This is the Pre-Published Version.

J. Shi, W. Xie, X. Huang, F. Xiao, A. Usmani, F. Khan, X. Yin, G. Chen (2022) *Real-time natural gas release forecasting by using physics-guided deep learning probability model*, **Journal of Cleaner Production**, 368, 133201. https://doi.org/10.1016/j.jclepro.2022.133201

Real-time natural gas release forecasting by using physics-guided deep learning probability model

Jihao Shi[1,*], Weikang Xie[1], Xinyan Huang[2,*], Fu Xiao[2], Asif Sohail Usmani[2], Faisal Khan[3], Xiaokang Yin[1], Guoming Chen[1]

[1] Centre for Offshore Engineering and Safety Technology, China University of Petroleum, Qingdao 266580, China

[2] Department of Building Environment and Energy Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, China

[3] Mary Kay O'Connor Process Safety Center (MKOPSC), Artie McFerrin Department of Chemical Engineering, Texas A&M University, College Station TX 77843-3122, USA

[*] *Corresponding to shi_jihao@163.com; xy.huang@polyu.edu.hk*

Abstract

Natural gas release from oil and gas facilities contributes significantly to the greenhouse effect and reduces the benefit of displacing heavy fossil fuels with natural gas. Real-time concentration spatiotemporal evolution forecasting of natural gas release is essential to predetermine atmospheric carbon trajectory and devise timely strategy to mitigate the expected impact on the environment. Deep learning approaches have recently been applied for spatiotemporal forecast tasks, but they still exhibit poor performance pertaining to uncertainty and boundary estimations. This study proposes an advanced Hybrid-Physics Guided-Variational Bayesian Spatial-Temporal neural network. Experimental study based on a benchmark experimental and simulation dataset was conducted. The results demonstrated that the additional uncertainty information estimated contributes to reducing the harmful 'over confidence' of the point-estimation models at the plume area. Also, the proposed normalized uncertainty and physical inconsistency constraint term ensured the accuracy at the plume boundary. By adopting the Monte Carlo sampling number m=100, penalty factor λ=0.1, and drop probability p=0.1, the model achieves a real-time capacity of an inference time less than 1s and a competitive accuracy of R^2=0.988. Overall, our proposed model could provide reliable support to maximize the environmental benefits of natural gas energy usage and contribute to the carbon peak and neutrality target.

Keywords: Greenhouse gas emission, Spatiotemporal forecasting, Variational Bayesian inference, Physic-informed neural network, Environmental pollution, Carbon peak and neutrality

Abbreviations:

CO_2 Carbon dioxide
CH_4 Methane
CFD Computational fluid dynamics
ARIMA Autoregressive moving average model
GRNN Generalized regression neural network
GM Gaussian Mixture
Hybrid-PG-VBSTnn Hybrid-Physics Guided-Variational Bayesian Spatial-Temporal neural network
KL Kullback-Leibler
ConvGRU Convolutional Gated Recurrent Unit
MLE Maximum likelihood estimation
MAP Maximum a posterior
EM Expectation maximization
MCMC Markov chain Monte Carlo
ELBO Evidence Lower Bound
MSE Mean square error
SGD Stochastic gradient descends
FLACS FLame ACceleration Simulator
CFLC Courant-Friedrichs-Lewy number based sound speed
CFLV Courant-Friedrichs-Lewy number based flow velocity
LNG Liquid natural gas
CDF Cumulative Density Function
PDF Probability Density Function

1. Introduction

Natural gas is a 'cleaner fuel source', compared to the traditional fossil fuels such as coal and petroleum [1, 2]. Burning natural gas produces nearly half as much carbon dioxide (CO_2) per energy unit compared with coal and oil [3]. This quality has made natural gas the most significant transition fuel in the journey towards carbon neutrality. The past decades have witnessed a tremendous growth of worldwide natural gas production and consumption [4-6]. However, it has also been the cause of many environmental and safety issues as a result of the fugitive release of natural gas [7]. No matter whether it is intentional or accidental, such release into the atmosphere could significantly reduce the climate benefit of the displacement of the traditional fossil fuels with natural gas due to the fact that its primary component, namely methane (CH_4) is a potent greenhouse gas with 34 times the global warming potential of CO_2 on a 100-year time scale [8, 9]. In order to effectively determine the atmospheric carbon trajectory and mitigate the negative impact on the future environment, forecasting the spatiotemporal dynamics of natural gas release is essential considering that the production and consumption of natural gas is unlikely to cease in a very short time such as to coincide with the 2030 carbon

emission peak committed to China [10].

To date, oil and gas facilities are still the major release source of natural gas in the atmosphere [11]. Natural gas release from one specific facility generally has micro-space-time-scale characteristic, which means the release distance may range from miles to kilometers and the duration may range from minutes to days. Furthermore, each natural gas release from the oil and gas facility plays a significant role in increasing the worldwide atmospheric greenhouse gas concentration. The damage to the environment could be considerably enhanced if an accidental natural gas release occurs, since a large amount of natural gas would be instantaneously released into the atmosphere [12-14]. Furthermore, if ignited, the release of natural gas into congested oil and gas facilities may also lead to fire and explosion disasters and aggravate environmental pollution and inflict financial losses in the longer term [15, 16]. Real-time forecasting of the spatiotemporal evolution of natural gas release from oil and gas facilities can become an essential tool to predetermine the atmospheric carbon trajectory and enable timely development of appropriate mitigation strategies towards reducing atmospheric greenhouse gases [17].

Efforts have been made toward forecasting various greenhouse gas emissions over large temporal and spatial scales by using various machine (deep) learning techniques. Xu et al. [18] proposed a novel adaptive grey model with buffered rolling method to forecast Chinese greenhouse gas emissions from 2017 to 2025 and verified its advantages over other models. The long-term (e.g. annual) and short-term (e.g. hourly) forecasting of CO_2 emissions intensity in power grids was conducted by implementing generalized regression neural network (GRNN) and autoregressive moving average model (ARIMA), respectively [19]. Jeffrey et al. [20] applied a hybrid Verhulst-GM model to forecast CO_2 emissions of China's cement industry. A novel sequence-to-sequence deep learning model with an attention mechanism was proposed to forecast the regional ozone concentration by using the benchmark data collected from 35 air quality monitoring stations and 651 meteorological grids in Beijing, China [21]. A novel robust reweighted multivariate grey model was proposed by Xie et al. [22] to forecast the greenhouse gas emissions in European Union member countries from 2010 to 2016. Bakay et al. conducted the comparisons among three algorithms, namely deep learning, support vector machine and artificial neural network algorithms for electricity production based forecasting of greenhouse gas emissions in Turkey, which demonstrated all algorithms could give desirable forecasting results[23]. Ren et al. proposed hybrid machine learning approach, namely the Fast Learning Network optimized by Chicken Swarm Optimization algorithm, which was verified to more accurate to forecasting carbon emission in Guangdong province, China, compared to single machine learning approaches [24]. Such recent research works focused on time-series greenhouse emission forecasting rather than the spatiotemporal forecasting.

The key challenge for the spatiotemporal forecasting of natural gas release from oil and gas facilities is collecting real-world spatiotemporal evolution concentration data for the machine (deep) learning model development. This is because the concentration sensors are generally sparsely distributed around the facility and could only record the time-series concentration data and not truly spatiotemporal data. Recently developed

advanced techniques have been applied to help make the spatiotemporal evolution of greenhouse gas, e.g., methane, visible and quantifiable, however, it remains difficult to collect a sufficiently large real-world dataset of atmospheric concentration plumes, especially ones caused by unprecedented accidental releases of natural gas [25, 26]. Some researchers have resorted to using high-fidelity computational fluid dynamics (CFD) tools well-validated by the experiments to generate synthetic datasets, including the spatiotemporal concentration plume from accidental natural gas release [27]. Na et al. [28] proposed a hybrid model by using variational autoencoder with convolutional neural networks to estimate released toxic gas dispersion in urban areas based on a large number of CFD-based benchmark datasets. Also, by using CFD-based benchmark dataset, Song et al. [29] applied the encoding-prediction neural network for spatiotemporal and layout-adaptive forecasting of gas release and dispersion. However, the above models still have inherent limitations in terms of spatiotemporal concentration evolution forecasting accuracy of natural gas release. Firstly, such deep learning models were based on point-estimation theory and thereby forecast the 'over-confident' and unreliable spatiotemporal evolution output at the plume area with high concentration. Secondly, due to not considering physical constraints, such models are likely to exhibit physical inconsistence and exhibit poor performance regarding spatiotemporal evolution forecasting at the plume boundary [30]. Therefore, there is still a room to improve spatiotemporal concentration evolution forecasting accuracy of natural gas release to reliably support real-time mitigation of the negative impact of natural gas release from oil and gas facilities

This study proposes a deep neural network namely, Hybrid-Physics Guided-Variational Bayesian Spatial-Temporal neural network (Hybrid-PG-VBSTnn). In this advanced model, the Spatial-Temporal neural network is employed as the backbone to discover both the spatial and temporal correlations from the released gas concentration data. Variational Bayesian inference is incorporated to model the probability density of the backbone's hyper-parameters. Finally, the known physical constraint term is introduced to the loss function as well as the training process to ensure the physical consistency of the boundary estimation. In terms of uncertainty estimation with deep learning, Variational Bayesian inference is one of the most well-known alternatives, which replaces the true posterior distribution with an approximate distribution and solves the Kullback-Leibler (KL) divergence between two distributions [31, 32]. Liu et al. integrated variation Bayesian inference with deep learning backbone for spatiotemporal wind speed forecasting [33]. The forecasted spatiotemporal uncertainty significantly improves the robustness and accuracy of state-of-the-art deep learning models . In addition, Ni et al. integrated variation Bayesian inference with Gaussian process to reliably update the critical parameters for structural health monitoring task [34]. Additionally, recent studies have demonstrated certain advantages regarding the incorporation of physics-guided constraints into loss function and training process in terms of the simplicity of hyper-parameter tuning and the computational efficiency [35, 36]. The major contributions and novelty of this study are listed as follows:

(1) Variational inference and physics-guided constraint are incorporated into deep learning backbone to improve spatiotemporal concentration evolution forecasting accuracy of natural gas release compared to the point-estimation deep learning model.

(2) Optimal pre-defined parameters are determined to ensure our proposed Hybrid-PG-VBSTnn model's accuracy as well as the real-time capability of the released plume concentration spatiotemporal evolution forecasting.

(3) Our proposed Hybrid-PG-VBSTnn model is a reliable technique to enable real-time mitigation of the negative impact of natural gas fugitive release from oil and gas facilities.

