in the data regions $\mathcal{R}_1$, $\mathcal{R}_2$, and $\mathcal{R}_3$ of Fig. 5, many reaction conditions of the test dataset are distributed on the data regions where are not explored on the initial training dataset. Therefore, the active learning methods should generate informative reaction conditions on unexplored data space for successful machine learning to predict the C_2 yields of the high-performance reaction conditions.

In this experiment, we measured R^2-scores and MAEs of the prediction models on $\mathcal{D}_{tr}$ for each active learning cycle of ALMS. The experiment was repeated by 10 times, and we report the means and standard deviations of R^2-scores and MAEs. Fig. 6 shows the measured R^2-scores and MAEs of the prediction models in predicting the C_2 yields of the reaction conditions for each active learning cycle of ALMS. Black line indicates the prediction performance of the prediction models trained by a baseline density-based active learning that generates new reaction conditions based on data density of the current training dataset in the training dataset update. For the generated data, the density-based active learning also selects new training data based on the nearest neighbor search on $\mathcal{D}_{un}$. We refer to the prediction models trained by ALMS and the baseline method as f_{alms} and f_{base}, respectively.

As shown in Fig. 6, the prediction model trained by ALMS outperformed the prediction model trained by the baseline active learning method for all active learning cycles, and this result shows that ALMS generated more informative training data during the generative active learning to predict the C_2 yields of the reaction conditions. In particular, f_{alms} outperformed f_{base} just after the first active learning cycle. Quantitatively, f_{alms} achieved the R^2-score and MAE of 0.75 and 1.35 after 10 active learning cycles, respectively. As a result, we were able to improve the R^2-score of the prediction model by 27.12% based on the active learning by ALMS. These experimental results show that ALMS is useful to improve the prediction capabilities of the prediction models in the task of predicting the C_2 yields of the reaction conditions, which is an essential property to determine the performance of nonoxidative methane conversion. In the following experiment, we will conduct a high-throughput screening based on the prediction model of ALMS to discover new reaction conditions of high-performance nonoxidative methane conversion.

4.4. High-throughput screening on reaction conditions

We performed a high-throughput screening based on the prediction models trained by ALMS in a task of discovering new reaction conditions for high-performance nonoxidative methane conversion to validate the practical potential of ALMS in real-world applications of chemical engineering. For the high-throughput screening, we used the prediction model trained by ALMS in Section 4.3, which was constructed to predict the C_2 yields under the given the input reaction conditions. Note that C_2 yield is an essential property that determines the performance of the methane conversion. To evaluate the high-throughput screening performance of ALMS, we separated ten reaction conditions of top-10 performance scores from the initial training and unexplored datasets, i.e., we did not provide any information about the top-10 reaction conditions to ALMS in the active learning process. In

Table 2
Top-10 reaction conditions in terms of the calculated conversion score. Each symbol indicates the input engineering condition as: p: pressure (barg), T: temperature (°C), f_r: flow rate (sccm), c_H: H_2 content, l_r: reactor length (cm), d_r: reactor diameter (mm). Score and y_C denote the calculated conversion score and the experimentally measured C_2 yield of each reaction condition. $f^{(0)}$ and $f^{(K)}$ are the predicted C_2 yields of the prediction models at the initial ($k = 0$) and final ($k = K$) active learning cycles, respectively. The initial prediction model $f^{(0)}$ is the same as the prediction model trained without metaheuristic-guided data generation of ALMS. Note that we did not display the reactor length l_r of the top-10 reaction conditions because l_r was 15 in all reaction conditions.

#	p (barg)	T (°C)	f_r (sccm)	c_H	d_r (mm)	score	y_C (%)	$f^{(0)}$ (%)	$f^{(K)}$ (%)
1	0.288	1195	160	0.6	6.8	20.66	12.84	11.34	12.60
2	0.286	1175	160	0.6	6.8	18.95	9.04	8.55	8.91
3	0.157	1150	80	0.6	6.8	18.37	8.61	8.48	8.54
4	0.377	1190	320	0.2	6.8	17.81	11.95	9.85	12.76
5	0.376	1185	320	0.2	6.8	17.75	10.89	9.23	11.62
6	0.193	1188	240	1	16	17.69	8.64	9.33	9.48
7	0.288	1185	160	0.6	6.8	17.64	10.82	10.16	10.70
8	0.385	1200	320	0.2	6.8	17.26	14.06	11.21	14.98
9	0.156	1160	80	0.6	6.8	16.66	10.26	9.77	9.93
10	0.119	1135	80	0.2	6.8	16.55	10.36	7.17	10.30
MAE on the top-10 reaction conditions								1.38	0.43

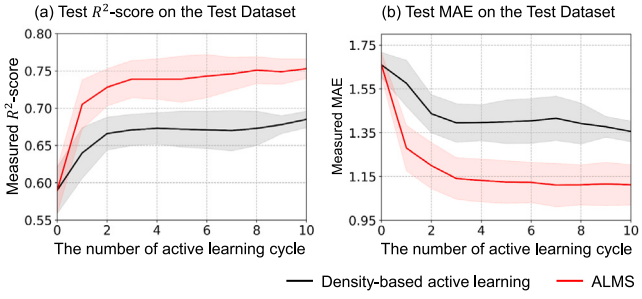

Fig. 6. Test R^2-scores and MAEs of the prediction models for each active learning cycle. Black and red lines indicate the test R^2-score or MAE of the prediction models trained by the baseline density-based active learning method and ALMS, respectively.

this experiment, we compared the prediction accuracy of the prediction models before and after the active learning by ALMS to evaluate the effectiveness of ALMS in the high-throughput screening.

Table 2 shows the top-10 reaction conditions and the prediction results of the prediction models in predicting the C_2 yields of the top-10 reaction conditions. In the table, $f^{(0)}$ and $f^{(K)}$ indicate the predicted C_2 yields of the prediction models before and after the active learning by ALMS, respectively. As shown in the prediction results of $f^{(K)}$, the prediction accuracy is significantly improved by the prediction model at $k = K$, and MAE on the top-10 reaction conditions were reduced from 1.38 at $k = 0$ to 0.43 at $k = K$. Quantitatively, we were able to reduce MAE on the top-10 reaction conditions by 69.11% by applying ALMS to the training of the prediction models. These experimental results show the practical potentials of ALMS in the high-throughput screening to discover the reaction conditions for high-performance methane conversion.

4.5. Qualitative analysis of data generation results

To investigate the reason for the performance improvement by ALMS, we visualized the data distributions of the ALMS-generated reaction conditions on the initial training and test datasets. For the visualization, we used t-SNE [85] to project the generated reaction conditions into the two-dimensional embedding space. Fig. 7-(a) and (b) show the data distributions of the reaction conditions generated by the baseline active learning method and ALMS, respectively.

As shown in Fig. 7-(a), the baseline active learning method generated new reaction conditions similar to those of the initial training dataset $D_{tr}^{(0)}$, and the generated data did not cover all data regions of the test dataset D_{ts}. By contrast, ALMS generated the reaction conditions distributed in the unexplored data regions of $D_{tr}^{(0)}$, as shown in $\mathcal{R}_1$ and $\mathcal{R}_2$ of Fig. 7-(b). In addition, the reaction conditions generated by ALMS covered the test reaction conditions on D_{ts} distributed in the unexplored data regions of $D_{tr}^{(0)}$. The visualization results clarify the reason for the improvement by ALMS in predicting C_2 yields of the test reaction conditions. Since the reaction conditions generated by the baseline method could not cover all data regions of D_{ts} as shown in Fig. 7-(a), the prediction model by the baseline active learning method could not achieve a R^2-score greater than 0.7. On the other hand, the prediction model trained by ALMS achieved the R^2-score of 0.75 because ALMS supplemented many training data similar to the test data during active learning. These qualitative analysis results show the effectiveness of the active learning based on a metaheuristic method called the ABC algorithm to maximize the predictive uncertainty in the chemical engineering problem of the nonoxidative direct conversion of methane.

4.6. Comparison of data generation performances for different metaheuristic algorithms

There are various choices for implementing the metaheuristics-guided data generation method of ALMS in Section 3.4. In this experiment, we evaluated the active learning performances of ALMS for different implementations of the metaheuristic algorithms. For each active learning cycle, we measured the R^2-scores and MAEs of the prediction models trained with ALMS of various metaheuristic algorithms. We employed eight metaheuristic algorithms as: genetic algorithm (GA) [72], particle swarm optimization (PSO) [73], differential evolution algorithm (DEA) [86], equilibrium optimizer (EO) [74], energy valley optimization (EVO) [87], queuing search algorithm (QSA) [88], war strategy optimization (WSO) [89], and artificial bee colony algorithm (ABCA) [75].