2. Problem formulation

Our objective is to forecast the K-steps-ahead carbon concentration spatiotemporal evolution induced by natural gas release based on the historical concentration sequences. From a mathematical perspective, this is to say we require to model the probability density $P(Y|X)$ of K-steps-ahead spatiotemporal concentration data Y given the previous J-steps concentration sequence X. X is expressed as $[X_1,X_2,\ldots,X_J]$, and Y is expressed as $[Y_{J+1},Y_{J+2},\ldots, Y_{J+K}]$, in which the concentration data X_J at time J, and Y_{J+K} at time $J+K$ are expressed as:

$$X_J=\begin{pmatrix} X_{1,1}^{J} & \cdots & X_{1,n2}^{J} \\ \vdots & \ddots & \vdots \\ X_{n1,1}^{J} & \cdots & X_{n1,n2}^{J} \end{pmatrix}, Y_{J+K}=\begin{pmatrix} Y_{1,1}^{J+K} & \cdots & Y_{1,n2}^{J+K} \\ \vdots & \ddots & \vdots \\ Y_{n1,1}^{J+K} & \cdots & Y_{n1,n2}^{J+K} \end{pmatrix} \quad (1)$$

where n_1 and n_2 denote any two of the length x, width y and height h of the potential hazardous domain generated by the released gas. n_1 and n_2 may be very large for real-world gas release and dispersion scenarios, which lead to the high dimensions of both X and Y at any specific time. For example, n_1 and n_2 may be over than 100 for a chemical plant with a large number of facilities, and the dimension of X or Y could be over $10000=100\times100$.

In this study, we apply deep neural network-based approach to model the probability density $P(Y|X)$, which is expressed as:

$$P(Y\mid X)=\int P(Y\mid w,X)P(w)dw,\ w\sim P(w\mid X,Y) \quad (2)$$

where w presents the hyper-parameters in the deep neural network. $P(Y|w,X)$ denotes the conditional probability density of the K-steps-ahead spatiotemporal concentration data Y given the previous J-steps concentration sequence X as well as the hyper-parameters w of the deep neural network. $P(w|X,Y)$ denotes the probability density of the model hyper-parameters given X and Y. As is expressed in Eq. (2), we require to first model $P(w|X,Y)$ and then extract w from such probability distribution to determine $P(Y|X)$. However, three main challenges still exist as follows:

(1) build the deep neural network not only to extract the high dimension spatial features of concentration data at any historical time, but also capture the temporal evolution characteristics of the spatial concentration features;

(2) model the probability density $P(w|X,Y)$ from which we could iteratively extract w to estimate the uncertainty of the spatiotemporal concentration feature in the case of the complex architecture of deep neural network;

(3) ensure the physical consistency of spatiotemporal concentration evolution forecasting at the plume boundary by incorporating the already-known physical constraint.

3. Physics-guided probabilistic spatial-temporal neural network

This study develops the physics-guided probabilistic spatiotemporal deep neural network to solve the above three challenges. The Spatial-Temporal deep neural network integrating the Convolutional encoder, ConvGRU, and Transpose Convolutional forecast, discovers the correlations of both spatial and temporal features with high dimensions in the spatiotemporal concentration data. Subsequently, Variational Bayesian inference is incorporated to model the probability density $P(w|X,Y)$ and accordingly estimate the spatiotemporal evolution uncertainty. At last, the already-known physical constraint term is introduced to the loss function as well as the training process to ensure the physical consistency of the boundary estimation. Such consistency would contribute to identifying the boundary between the areas with and without the released gas by using the proposed normalized uncertainty contour. We name the constructed deep neural network as Hybrid-Physics Guided-Variational Bayesian Spatial-Temporal neural network (Hybrid-PG-VBSTnn).

a) Spatial-Temporal deep neural network

The proposed Spatial-Temporal deep neural network first applies the Convolutional encoder to reduce the high dimensionality and then extract the critical spatial features X_s of the previous J-steps concentration sequence X, which is expressed as:

$$X_s = Conv\_encoder^{w1}(X) \tag{3}$$

where $w1$ denotes the hyper-parameters in the convolutional encoder neural layers.

Subsequently, ConvGRU is applied to capture the temporal features $X_{s\text{-}t\_corr}$ of the critical spatial information in X_s as follows:

$$X_{s-t\_corr} = ConvGRU^{w2}(X_s, H_0) \tag{4}$$

Eq.(4) could be further expressed as below:

$$X_{s-t\_corr} = f^{wy}(H_t, \ldots, H_0) \tag{5}$$

where $f(*)$ presents the activation function of the output layer in ConvGRU. wy denotes the hyper-parameters in the output layer. In Eq.(4), H_t could be expressed as :

$$H_t = (1 - Z_t) \circ H_{t-1} + Z_1 \circ H_t^{'} \tag{6}$$

$$H_t^{'} = \tanh(W_{xh} * X_t + R_t \circ (W_{hh} * H_{t-1}) + b_h) \tag{7}$$

$$R_t = \text{sigmoid}(W_{xr} * X_t + W_{hr} * H_{t-1} + b_r) \tag{8}$$

$$Z_t = \text{sigmoid}(W_{xz} * X_t + W_{hz} * H_{t-1} + b_z) \tag{9}$$

where W_{xh} W_{hh} W_{xr} W_{hr} W_{xz} W_{hz} denote the weights, b_h b_z b_r denote the bias in the hidden layers. Please noting that $w2$ presents all the hyper-parameters in ConvGRU, i.e., $w2$={ W_{xh} W_{hh} W_{xr} W_{hr} W_{xz} W_{hz} b_h b_z b_r wy}. Furthermore, * denotes the convolutional operator; o denotes the Hadamard product; sigmoid(·) denotes the sigmoid activation function; tanh(·) denotes the tanh activation function; H_t, $H_t^{'}$, Z_t and R_t present the hidden states, new information, reset gate, and update gate, respectively. Fig.1 demonstrates the architecture of ConvGRU.

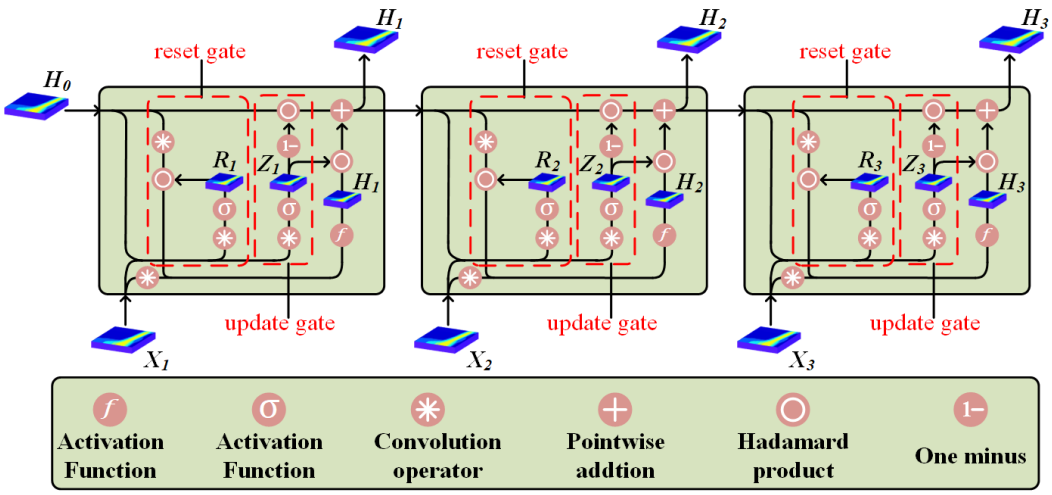


Fig. 1. Architecture of ConvGRU

At last, the Convolutional forecast consisting of the transposed convolutional layers and 3D convolutional layer is constructed to estimate the output, the dimensions of which are same with the input X. The deconvolution operation can be expressed as:

$$Y = Conv\_forecast^{w3}(X_{s-t\_corr}) \tag{10}$$

where $w3$ denotes the hyper-parameters in the convolutional forecast. Given the previous J-steps concentration sequence X, we could calculate the K-steps-ahead spatiotemporal concentration data Y by using the proposed Spatial-Temporal deep neural network as follows:

$$Y = Conv\_forecast^{w3}(ConvGRU^{w2}(Conv\_encoder^{w1}(X), H_0)) \tag{11}$$

where we denote all the hyper-parameters $\{w1\ w2\ w3\}$ as w in the proposed Spatial-Temporal deep neural network. In the next step, we would optimize the model hyper-parameters w to maximize the log probability of $P(Y \mid X) = \int P(Y \mid X, w)P(w)dw$

b) Variational Bayesian Spatial-Temporal deep neural network

Point estimation approaches, e.g., maximum likelihood estimation (MLE), maximum a posterior (MAP), expectation maximization (EM) etc., have been widely applied to optimize w to maximize the log probability of $P(Y|X)$ [34, 37]. However, the point estimation approaches could only determine an optimal list of w values rather the posterior distribution $P(w|X,Y)$, from which the proposed Spatial-Temporal deep neural network could estimate the uncertainty related to the forecasted spatiotemporal concentration by iteratively extracting a list of w values. In this case, Bayesian inference is applied to model $P(w|X,Y)$, the log probability of which is expressed as:

$$\log P(w \mid D) = \log\left(\frac{P(D \mid w)P(w)}{P(D)}\right) \tag{12}$$

where we denote (X,Y) as D. $P(D|w)$ is the likelihood of D given w, $P(w)$ is the prior probability of initially assumed a list of w values, $P(D)$ is the marginal probability. $P(D|w)$ could be further expressed as [34]:

$$P(D|w)=\frac{1}{\sqrt{(2\pi)^N \det(\sum \varepsilon)}}\exp\left(-\frac{1}{2}\left(Y-M(w)\right)^T \sum\nolimits_{\varepsilon}^{-1}\left(Y-M(w)\right)\right) \tag{13}$$

where $M(w)=Conv\_forecast^{w3}(ConvGRU^{w2}(Conv\_encoder^{w1}(X),H_0))$. The posterior density $P(w|D)$ usually does not have a closed-form solution. In the previous study, Markov chain Monte Carlo (MCMC) methods have been widely used to estimate such posterior probability distribution [38]. The application of MCMC requires the repeated evaluations of the likelihood $P(D|w)$ and the forward model $M(w)$. This process is much computational intensity because of the complex architecture and large number of hyper-parameters of the proposed Spatial-Temporal deep learning model [39]. In this case, we introduce Variational Bayesian inference approach which assumes a Variational posterior distribution $q_\varphi(w)$ to approximate the true posterior probability distribution $P(w|D)$. Then, we introduce $q_\varphi(w)$ into Eq.(13) to determine the expression as below[40] :