Fig. 8 shows the R^2-scores measured for each active learning cycle. The R^2-scores of all prediction models were improved by ALMS over the number of active learning cycles regardless of the metaheuristic algorithms. This result shows that ALMS is robust to the implementation of the metaheuristic-guided data generation. Furthermore, ALMS with GA showed consistent active learning performances, even though GA is one of the simplest metaheuristic algorithms. However, ALMS with ABCA showed remarkable performances, and the prediction model trained by ALMS with ABCA achieved the R^2-score of 0.75 after the 9th active learning cycle.

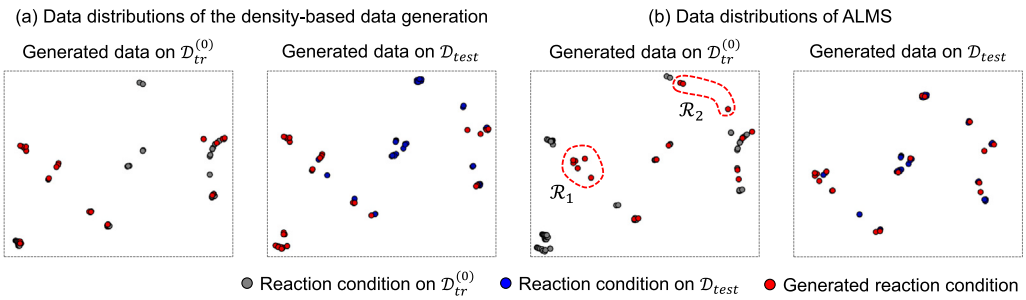

Fig. 7. Data distributions of the training data before and after the active learning of ALMS. Black, red, and blue data points indicate the initial training, test, and supplemented data in active learning. $\mathcal{R}_1$ and $\mathcal{R}_2$ are the ALMS-generated training data that covers unexplored data spaces in the initial training dataset $\mathcal{D}_{tr}^{(0)}$.

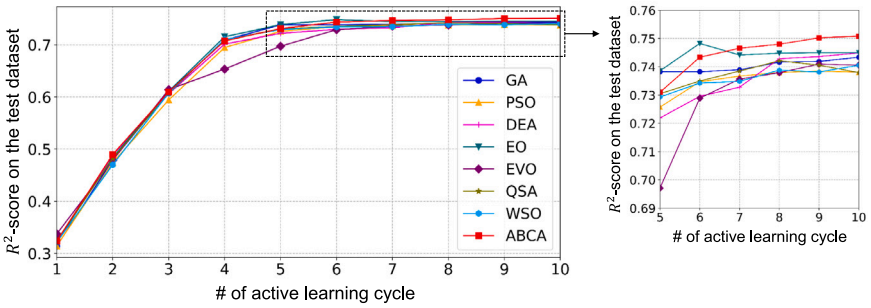

Fig. 8. Test R^2-scores of the prediction models trained with ALMS for different implementations of the metaheuristic-guided data generation of ALMS. Eight metaheuristic algorithms were employed to implement the metaheuristic-guided data generation of ALMS.

4.7. Limitations and future work of method development

Although ALMS finally achieved the R^2-score of 0.75 through the generative active learning, the six reaction conditions in Table 1 would be insufficient to completely describe the overall process of NOCM and there could be hidden variables that were not recorded during the experiments. However, it is not feasible to collect all the reaction conditions in NOCM due to the technical limitations of the measurements. The experiments of measuring NOCM in an actual reaction environment present a significant challenge [90,91]. In other words, we need to consider the trade-off between the accurate input features and the practicality of the machine learning models. Therefore, in future work, we need to conduct a comprehensive investigation to select more reaction conditions that can be efficiently collected.

Another limitation of ALMS is the approximation errors from the sliced Wasserstein distance. ALMS employed the sliced Wasserstein distance to efficiently calculate the Wasserstein distance in multi-dimensional feature spaces during the active learning cycle and the learning rate adaptation process. In this process, an approximation error between the exact and sliced Wasserstein distances is inevitable, and it can be propagated to the entire active learning performance of ALMS. In future work, it would be necessary to consider a more accurate approximation method than the sliced Wasserstein distance.

In addition to the limitations related to the prediction accuracy, the general applicability of ALMS for other chemical engineering applications presents a new challenge. However, this issue can be resolved efficiently if the input data of the target problem can be defined as a feature vector because ALMS is a universally applicable method to the problems in the vector spaces. Therefore, ALMS can be applied to problems in other engineering fields as well as further developments in NOCM containing different and more complicated engineering conditions.

5. Conclusion

We proposed a new method for active learning without pre-defined unlabeled datasets to avoid the expensive costs of preparing unlabeled datasets in chemical engineering. The proposed method, called ALMS, employs the metaheuristic algorithm in the framework of the active learning to generate new training data based on the predictive uncertainty of the prediction models. We applied ALMS to the engineering problem to maximize the C_2 yields that indicate the amount of the production of the desired products, while suppressing the formation of by-products by optimizing the reaction conditions of nonoxidative direct conversion of methane. By employing the prediction model trained by ALMS, we successfully discovered new reaction conditions that maximize C_2 yield. Quantitatively, based on the ALMS-based prediction model, we significantly reduced MAE from 1.38 to 0.4 in predicting the C_2 yields of the top-10 reaction conditions. The experimental results demonstrate the usefulness of ALMS in the green manufacturing systems of chemical engineering, where it is impractical to prepare extensive unlabeled datasets for active learning. One of the promising future work of ALMS is to employ ALMS as a prior optimization method to other regression tasks in chemical engineering. As a direct extension of ALMS in NOCM problem, we can consider developing a new catalytic process for the reaction conditions optimized by ALMS.

CRediT authorship contribution statement

Gyoung S. Na: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Hyun Woo Kim:** Writing – review & editing, Validation, Resources, Methodology, Funding acquisition, Formal analysis.

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

Our data and code have been shared at https://github.com/ngs00/alms.

Acknowledgments

This study was supported by a project from the Korea Research Institute of Chemical Technology (KK2351-10). This research was supported by the Korea Instititue for Advancement of Technology (KIAT) through the Virtual Engineering Platform Program (P0022334).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.asoc.2024.111935.