$$\begin{aligned}\log P(D) &= \int q_\varphi(w)\log\left(\frac{P(D|w)P(w)}{P(w|D)}\frac{q_\varphi(w)}{q_\varphi(w)}\right)dw \\ &= \int q_\varphi(w)\log\left(\frac{P(D|w)P(w)}{q_\varphi(w)}\right)dw+\int q_\varphi(w)\log\left(\frac{q_\varphi(w)}{P(w|D)}\right)dw \\ &= \int q_\varphi(w)\log\left(\frac{P(D|w)P(w)}{q_\varphi(w)}\right)dw+D_{KL}\left(q_\varphi(w)\,\|\,P(w|D)\right)\end{aligned} \tag{14}$$

where φ denotes the Variational parameters. The term of the left hand $\log P(D)$ is fixed for the certain dataset $D=(X,Y)$. The first term of the right hand is denoted as *Evidence Lower BOund* (*ELBO*), and the second terms is Kullback-Leibler (KL) divergence between both distributions, i.e., $P(w|D)$ and $q_\varphi(w)$. With the purpose of approximating $q_\varphi(w)$ to the true posterior probability distribution $P(w|D)$, we need to minimize the $KL(P(w|D)||q_\varphi(w))$. Since $\log P(D)$ is fixed, minimizing $D_{KL}(q_\varphi(w)||P(w|D))$ means maximizing the *ELBO* expressed as below [40]:

$$\begin{aligned}ELBO &= \int q_\varphi(w)\log\left(\frac{P(D|w)P(w)}{q_\varphi(w)}\right)dw \\ &= -\int q_\varphi(w)\log\left(\frac{q_\varphi(w)}{P(w)}\right)dw+\int \log P(D|w)q_\varphi(w)dw \\ &= -D_{KL}(q_\varphi(w)\,\|\,P(w))+E_{q_\varphi(w)}\log(P(D|w))\end{aligned} \tag{15}$$

The first term of the right hand is denoted as Kullback-Leibler (KL) divergence between both distributions. The second term is the expectation of $\log P(D|w)$, which can be further unbiasedly estimated as :

$$\begin{aligned}E_{q_\varphi(w)}\log(P(D|w)) &= \sum \log(P(Y|Conv_{forecast}{}^{w3}(ConvGRU^{w2}(Conv\_encoder^{w1}(X),H_0)))) \\ &= -MSE(Y-Conv_{forecast}{}^{w3}(ConvGRU^{w2}(Conv\_encoder^{w1}(X),H_0)))\end{aligned} \tag{16}$$

From Eq.(16), it can be seen that that maximizing the expected log probability is equivalent to minimizing the *mean square error* (*MSE*) between the real and predicted values. For the first term, i.e., KL divergence, it is assumed $P(w)=N(0,\sigma^2 I)$ to be Gaussian distribution with mean vector 0 and covariance matrix $\sigma^2 I$ [32].

According to the rule that any distribution could be modeled by multi Gaussian mixture distributions, $q_\varphi(w)$ is assumed to be a Gaussian mixture distribution with two components expressed as below[32].

$$q(w) = pN(\varphi, \sigma^2 I) + (1-p)N(0, \sigma^2 I) \tag{17}$$

Then $-D_{KL}(q_\varphi(w) \| P(w)) \approx \frac{p}{2}\varphi^T\varphi + K(\sigma^2 - \ln\sigma^2 - 1) + c$. φ is the variational parameter required to be optimized. p is the pre-defined probability. σ and c are scalar. We then calculate *ELBO* as:

$$ELBO \approx -MSE(Y - Conv_{forecast}{}^{w3}(ConvGRU^{w2}(Conv\_encoder^{w1}(X), H_0))) + \frac{p}{2}\varphi^T\varphi + K(\sigma^2 - \ln\sigma^2 - 1) + c \tag{18}$$

$MSE(\cdot)$ presents the constraints in the training process to ensure the forecasting accuracy at both areas with and without the released gas. The second term of the right hand presents the regularization constraints over the variational parameter φ. However, since there is no specific constraint of the boundary identifying the areas with and without the released gas, only considering Eq.(18) as the loss function might cause the physics inconsistency of the spatiotemporal forecasting at the boundary.

c) Already-known physical constraint term into the loss function

Before introducing the already-known physical constraint term, the mathematical model for compressible fluid flow is first introduced. The conservation of mass, momentum and energy are expressed as [41]:

$$\frac{\partial(\beta_v \rho)}{\partial t} + \frac{\partial(\beta_j \rho u_j)}{\partial x_j} = \frac{m}{V} \tag{19}$$

$$\frac{\partial(\beta_v \rho u_i)}{\partial t} + \frac{\partial(\beta_i \rho u_j u_i)}{\partial x_j} = -\beta_v \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}(\beta_i \delta_{ij}) + F + \beta_v(\rho - \rho_0) g_i \tag{20}$$

$$\frac{\partial(\beta_v \rho h)}{\partial t} + \frac{\partial(\beta_i \rho h u_j)}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\beta_i \frac{\mu_{\text{eff}}}{c_h}\frac{\partial h}{\partial x_j}\right) + \beta_v \frac{Dp}{Dt} + \frac{Q}{V} \tag{21}$$

where t is time, ρ is the density, ρ_0 is the ambient density, β_v is the volume porosity, β_j is the area porosity in the j^{th} direction, u_i is the mean velocity vector in the i^{th} direction, u_j is the mean velocity vector in the j direction, m is the mass rate, V is the volume, x_i is the length coordinate in the i direction, x_j is the length coordinate in the j^{th} direction, p is the absolute pressure, δ_{ij} is the stress tensor, F is the sum of flow resistance due to walls and flow resistance due to sub-grid obstructions, g_i is the gravitational acceleration in the i^{th} direction, h is the specific enthalpy, μ_{eff} is the effective viscosity, c_h is the Prandtl-Schmidt number, Q is the heat flow rate, V is the volume.

Furthermore, the transport equation for natural gas mass fraction Y_F[41]:

$$\frac{\partial(\beta_v \rho Y_F)}{\partial t} + \frac{\partial(\beta_i \rho Y_F u_j)}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\beta_i \frac{\mu_{\text{eff}}}{c_k}\frac{\partial Y_F}{\partial x_j}\right) + W_F \tag{22}$$

where W_F is the reaction rate for fuel, c_k is the Prandtl-Schmidt number.

Generally, the area closed to the release point is occupied by the released gas with high concentration.

Therefore, the gas relevant mass, momentum, energy and mass fraction values are positive. As the area becomes far away from the release point, these relevant values become closed and even equal to 0 at the boundary. This leads to the gas concentration as well as its variation closed to or even being 0. Using $MSE(\cdot)$ which presents the expectation of the posterior log probability as expressed in Eq.(18) could constraint the forecasted concentration at the area without gas approaching 0 at the most extent. However, it is unavoidable that such forecasted concentration at the area without released gas may be negative, and the corresponding variation may be large as well, which do not follow the physical law. Therefore, we propose the prior physics-based constraint term into the *ELBO* in order to constraint the forecasted value and its variation being 0 at the area without released gas. The loss function is expressed as:

$$loss = -ELBO + \lambda\left(\sum_{i=1}^{n}(Y_{bc_f}^{i} - Y_{bc_f}^{mean})^2/(n-1)\right)^{1/2} \tag{23}$$

where λ is penalty factor, $Y_{bc_f}^{i}$ denotes the i_th forecasted concentration at the area without the released gas, and $Y_{bc_f}^{mean} = \sum_{1}^{n} Y_{bc_f}^{i} / n$ denotes the mean forecasted concentration at the area without the released gas. The second term of the right hand is to constraint the variance of the forecasted concentrations approaching 0 at the boundary at each training step. By considering Eq.(23) as the objective function, the stochastic gradient descends (SGD) optimization algorithm is then applied to determine the optimal variational parameter φ.

d) Spatiotemporal concentration evolution forecasting with uncertainty

After determining the optimal φ, we obtain the $q_\varphi(w)$ to replace the $P(w|D)$. Given new input X_{new}, we further calculate $P(Y_{new}|X_{new})$ according to $P(Y_{new}\,|\,X_{new}) = \int P(Y_{new}\,|\,w, X_{new}) q_\varphi(w) dw, w \sim q_\varphi(w)$. Since analytically computing $P(Y_{new}|X_{new})$ is difficult, kernel density estimation (KDE) which is a classic non-parametric estimation method is applied [42].

We accordingly implement MC sampling m times to extract m groups of $[w^1,w^2,w^3,\ldots,w^n]$ from $q_\varphi(w)$, where n presents the number of hyper-parameters in the proposed Hybrid_PG_VBSTnn. Then, we calculate m groups of forecasted spatiotemporal results at time J+K expressed as follows:

$$\begin{gathered}[Y_{new}^{1}, Y_{new}^{2}, \ldots, Y_{new}^{m}]_{J+K} = Hybrid\_PG\_VBSTnn_{model}^{[w^1,w^2,w^3,\ldots,w^n]_m}(X_{new}), \\ [w^1, w^2, w^3, \ldots w^n]_m \sim q_\varphi(w)\end{gathered} \tag{24}$$

Furthermore, both the mean and variance of the predicted Y_{new} at time J+K given X_{new}=$(X_1 \ldots X_{n2})_{new}$ are eventually obtained.

$$Y_{new}^{mean} = \begin{pmatrix} Y_{1,1}^{mean} & \cdots & Y_{1,n2}^{mean} \\ \vdots & \ddots & \vdots \\ Y_{n1,1}^{mean} & \cdots & Y_{n1,n2}^{mean} \end{pmatrix}_{new} \tag{25}$$

$$Y_{new}^{var}=\begin{pmatrix} Y_{1,1}^{var} & \cdots & Y_{1,n2}^{var} \\ \vdots & \ddots & \vdots \\ Y_{n1,1}^{var} & \cdots & Y_{n1,n2}^{var} \end{pmatrix}_{new} \tag{26}$$

$$Y_{n1,n2}^{mean}=\frac{1}{m}\sum_{i=1}^{m}Y_{n1,n2}^{i},\ Y_{n1,n2}^{var}=\frac{1}{m}\sum_{i=1}^{m}(Y_{n1,n2}^{i}-Y_{n1,n2}^{mean})^{2} \tag{27}$$

where $Y_{n1,n2}^{i}$ is the forecasted concertation of each MC sampling iteration at the specific position. We denote the Y_{new}^{mean} as the forecasted spatiotemporal concentration, and $Y_{new}^{mean}+Y_{new}^{var}$ and $Y_{new}^{mean}-Y_{new}^{var}$ as the upper uncertainty and lower uncertainty, respectively. Also, we calculate the normalized variance $Y_{new}^{nor\_var}$ as the normalized uncertainty to identify the boundary between the areas with and without the released gas as follows:

$$Y_{new}^{nor\_var}=\begin{pmatrix} Y_{1,1}^{nor\_var} & \cdots & Y_{1,n2}^{nor\_var} \\ \vdots & \ddots & \vdots \\ Y_{n1,1}^{nor\_var} & \cdots & Y_{n1,n2}^{nor\_var} \end{pmatrix}_{new} \tag{28}$$

$$Y_{n1,n2}^{nor\_i}=(Y_{n1,n2}^{i}-min(Y_{n1,n2}))/max(Y_{n1,n2})-min(Y_{n1,n2})) \tag{29}$$

$$Y_{n1,n2}^{nor\_mean}=\frac{1}{m}\sum_{i=1}^{m}Y_{n1,n2}^{nor\_i},\ Y_{n1,n2}^{nor\_var}=\frac{1}{m}\sum_{i=1}^{m}(Y_{n1,n2}^{nor\_i}-Y_{n1,n2}^{nor\_mean})^{2} \tag{30}$$

Additionally, one should note that the larger number of MC samplings *m,* the more closed approximation to the probability distribution $P(Y_{new}|X_{new})$ by using Eq.(21) and Eq.(22). However, a large number of MC sampling significantly influence the real-time inference speed. Therefore, the optimal *m* is necessary to determine the trade-off between model accuracy and real-time capability. Also, we pre-define two parameters, i.e., penalty factor λ and drop probability *p* in the proposed Hybrid_PG_VBSTnn. The optimal values of both parameters are also required to be determined to ensure the accuracy as well as the real-time capacity of the proposed Hybrid_PG_VBSTnn in the following analysis.

4. Benchmark dataset

In this section, natural gas release experiment of a medium-scale LNG terminal is first conducted. Based on the experimental data, numerical model of gas release and dispersion is built and its accuracy is validated as well. At last, a large number of simulations are accordingly carried out to generate the benchmark synthetic dataset for the Hybrid_PG_VBSTnn development.

4.1. Natural gas release experimental and numerical modeling of a medium-scale LNG terminal

We first configure a medium-scale experiment system of natural gas release from an LNG terminal. The experimental system consists of three parts, namely a real medium-scale LNG terminal geometry, a gas transmission system, and a real-time concentration data collection system. The dimensions of the real LNG terminal geometry are length (x)= 6000mm, width (*y*)= 7300mm, and height (*z*)= 2000mm, respectively. The gas transmission system contains a steel cylinder filled with natural gas, and a flowmeter that could adjust the

gas release rate, and a release nozzle with diameter of 6 mm. Furthermore, we put the release nozzle at a point close to a pipeline. The coordinates of the release point are x=4260, y=5170, z=105, respectively. We also set one infrared sensor (Gon760-C3H8-200 fixed sensor with linearity error less than 1% and sampling interval 1s) closed to the release point. The coordinates of the sensor' location are x=4250, y=5083, z=100. Noting that such infrared sensor is. The in this experiment. Fig. 2 demonstrates the experimental schematic. Fig. 3 shows the real geometry of the medium-scale LNG terminal. In addition to the above configuration, we select three volumetric release rates, i.e., 4.7 L/min, 6.2 L/min, 7.7 L/min with release duration being 90s, and set the positive direction of y axis as the release direction. Table. 1 shows all the experimental configurations.

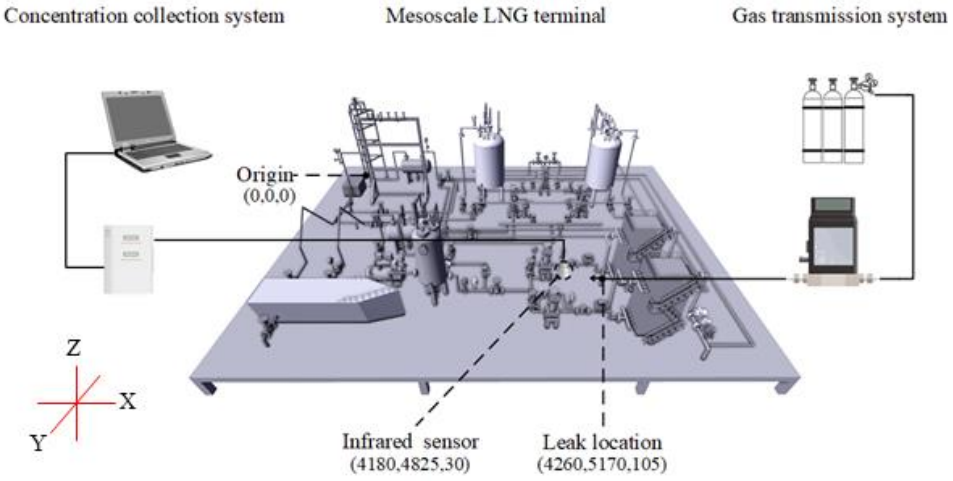


Fig. 2. Natural gas release experiment systerm schematic of a medium-scale LNG terminal

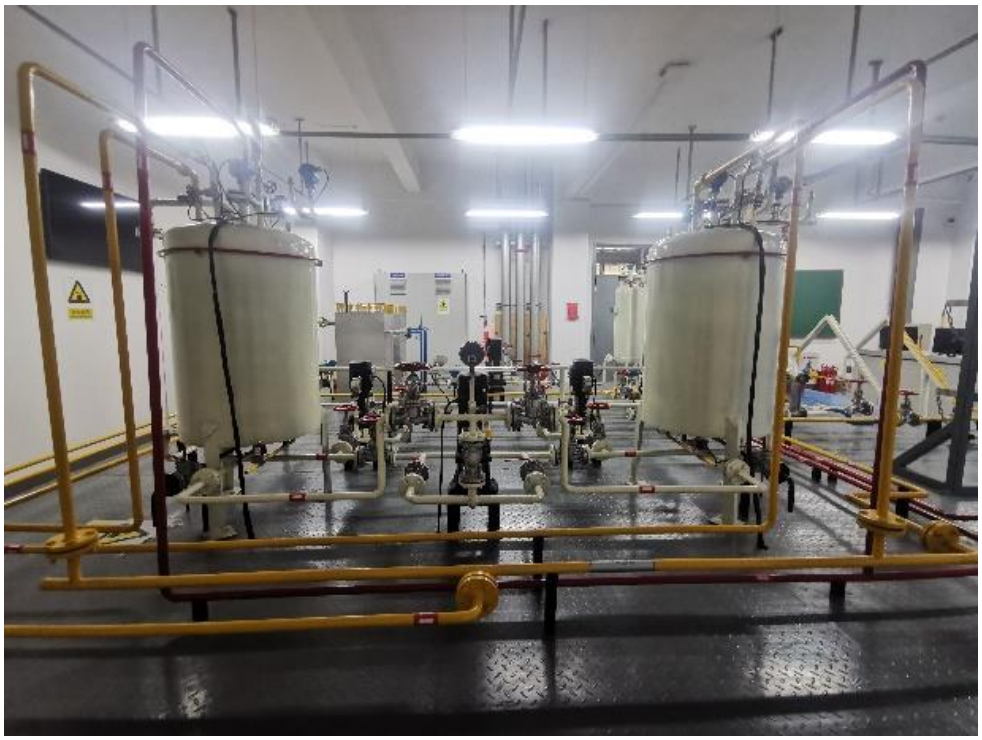


Fig. 3. Photo of the medium-scale LNG terminal

Table. 1 Experimental configurations

Experimental parameters	Value
Geometry size (mm)	6000×7300×2000
Location of release point (mm)	(4260,5170,105)
Release diameter (mm)	6
Released gas	Natural gas
Volumetric release rate (L/min)	4.7, 6.2, 7.7
Release during (s)	60
Release direction	+X
Location of sensor 1	(4180,4825,30)

Subsequently, we construct the FLACS-based natural gas release numerical modeling of the medium-scale LNG terminal, the configurations of which, e.g., the geometry size, release-relevant parameters etc., are the same as the above experimental ones. The numerical dimensions of the natural gas dispersion domain are x=7.5m, y=7.3m, and h=0.2m, respectively. Based on the prior sensitivity analysis of grid size, a 0.1 grid size is applied for the core domain, while a factor of 1.2 is used to stretch the grid from the core domain to the numerical boundary. The numerical model applies the WIND as the inflow boundary condition while the outflow boundary is considered as Euler. The initial CFLC and CFLV values are initially set to be 20 and 2, respectively. However, due to the fact that the grid nearby the release point should be refined in order to get accurate numerical results while maintaining moderate computational cost, the CFLC and CFLV values are allowed to be automatically adjusted. Furthermore, the release duration is 60s, and we sample each concentration sequence at very 0.5s interval. Fig. 4 shows the numerical modeling of the medium-scale LNG terminal. Please noting that we build the numerical model in open space without considering the same confined boundary condition with the experimental one in Fig. 3 because of the very limited influence of experimental boundary on the small-scale released plume.