References

[1] L. Chen, G. Msigwa, M. Yang, A.I. Osman, S. Fawzy, D.W. Rooney, P.-S. Yap, Strategies to achieve a carbon neutral society: a review, Environ. Chem. Lett. 20 (4) (2022) 2277–2310.
[2] R.O. Yusuf, Z.Z. Noor, A.H. Abba, M.A.A. Hassan, M.F.M. Din, Methane emission by sectors: a comprehensive review of emission sources and mitigation methods, Renew. Sustain. Energy Rev. 16 (7) (2012) 5059–5070.
[3] S.T. Wismann, J.S. Engbæk, S.B. Vendelbo, F.B. Bendixen, W.L. Eriksen, K. Aasberg-Petersen, C. Frandsen, I. Chorkendorff, P.M. Mortensen, Electrified methane reforming: A compact approach to greener industrial hydrogen production, Science 364 (6442) (2019) 756–759.
[4] A.J. Pieja, M.C. Morse, A.J. Cal, Methane to bioproducts: the future of the bioeconomy? Curr. Opin. Chem. Biol. 41 (2017) 123–131.
[5] J.J. Spivey, G. Hutchings, Catalytic aromatization of methane, Chem. Soc. Rev. 43 (3) (2014) 792–803.
[6] S. Nachimuthu, H.-J. Lai, Y.-C. Chen, J.-C. Jiang, Comparable catalytic activity of a low-cost catalyst IrO2/TiO2 for methane conversion–A density functional theory study, Appl. Surf. Sci. 577 (2022) 151938.
[7] M. Varbar, S.M. Alavi, M. Rezaei, E. Akbari, Cobalt promoted Ni/MgAl 2 O 4 catalyst in lean methane catalytic oxidation, Res. Chem. Intermed. (2022) 1–22.
[8] P. Schwach, X. Pan, X. Bao, Direct conversion of methane to value-added chemicals over heterogeneous catalysts: challenges and prospects, Chem. Rev. 117 (13) (2017) 8497–8520.
[9] M. He, Y. Sun, B. Han, Green carbon science: efficient carbon resource processing, utilization, and recycling towards carbon neutrality, Angew. Chem. Int. Edn 61 (15) (2022) e202112835.
[10] S. Konnov, Direct non-oxidative conversion of methane over metal-containing zeolites: Main strategies for shifting the thermodynamic equilibrium (a review), Pet. Chem. 62 (3) (2022) 280–290.
[11] X. Meng, X. Cui, N.P. Rajan, L. Yu, D. Deng, X. Bao, Direct methane conversion under mild condition by thermo-, electro-, or photocatalysis, Chem 5 (9) (2019) 2296–2325.
[12] T. Zhang, Recent advances in heterogeneous catalysis for the nonoxidative conversion of methane, Chem. Sci. 12 (38) (2021) 12529–12545.
[13] E.-h. Sim, S.W. Lee, J.J. Lee, S.J. Han, J.H. Shin, G. Lee, S. Ko, K.-Y. Lee, Y.T. Kim, Effect of silicon carbide-based iron catalyst on reactor optimization for non-oxidative direct conversion of methane, J. Energy Chem. 81 (2023) 519–532.
[14] X. Guo, G. Fang, G. Li, H. Ma, H. Fan, L. Yu, C. Ma, X. Wu, D. Deng, M. Wei, et al., Direct, nonoxidative conversion of methane to ethylene, aromatics, and hydrogen, Science 344 (6184) (2014) 616–619.
[15] H.W. Kim, S.W. Lee, G.S. Na, S.J. Han, S.K. Kim, J.H. Shin, H. Chang, Y.T. Kim, Reaction condition optimization for non-oxidative conversion of methane using artificial intelligence, React. Chem. Eng. 6 (2) (2021) 235–243.
[16] J. Hao, P. Schwach, G. Fang, X. Guo, H. Zhang, H. Shen, X. Huang, D. Eggart, X. Pan, X. Bao, Enhanced methane conversion to olefins and aromatics by H-donor molecules under nonoxidative condition, ACS Catal. 9 (10) (2019) 9045–9050.
[17] R. Postma, P. Mendes, J. Thybaut, L. Lefferts, Modelling of the catalytic initiation of methane coupling under non-oxidative conditions, Chem. Eng. J. 454 (2023) 140273.
[18] J. Liu, J. Yue, M. Lv, F. Wang, Y. Cui, Z. Zhang, G. Xu, From fundamentals to chemical engineering on oxidative coupling of methane for ethylene production: A review, Carbon Resour. Convers. 5 (1) (2022) 1–14.
[19] C.J. Taylor, A. Pomberger, K.C. Felton, R. Grainger, M. Barecka, T.W. Chamberlain, R.A. Bourne, C.N. Johnson, A.A. Lapkin, A brief introduction to chemical reaction optimization, Chem. Rev. 123 (6) (2023) 3089–3126.
[20] C. Hasselgren, T.I. Oprea, Artificial intelligence for drug discovery: Are we there yet? Annu. Rev. Pharmacol. Toxicol. 64 (2024) 527–550.
[21] A.V. Singh, M. Varma, P. Laux, S. Choudhary, A.K. Datusalia, N. Gupta, A. Luch, A. Gandhi, P. Kulkarni, B. Nath, Artificial intelligence and machine learning disciplines with the potential to improve the nanotoxicology and nanomedicine fields: a comprehensive review, Arch. Toxicol. 97 (4) (2023) 963–979.
[22] A.V. Singh, M. Varma, M. Rai, S. Pratap Singh, G. Bansod, P. Laux, A. Luch, Advancing predictive risk assessment of chemicals via integrating machine learning, computational modeling, and chemical/nano-quantitative structure-activity rrelationship approaches, Adv. Intell. Syst. (2024) 2300366.
[23] A. Nandy, C. Duan, M.G. Taylor, F. Liu, A.H. Steeves, H.J. Kulik, Computational discovery of transition-metal complexes: from high-throughput screening to machine learning, Chem. Rev. 121 (16) (2021) 9927–10000.