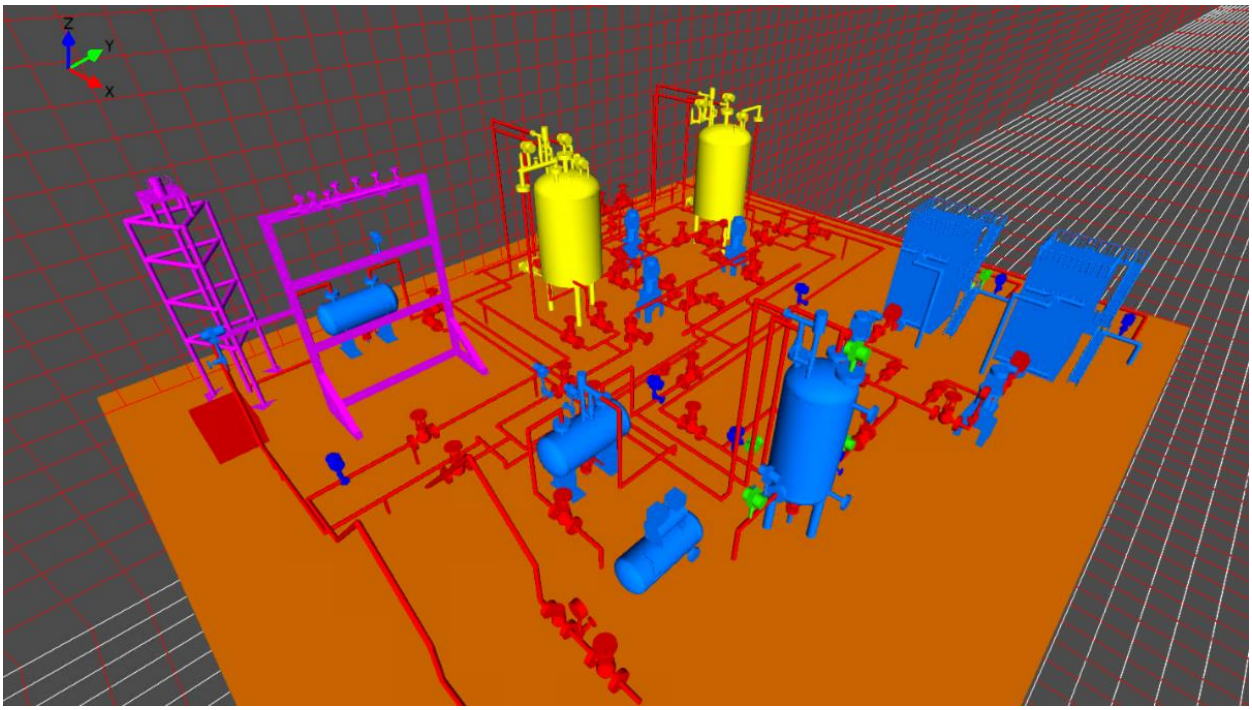
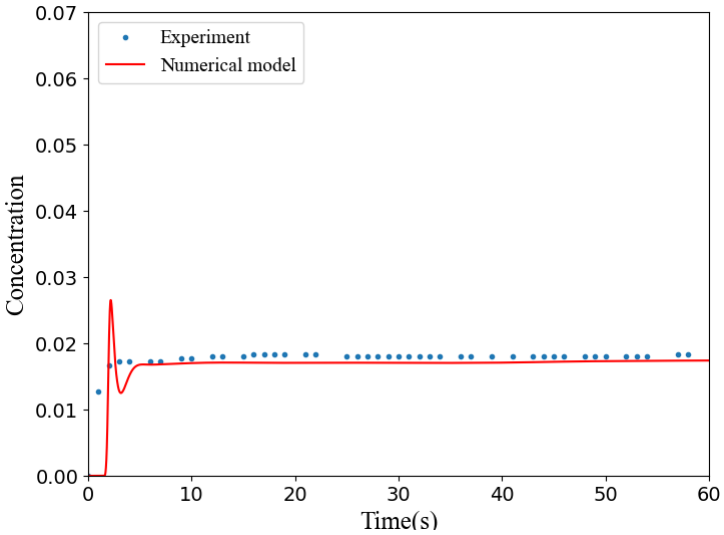
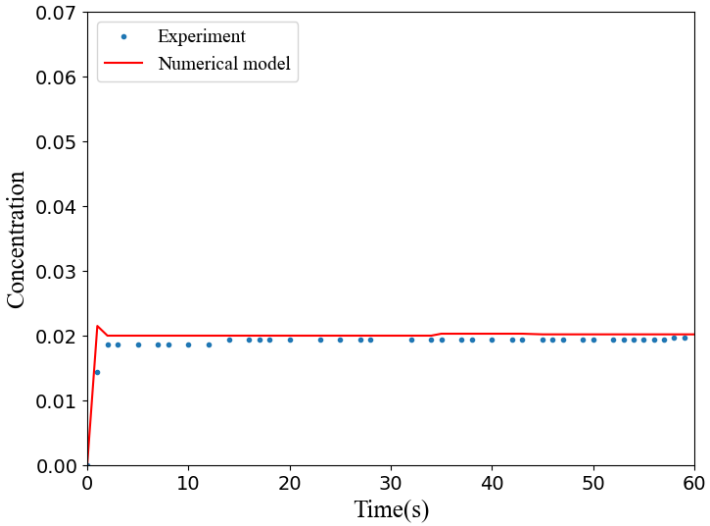


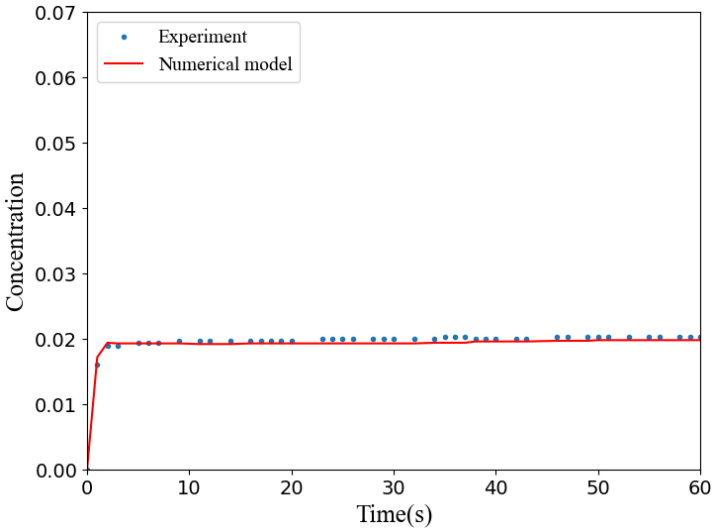
Fig. 4 Natural gas release numerical modelling of a medium-scale LNG terminal



a) 4.7 L/min



b) 6.2 L/min



c) 7.7 L/min

Fig. 5 Validation of natural gas release numerical modelling by experimental data

Based on the experimental data, we validate the constructed natural gas release and dispersion numerical model.

Fig. 5 demonstrates the comparisons of the historical concentration data under 3 various release rates, i.e., 4.7 L/min, 6.2 L/min, 7.7 L/min, between the experiment and numerical model. From Fig. 5, it can be seen that the experimental concentration data steeply increases at the initial stage. After about 4s, the concentration data becomes stable. Furthermore, with the increase of the release rate from 4.7 L/min, to 7.7 L/min, the stable concentration becomes more closed to 0.02 (m^3/m^3). Additionally, although the sudden concentration peaks at the initial stage occur under 4.7 L/min and 6.2 L/min, the historical concentration data derived by the numerical model is closed to the experimental data, which indicates the desirable accuracy of the constructed natural gas release and dispersion numerical model.

4.2. Benchmark data preprocessing

By using the validated numerical model, we totally simulate 54 scenarios of natural gas release scenarios including 6 release velocities, i.e., 4.7 L/min, 6.2 L/min, 7.7 L/min, 9.2 L/min, 10.7 L/min, 12.2 L/min, 2 wind directions, i.e., +Y and –Y directions, 6 wind speeds, i.e., 0m/s, 0.04m/s, 0.10m/s, 0.20m/s, 0.30m/s, 0.40m/s. The detailed numerical configurations can be seen in Table. 2. For each dispersion scenario, we extract 3 spatiotemporal evolution sequences, and each sequence contains the previous 10-steps spatial concentration contours (*J=10*) from 3.5s to 8.0s with 0.5s interval and the 10-steps-ahead concentration contours (*K=10*) from 8.5s to 13.0s with 0.5s interval. Then, we totally collect 162 spatiotemporal evolution sequences, which include 3240 spatial concentration contours, each of which has dimensions *n1=48* and *n2=9*. We denote $D = [X, Y]$ ($X \in \mathbb{R}^{162\times48\times9\times10}$ and $Y \in \mathbb{R}^{162\times48\times9\times10}$) as the benchmark numerical dataset where $[X_1, X_2, \dots, X_{10}]_{162}$ are the input sequences and $[Y_{11}, Y_{12}, \dots, Y_{20}]_{162}$ are the output sequences. We normalize the dataset D by dividing by the maximum concentration value of the dataset (0.07 m^3/m^3) and setting the concentration value less than 0.01 m^3/m^3 to be 0. The normalized dataset $D = [X, Y]$ is randomly divided into two sets, namely training set D_{train} (80%) and test set D_{test} (20%).

Table. 2 Numerical configurations

Parameters	Value
Release position	(4260,5170,105)
Release direction	-Y
Duration (s)	60
Release velocity (L/min)	4.7, 6.2, 7.7, 9.2, 10.7, 12.2
Wind direction	+Y, -Y
Wind speed (m/s)	0, 0.04, 0.1, 0.2, 0.3, 0.4

5. Hybrid_PG_VBSTnn development

By using the training set D_{train}, we optimize the Variational parameter φ of the proposed Hybrid_PG_VBSTnn model. Subsequently, its pre-defined parameters such as m, λ, p are determined. At last, comparisons among our proposed model and the deep learning models without considering the uncertainty inference and the physical constraint are conducted as well.

As is described in Section 3, our proposed Hybrid-PG-VBSTnn model consists of 3 parts. In Part 1, namely Convolutional encoder, 3 Time Distributed convolution layers with 128 filters, 11×11 kernel size; 64 filters, 7×7 kernel size; 64 filters, 7×7 kernel size, respectively, stride of 1, ReLU activation function, and padding of the same dimension are used. Part 2 consists of 3 ConvGRU2D layers with 32 filters, 7×7 kernel size; 32 filters, 5×5 kernel size; 32 filters, 7×7 kernel size, respectively, stride of 1, ReLU activation function, and padding of the same dimension. Furthermore, in Part 3, 3 Time Distributed deconvolution layers with 64 filters, 7×7 kernel size; 64 filters, 7×7 kernel size; 128 filters, 11×11 kernel size, respectively, stride of 1, ReLU activation function, and padding of the same dimension are first used. Then, a 3D convolution layer with 1 filter, 1×1×1 kernel size, stride of 1, Linear activation function, and padding of the same dimension is considered as the output layer. In addition, we configure the Batch Normalization layer and Bayesian dropout layer between each layer. Fig. 6 demonstrates the architecture of our proposed Hybrid-PG-VBSTnn model. Table. 3 shows the configuration of our proposed Hybrid-PG-VBSTnn model. Our model is compiled by using Tensorflow 1.14.0. All the subsequent analysis is conducted by using the computer system with the configuration of 64GB RAM, Intel (R) Core™ i9-9900K CPU, and an NVIDIA GeForce RTX 2080 Ti GPU card.

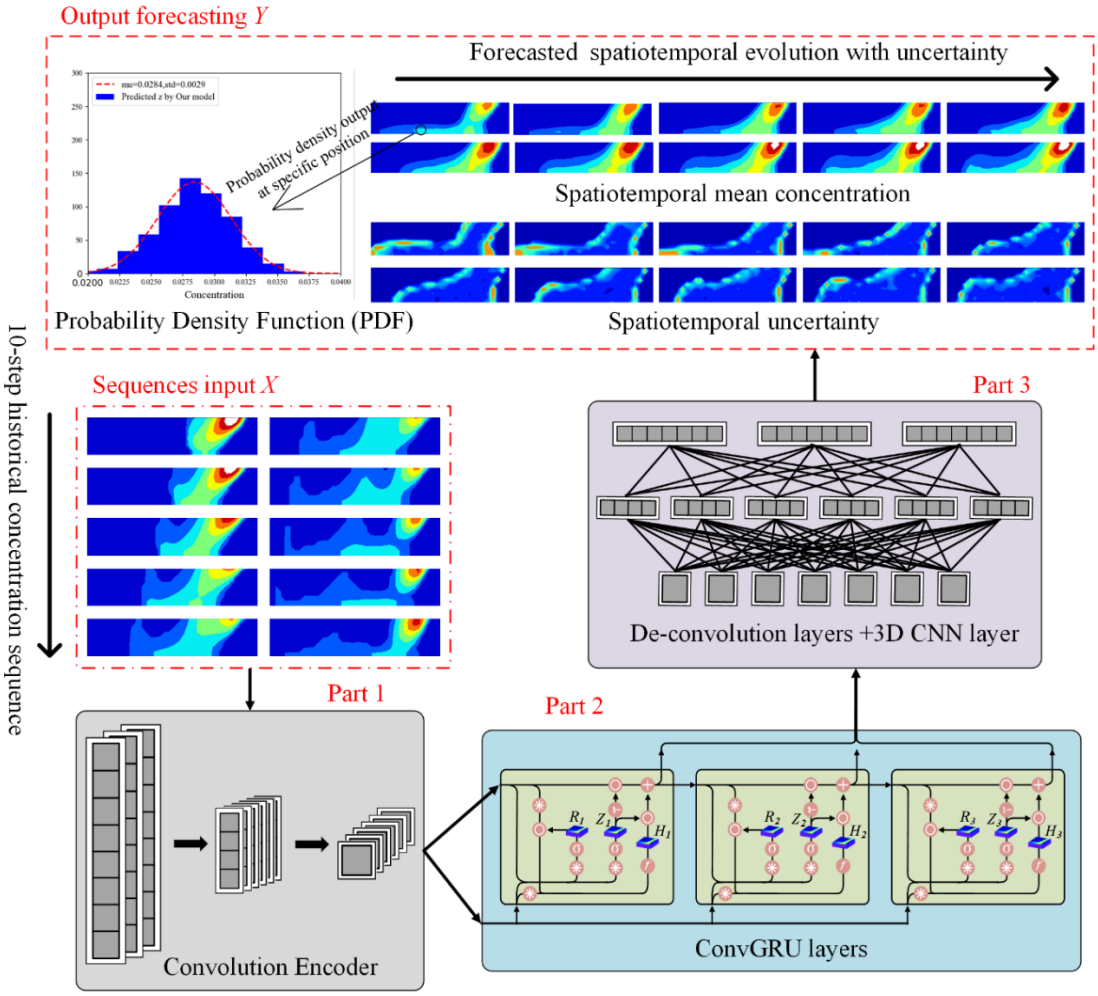


Fig. 6 Architecture of Hybrid-Physics Guided-Variational Bayesian Spatial-Temporal neural network (Hybrid-PG-VBSTnn)

Table. 3 Configuration of the Hybrid-PG-VBSTnn model

Part 1: Convolution encoder	Part 2: ConvGRU2D	Part 3: Convolution_forecasting
11×11 128 Time Distributed Conv2D. ↓, ReLU	7×7 32 ConvGRU2D↓, ReLU	7×7 64 Time Distributed DeConv2D ↑, ReLU
7×7 64 Time Distributed Conv2D. ↓, ReLU	5×5 32 ConvGRU2D↓, ReLU	7×7 64 Time Distributed DeConv2D ↑, ReLU
7×7 64 Time Distributed Conv2D. ↓, ReLU	7×7 32 ConvGRU2D ↑, ReLU	11×11 128 Time Distributed DeConv2D ↑, ReLU
		1×1×1 1 Conv3D ↑, Linear

5.1. Monte Carlo sampling number m

By using the already developed model given the pre-defined dropout probability p=0.1, and penalty factor λ=0.1, all the sequences from 3.5s to 8s of the test set D_{test} are used as the input X to forecast the sequences from 8.5s to 13s. We take the forecasted concentration at the position (x=34, y=2) under t=8.5s of the 2nd sequence X as the example to analyze the effect of the MC sampling number m on the convergence of the spatiotemporal forecasting.

Fig. 7 demonstrates the benchmark spatial concentration contour at time t=8.5s of the 2nd sequence from the test set D_{test}, where the selected point is marked as well. From it, it can be seen the benchmark concentration value at the position (34, 2) equals to 0. 0289.

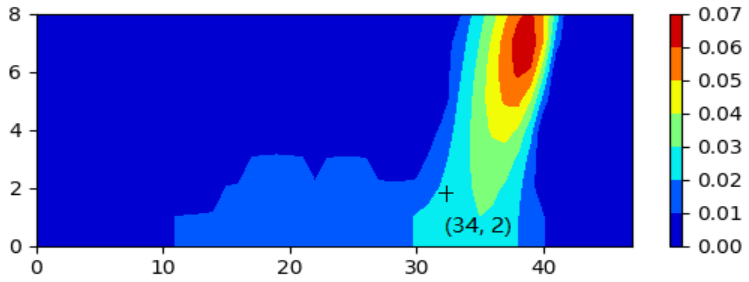
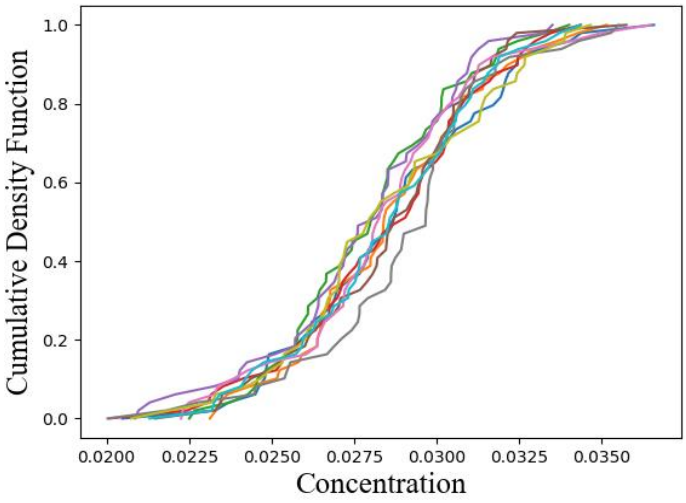
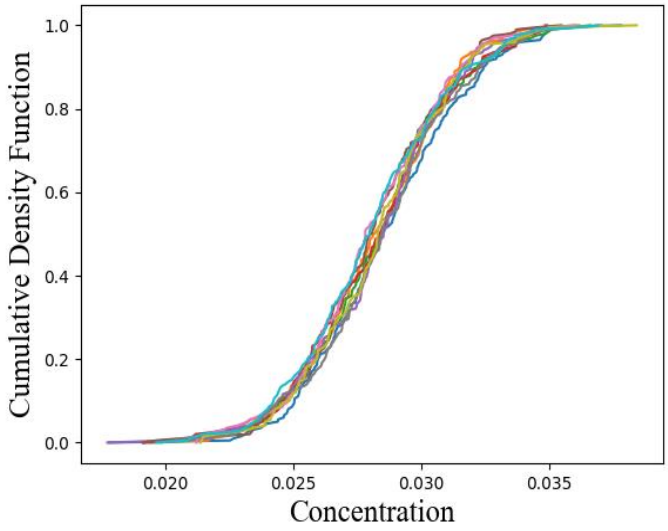


Fig. 7 Benchmark spatial concentration contour at time t=8.5s of the 2nd sequence in the testing dataset. Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

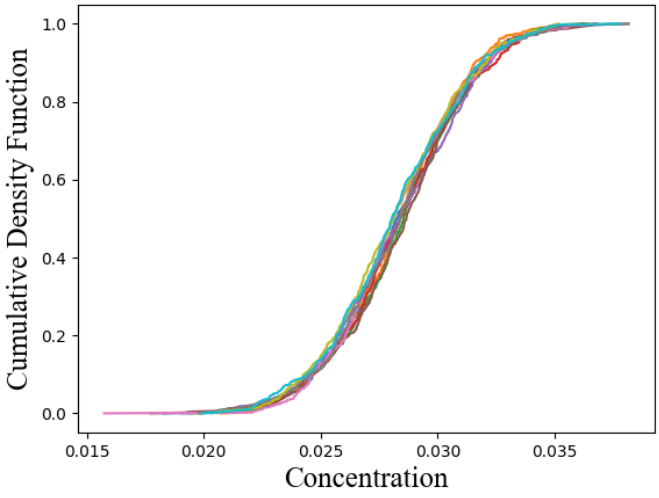
We generate10 groups of concentration values by randomly sampling the weights 10 times from the Variational posterior distribution $q_\varphi(w)$. Noting that for each group, we obtain m samples. Fig. 8 demonstrates the Cumulative Density Function (CDF) curves of 10 groups of concentration values under m=*50*, 100 and 500 samples at the position (34, 2). Noting that each CDF curve corresponds to each group of concentration values. As can be seen, the interval of the uncertainty in forecasted concentration when CDF value=0.5 is about 0.003 under 50 samples. However, as m increases from 50 to 100, the interval is significantly reduced to be less than 0.0001.



a) 50 samples



b) 100 samples



c) 500 samples

Fig. 8 Cumulative Density Function (CDF) curves of 10 groups of forecasted concentration values under *m =50*, 100, and 500 samples at the position (x=34, y=2). Note that each CDF curve corresponds to each group of the forecasted values.