[24] T. Wuest, D. Weimer, C. Irgens, K.-D. Thoben, Machine learning in manufacturing: advantages, challenges, and applications, Prod. Manuf. Res. 4 (1) (2016) 23–45.
[25] K.K. Yang, Z. Wu, F.H. Arnold, Machine-learning-guided directed evolution for protein engineering, Nat. Methods 16 (8) (2019) 687–694.
[26] E.H. Houssein, I.E. Ibrahim, M. Kharrich, S. Kamel, An improved marine predators algorithm for the optimal design of hybrid renewable energy systems, Eng. Appl. Artif. Intell. 110 (2022) 104722.
[27] A.M. Schweidtmann, E. Esche, A. Fischer, M. Kloft, J.-U. Repke, S. Sager, A. Mitsos, Machine learning in chemical engineering: A perspective, Chem. Ing. Tech. 93 (12) (2021) 2029–2039.
[28] G.S. Na, C. Park, Nonlinearity encoding for extrapolation of neural networks, in: ACM SIGKDD, 2022, pp. 1284–1294.
[29] H. Yamada, C. Liu, S. Wu, Y. Koyama, S. Ju, J. Shiomi, J. Morikawa, R. Yoshida, Predicting materials properties with little data using shotgun transfer learning, ACS Cent. Sci. 5 (10) (2019) 1717–1730.
[30] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B.B. Gupta, X. Chen, X. Wang, A survey of deep active learning, ACM Comput. Surv. 54 (9) (2021) 1–40.
[31] T. Lookman, P.V. Balachandran, D. Xue, R. Yuan, Active learning in materials science with emphasis on adaptive sampling using uncertainties for targeted design, NPJ Comput. Mater. 5 (1) (2019) 21.
[32] D. Reker, G. Schneider, Active-learning strategies in computer-assisted drug discovery, Drug Discov. Today 20 (4) (2015) 458–465.
[33] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B.B. Gupta, X. Chen, X. Wang, A survey of deep active learning, ACM Comput. Surv. 54 (9) (2021) 1–40.
[34] V.-L. Nguyen, M.H. Shaker, E. Hüllermeier, How to measure uncertainty in uncertainty sampling for active learning, Mach. Learn. 111 (1) (2022) 89–122.
[35] A. Amirinezhad, S. Salehkaleybar, M. Hashemi, Active learning of causal structures with deep reinforcement learning, Neural Netw. 154 (2022) 22–30.
[36] J. Gong, Z. Fan, Q. Ke, H. Rahmani, J. Liu, Meta agent teaming active learning for pose estimation, in: CVPR, 2022, pp. 11079–11089.
[37] P. Liu, L. Wang, R. Ranjan, G. He, L. Zhao, A survey on active deep learning: From model driven to data driven, ACM Comput. Surv. 54 (10s) (2022) 1–34.
[38] G. Towler, R. Sinnott, Chemical Engineering Design: Principles, Practice and Economics of Plant and Process Design, Butterworth-Heinemann, 2021.
[39] D.R. Jones, M. Schonlau, W.J. Welch, Efficient global optimization of expensive black-box functions, J. Global Optim. 13 (1998) 455–492.
[40] J.A.G. Torres, S.H. Lau, P. Anchuri, J.M. Stevens, J.E. Tabora, J. Li, A. Borovika, R.P. Adams, A.G. Doyle, A multi-objective active learning platform and web app for reaction optimization, J. Am. Chem. Soc. 144 (43) (2022) 19999–20007.
[41] B.J. Shields, J. Stevens, J. Li, M. Parasram, F. Damani, J.I.M. Alvarado, J.M. Janey, R.P. Adams, A.G. Doyle, Bayesian reaction optimization as a tool for chemical synthesis, Nature 590 (7844) (2021) 89–96.
[42] M. Christensen, L.P. Yunker, F. Adedeji, F. Häse, L.M. Roch, T. Gensch, G. dos Passos Gomes, T. Zepel, M.S. Sigman, A. Aspuru-Guzik, et al., Data-science driven autonomous process optimization, Commun. Chem. 4 (1) (2021) 112.
[43] A.M. Nambiar, C.P. Breen, T. Hart, T. Kulesza, T.F. Jamison, K.F. Jensen, Bayesian optimization of computer-proposed multistep synthetic routes on an automated robotic flow platform, ACS Cent. Sci. 8 (6) (2022) 825–836.
[44] A. Vriza, H. Chan, J. Xu, Self-driving laboratory for polymer electronics, Chem. Mater. 35 (8) (2023) 3046–3056.
[45] R.J. Hickman, P. Bannigan, Z. Bao, A. Aspuru-Guzik, C. Allen, Self-driving laboratories: A paradigm shift in nanomedicine development, Matter 6 (4) (2023) 1071–1081.
[46] L.V. Jospin, H. Laga, F. Boussaid, W. Buntine, M. Bennamoun, Hands-on Bayesian neural networks—A tutorial for deep learning users, IEEE Comput. Intell. Mag. 17 (2) (2022) 29–48.
[47] S. Wu, L. Wang, J. Zhang, Photocatalytic non-oxidative coupling of methane: Recent progress and future, J. Photochem. Photobiol. 46 (2021) 100400.
[48] N.R. Draper, H. Smith, Applied Regression Analysis, third ed., Wiley-Interscience, 1998.
[49] T.G. Gebreyohannes, S.W. Lee, S.J. Han, Y.T. Kim, S.K. Kim, Unveiling the complexity of non-oxidative coupling of methane: A simplified kinetics approach, Chem. Eng. J. (2023) 144216.
[50] X. Huang, D. Lv, C. Zhang, X. Yao, Machine-learning reveals the virtual screening strategies of solid hydrogen-bonded oligomeric assemblies for thermo-responsive applications, Chem. Eng. J. 456 (2023) 141073.
[51] M. Kim, S. Kang, H.G. Park, K. Park, K. Min, Maximizing the energy density and stability of Ni-rich layered cathode materials with multivalent dopants via machine learning, Chem. Eng. J. 452 (2023) 139254.