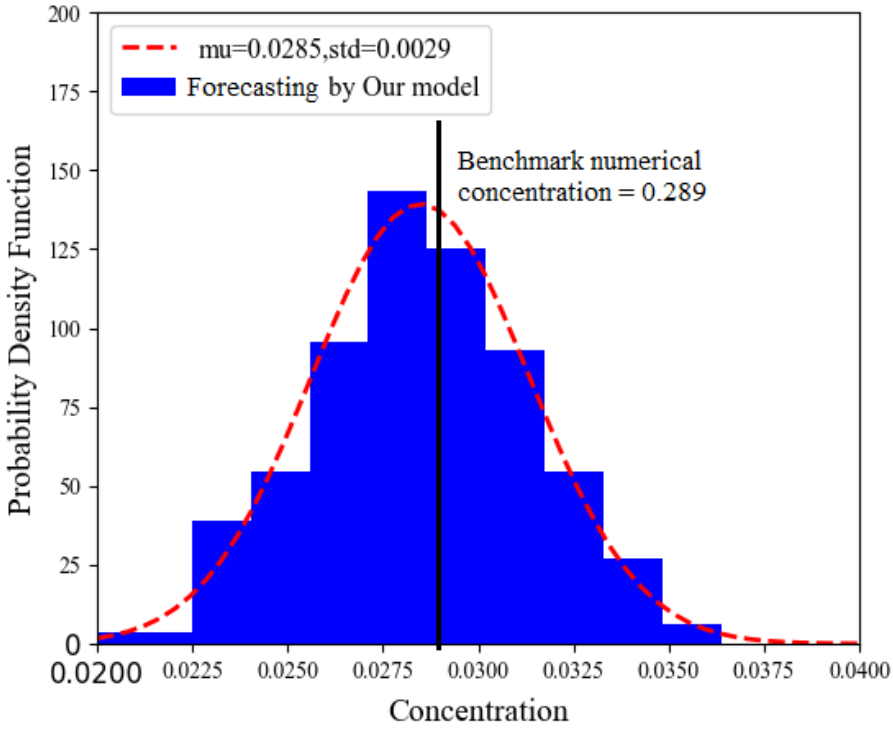
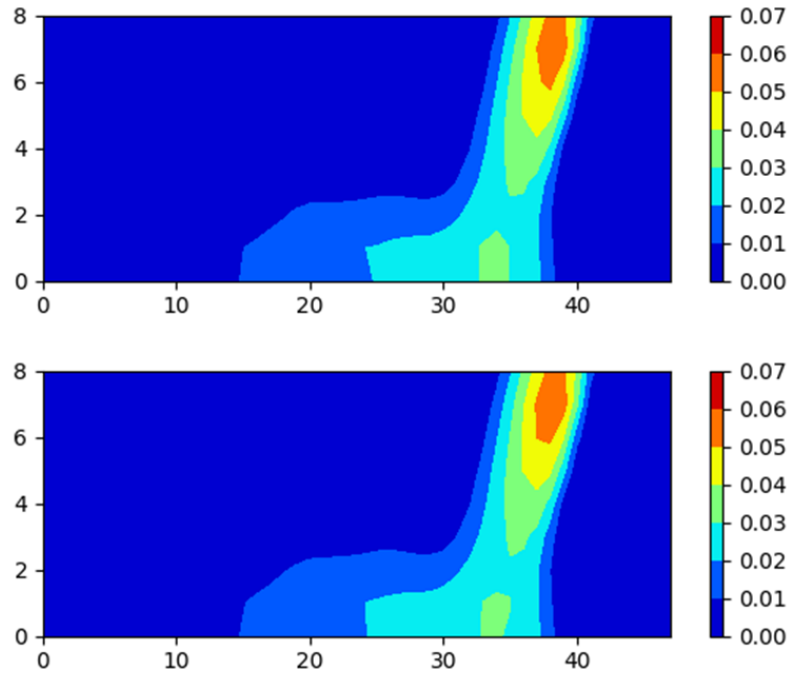


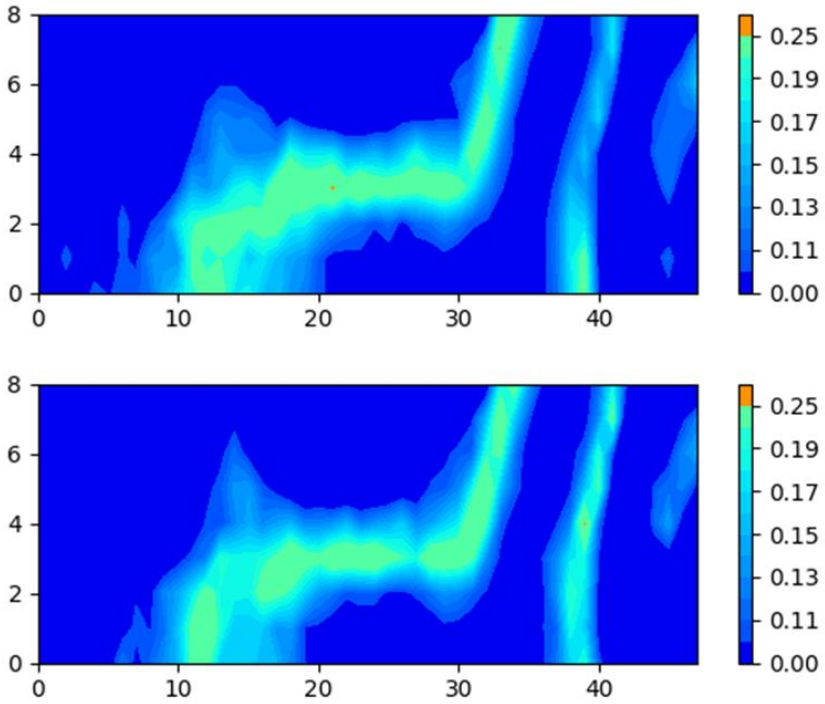
Fig. 9 Probability Density Function (PDF) curve of the forecasted concentration values of one specific group under 100 samples at the position (x=34, y=2).

Although the interval under 500 samples is less than that under 100 samples, m =500 could considerably increase the computational cost and thereby affect the real-time inference speed of the proposed model. Thus, we select m =100 as the trade-off between computational cost and uncertainty induced by various samples on the forecasted spatial contours by the proposed model. Furthermore, Fig. 9 is the probability density distribution of forecasted concentration values of one specific group under m=100 samples at the position (x=34, y=2), showing that the samples follow a Gaussian distribution with a mean concentration value=0.0285 and standard deviation=0.0029. And the difference between the forecasted mean concentration, i.e., 0.0285, and the benchmark concentration, i.e., 0.0289 at the position (x=34, y=2) is just 0.0005.

In addition to the convergence analysis at one specific position, we demonstrate the convergence of the forecasted concentration and normalized uncertainty contours under 100 samples. Fig. 10 shows the comparison of the forecasted concentration contours under two groups of 100 samplings at t=8.5s of the 2nd sequence from the test dataset. From it, it can be seen both forecasted concentration contours under two groups are almost the same. In terms of the normalized uncertainty contour, it can be seen large uncertainty values at the boundary between the areas with and without released gas. Also both areas corresponding to such large uncertainty values under two groups are almost the same as well. This indicates the feasibility to determine MC sampling number m=100 as the optimal to ensure the convergence of our proposed model.



a) Comparison of the forecasted concentration contours



b) Comparison of the forecasted concentration normalized uncertainty contours

Fig. 10 Comparison of the forecasted results under two groups of 100 samplings at t=8.5s of the 2nd ıce from the test dataset. Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

5.2. Penalty factor λ

In order to analyze the effect of the penalty factor λ on the forecasted concentration and its normalized uncertainty contour, we take the forecasted results at t=8.5s of the 2nd sequence X under various λ values, MC sampling number m=100 and pre-defined dropout probability p=0.1 as the example to demonstrate.

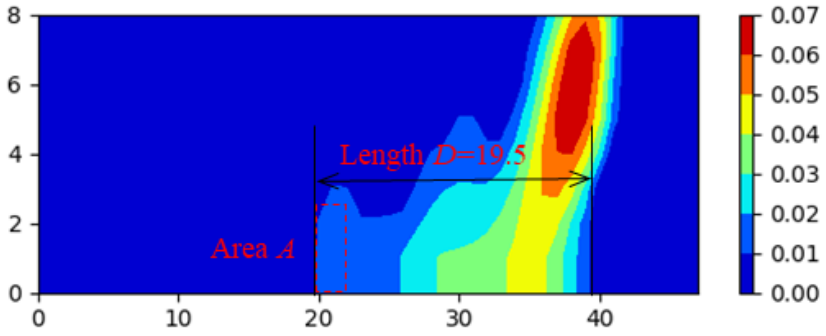


Fig. 11 Benchmark spatial concentration contour at time t=10.5s of the 2nd sequence of the testing dataset. Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis. And D=19.5 means the grid number is 19.5.

Fig. 11 demonstrates the benchmark spatial concentration contour at time t=10.5s of the 1st sequence of the testing dataset. As can be seen, the length D between the left and right boundaries of the concentration contour is 19.5. Also, one can obviously see the released gas plume exists in the area A. Fig. 12 demonstrates the forecasted spatial concentration contours by the models developed under various λ values from 0.1 to 0.4. As can be seen, all the developed models under various λ values forecast the almost same spatial concentration contours closed to the benchmark concentration contour in Fig. 11 even though some differences still exist, e.g., no released gas plume can be seen in the responding area A under various λ values. In addition, although smaller than the benchmark length D marked in Fig. 11, the lengths between the left and right boundaries under various λ are very closed to each other, which indicates the penalty factor λ has very limited influence on the forecasted concentration contour.

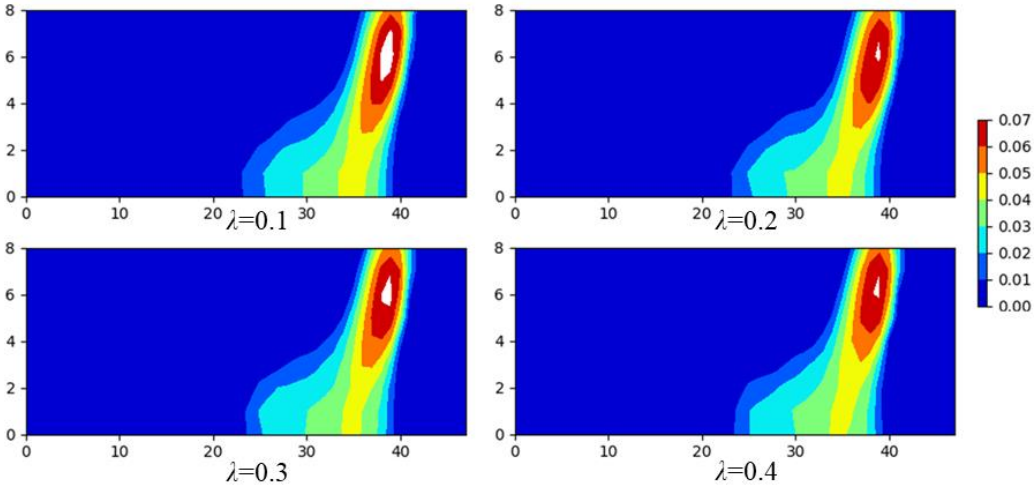


Fig. 12 Spatial concentration contours forecasted under various penalty factor values at time t=10.5s of the 2nd sequence of the testing dataset. Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

Fig. 13 demonstrates the normalized uncertainty contours of spatial concentration forecasted by the models developed under various λ values from 0.1 to 0.4. From it, it can be seen that the length between the left and

right uncertainty boundaries firstly decreases with increasing the penalty factor λ from 0.1 to 0.2, and then keep steady after λ=0.2. In addition, the length under λ=0.1 equals 19.5, which is the same with the benchmark length D marked in Fig. 11, while the lengths under λ=0.2, 0.3 and 0.4, respectively, are relatively smaller. This indicates the more accuracy of the proposed model under λ= 0.1 in terms of the normalized uncertainty contour forecasting.