[52] Y. Chen, C. Liu, G. Guo, Y. Zhao, C. Qian, H. Jiang, B. Shen, D. Wu, F. Cao, H. Sun, Machine-learning-guided reaction kinetics prediction towards solvent identification for chemical absorption of carbonyl sulfide, Chem. Eng. J. 444 (2022) 136662.
[53] B. Meredig, E. Antono, C. Church, M. Hutchinson, J. Ling, S. Paradiso, B. Blaiszik, I. Foster, B. Gibbons, J. Hattrick-Simpers, et al., Can machine learning identify the next high-temperature superconductor? examining extrapolation performance for materials discovery, Mol. Syst. Des. Eng. 3 (5) (2018) 819–825.
[54] K. Choudhary, B. DeCost, C. Chen, A. Jain, F. Tavazza, R. Cohn, C.W. Park, A. Choudhary, A. Agrawal, S.J. Billinge, E. Holm, S.P. Ong, C. Wolverton, Recent advances and applications of deep learning methods in materials science, NPJ Comput. Mater. 8 (1) (2022) 59.
[55] Z. Zhong, L. Zheng, G. Kang, S. Li, Y. Yang, Random erasing data augmentation, in: AAAI, Vol. 34, 2020, pp. 13001–13008.
[56] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6 (1) (2019) 1–48.
[57] S.C. Wong, A. Gatt, V. Stamatescu, M.D. McDonnell, Understanding data augmentation for classification: when to warp? in: DICTA, IEEE, 2016, pp. 1–6.
[58] C. Shorten, T.M. Khoshgoftaar, B. Furht, Text data augmentation for deep learning, J. Big Data 8 (2021) 1–34.
[59] Q. Wen, L. Sun, F. Yang, X. Song, J. Gao, X. Wang, H. Xu, Time series data augmentation for deep learning: A survey, 2020, arXiv preprint arXiv:2002.12478.
[60] A.V. Singh, G. Bansod, M. Mahajan, P. Dietrich, S.P. Singh, K. Rav, A. Thissen, A.M. Bharde, D. Rothenstein, S. Kulkarni, et al., Digital transformation in toxicology: improving communication and efficiency in risk assessment, ACS Omega 8 (24) (2023) 21377–21390.
[61] A. Antoniou, A. Storkey, H. Edwards, Data augmentation generative adversarial networks, 2017, arXiv preprint arXiv:1711.04340.
[62] P.M. Bueno, M. Jino, W.E. Wong, Diversity oriented test data generation using metaheuristic search techniques, Inform. Sci. 259 (2014) 490–509, http://dx.doi.org/10.1016/j.ins.2011.01.025.
[63] J. Correia, T. Martins, P. Machado, Evolutionary data augmentation in deep face detection, in: GECCO, Association for Computing Machinery, New York, NY, USA, 2019, pp. 163–164.
[64] E. Kaya, B. Gorkemli, B. Akay, D. Karaboga, A review on the studies employing artificial bee colony algorithm to solve combinatorial optimization problems, Eng. Appl. Artif. Intell. 115 (2022) 105311.
[65] Y. Kim, B. Shin, In defense of core-set: A density-aware core-set selection for active learning, in: ACM SIGKDD, 2022, pp. 804–812.
[66] B. Xie, L. Yuan, S. Li, C.H. Liu, X. Cheng, G. Wang, Active learning for domain adaptation: An energy-based approach, in: AAAI, Vol. 36, 2022, pp. 8708–8716.
[67] J. Wu, J. Chen, D. Huang, Entropy-based active learning for object detection with progressive diversity constraint, in: CVPR, 2022, pp. 9397–9406.
[68] D. Yoo, I.S. Kweon, Learning loss for active learning, in: CVPR, 2019, pp. 93–102.
[69] C.K. Williams, C.E. Rasmussen, Gaussian Processes for Machine Learning, Vol. 2, MIT press Cambridge, MA, 2006.
[70] Y. Wang, T.-Y. Chen, D.G. Vlachos, NEXTorch: a design and Bayesian optimization toolkit for chemical sciences and engineering, J. Chem. Inf. Model. 61 (11) (2021) 5312–5319.
[71] J.H. Dunlap, J.G. Ethier, A.A. Putnam-Neeb, S. Iyer, S.-X.L. Luo, H. Feng, J.A.G. Torres, A.G. Doyle, T.M. Swager, R.A. Vaia, et al., Continuous flow synthesis of pyridinium salts accelerated by multi-objective Bayesian optimization with active learning, Chem. Sci. 14 (30) (2023) 8061–8069.
[72] S. Mirjalili, S. Mirjalili, Genetic algorithm, Neural Comput. Appl. (2019) 43–55.
[73] J. Kennedy, R. Eberhart, Particle swarm optimization, in: ICNN, Vol. 4, IEEE, 1995, pp. 1942–1948.
[74] A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium optimizer: A novel optimization algorithm, Knowl.-Based Syst. 191 (2020) 105190.
[75] D. Karaboga, B. Akay, A comparative study of artificial bee colony algorithm, Appl. Math. Comput. 214 (1) (2009) 108–132.
[76] T. Ye, W. Wang, H. Wang, Z. Cui, Y. Wang, J. Zhao, M. Hu, Artificial bee colony algorithm with efficient search strategy based on random neighborhood structure, Knowl.-Based Syst. 241 (2022) 108306.
[77] A. Lipowski, D. Lipowska, Roulette-wheel selection via stochastic acceptance, Phys. A 391 (6) (2012) 2193–2196.
[78] M. Ren, W. Zeng, B. Yang, R. Urtasun, Learning to reweight examples for robust deep learning, in: ICML, PMLR, 2018, pp. 4334–4343.
[79] L. Rüschendorf, The Wasserstein distance and approximation theorems, Probab. Theory Related Fields 70 (1) (1985) 117–129.
[80] B.-E. Chérief-Abdellatif, Convergence rates of variational inference in sparse deep learning, in: ICML, PMLR, 2020, pp. 1831–1842.
[81] J.C. Bansal, A. Gopal, A.K. Nagar, Analysing convergence, consistency, and trajectory of artificial bee colony algorithm, IEEE Access 6 (2018) 73593–73602.
[82] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017, arXiv:1412.6980.
[83] S. Kolouri, K. Nadjahi, U. Simsekli, R. Badeau, G. Rohde, Generalized sliced wasserstein distances, in: NeurIPS, Vol. 32, 2019.
[84] N. Paudel, M. Rai, S. Adhikari, A. Thapa, S. Bharati, B. Maharjan, R.L.S. Shrestha, K. Rav, A.V. Singh, Green extraction, phytochemical profiling, and biological evaluation of dysphania ambrosioides: an in silico and in vitro medicinal investigation, J. Herbs Spices Med. Plants 30 (2) (2024) 97–114.
[85] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008).
[86] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (1997) 341–359.
[87] M. Azizi, U. Aickelin, H. A. Khorshidi, M. Baghalzadeh Shishehgarkhaneh, Energy valley optimizer: a novel metaheuristic algorithm for global and engineering optimization, Sci. Rep. 13 (1) (2023) 226.
[88] J. Zhang, M. Xiao, L. Gao, Q. Pan, Queuing search algorithm: A novel metaheuristic algorithm for solving engineering optimization problems, Appl. Math. Model. 63 (2018) 464–490.
[89] T.S. Ayyarao, N. Ramakrishna, R.M. Elavarasan, N. Polumahanthi, M. Rambabu, G. Saini, B. Khan, B. Alatas, War strategy optimization algorithm: a new effective metaheuristic algorithm for global optimization, IEEE Access 10 (2022) 25073–25105.
[90] T. Wada, U. Kashaboina, Y. Nishikawa, Y. Wakisaka, D. Bao, S. Takakusagi, Y. Inami, F. Kuriyama, A.L. Dipu, H. Ogihara, et al., The effect of structural change during the activation process on the catalysis of In/SiO2 nonoxidative coupling of methane: An operando XAFS study, J. Phys. Chem. C 127 (50) (2023) 24211–24222.
[91] D. Eggart, X. Huang, A. Zimina, J. Yang, Y. Pan, X. Pan, J.-D. Grunwaldt, Operando XAS study of pt-doped CeO2 for the nonoxidative conversion of methane, ACS Catal. 12 (7) (2022) 3897–3908.