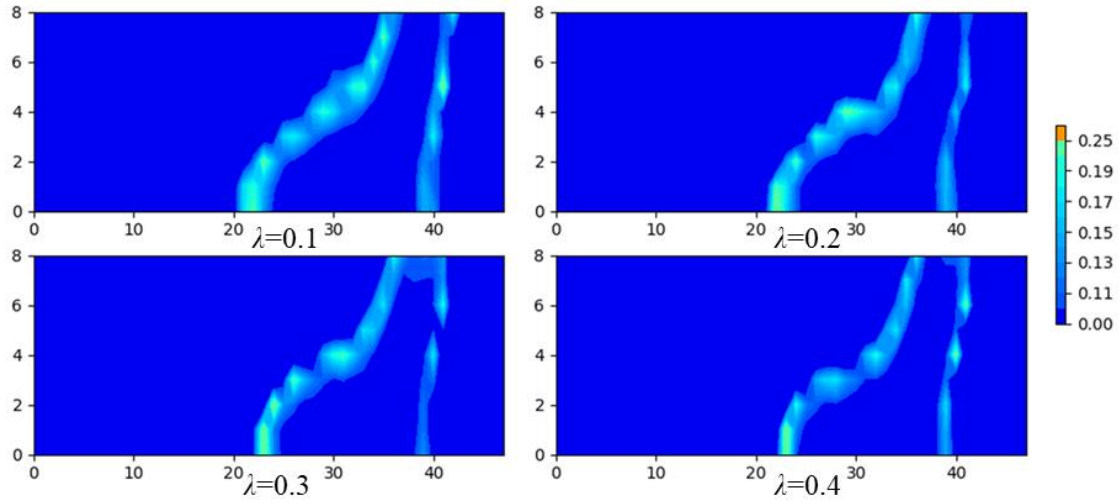


Fig. 13 Normalized uncertainty contours of spatial concentration forecasted under various penalty factor values at time t=10.5s of the 2nd sequence of the testing dataset. 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

Table. 4 demonstrates *MSE* between the forecasted concentration and benchmark concentration under various penalty factor values λ at time t=10.5s. From it, it can be seen that the *MSE* value decreases with the increase of the penalty factor λ. This indicates the Hybrid_PG_VBSTnn model developed under λ= 0.1 exhibits a relatively high accuracy in terms of spatial concentration forecasting compared to the models developed under the additional penalty factor values. Furthermore, considering that our model developed under λ= 0.1 more accurately forecast the normalized uncertainty contour as well, we thereby determine the penalty factor λ= 0.1 as the optimal for the gas release and dispersion spatiotemporal evolution forecasting.

Table. 4 Mean square error (MSE) between the forecasted concentration and benchmark concentration under various penalty factor λ values at time t=10.5s

Penalty factor λ	Mean square error (*MSE*)
0.1	0.68E-05
0.2	0.75E-05
0.3	0.87E-05
0.4	0.99E-05

5.3 Drop probability p

In order to analyze the effect of the drop probability p on the forecasted concentration and its normalized uncertainty contour, we further take the forecasted results at t=10.5s of the 2nd sequence X under various λ values, MC sampling number m=100 and pre-defined penalty factor value λ=0.1 as the example to demonstrate.

Fig. 14 shows the forecasted spatial concentration contours under various p values. From it, it can be seen the forecasted concentration contours under 4 values of p are almost the same. Fig. 15 demonstrates the normalized uncertainty contours of spatial concentration forecasted under 4 values of p. As can be seen, the length between the left and right uncertainty boundaries increases with increasing the p value from 0.1 to 0.4. Furthermore, the corresponding length under p=0.1 is closed to the benchmark length D marked in Fig. 11, while the lengths under λ=0.2, 0.3 and 0.4, respectively are larger. This indicates the more accuracy of the Hybrid_PG_VBSTnn model developed under p = 0.1 in terms of the normalized uncertainty contour forecasting.

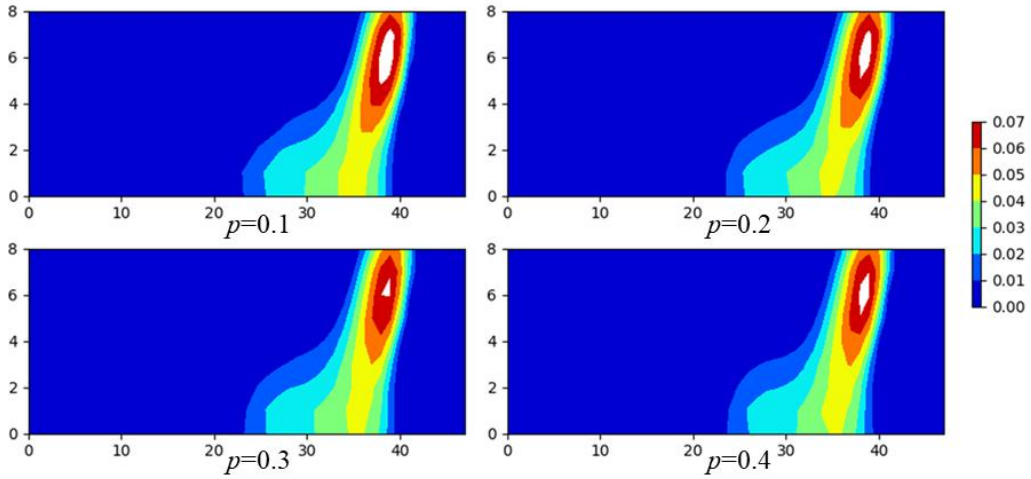


Fig. 14 Spatial concentration contours forecasted under various drop probability p values at time t=10.5s of the 2nd sequence of the testing dataset. Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

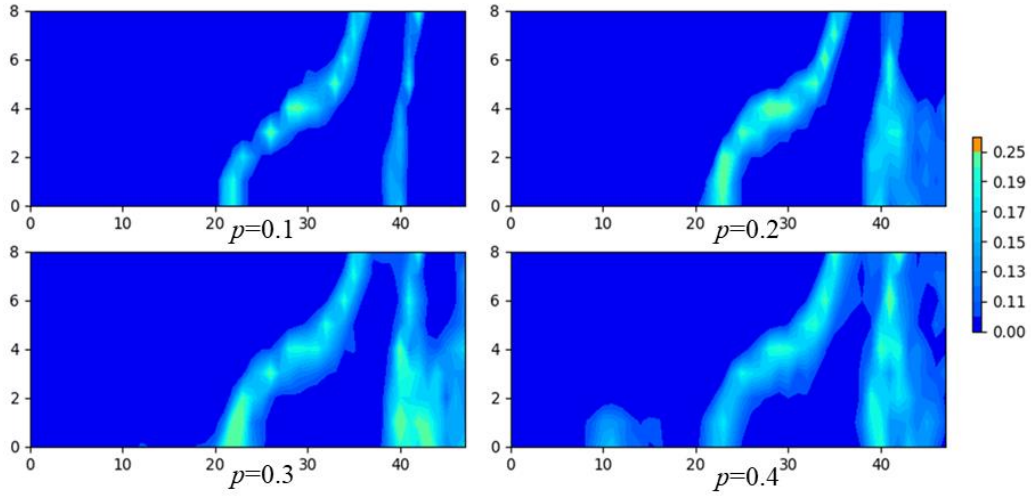


Fig. 15 Normalized uncertainty contours of spatial concentration forecasted various drop probability values at time t=10.5s of the 2nd sequence of the testing dataset. Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

Table. 5 demonstrates *MSE* between the forecasted concentration and benchmark concentration under various p values at time t=10.5s. From it, it can be seen that the *MSE* value decreases with increasing p value. This

indicates the Hybrid_PG_VBSTnn model developed under p= 0.1 exhibits the relatively high accuracy in terms of the forecast of concentration, compared to the models developed under other p values. Furthermore, considering that the proposed model developed under p = 0.1 forecast the more accurate uncertainty contours as well, we thereby determine the drop probability p = 0.1 as the optimal for the gas release and dispersion spatiotemporal evolution forecasting.

Table. 5 Mean square error between the forecasted concentration and benchmark concentration under various penalty factor values λ at time t=10.5s

Drop probability p	Mean square error (*MSE*)
0.1	0.68E-05
0.2	0.77E-05
0.3	0.78E-05
0.4	1.05E-05

5.4 Validation and comparison

This part validates the accuracy and real-time capability of our developed Hybrid_PG_VBSTnn model under Monte Carlo sampling number m=100, penalty factor λ= 0.1 and drop probability p=0.1 by using the benchmark numerical testing dataset as well as the experimental time-series concentration data. Also, comparison among our Hybrid_PG_VBSTnn model, the point-estimation deep learning model, and the hybrid model without considering the physical constraint term are conducted in order to demonstrate the advantages of our developed model.

Fig. 16 demonstrates the forecasted concentration spatiotemporal evolution with uncertainty by our Hybrid_PG_VBSTnn model. Please note that the benchmark 10-steps-ahead concentration contours from t=8.5s to t=13.0s of the 1st input sequence are from the benchmark numerical testing dataset. From it, it can be seen that the length D between the left and right boundaries of the forecasted concentration contours firstly decreases from 8.5s to 10.5s and then increases from 11.0s to 13.0s. Such forecasted evolution trend is the same as that of the benchmark contours. In addition, it can be seen that our model could forecast the spatiotemporal concentration very well at the area occupied by the released gas with a relatively larger concentration. However, at the boundary identifying the areas with and without the released gas, the only forecasted concentration contours could not provide the correct information. For example, the forecasted D equals to 18 smaller than the benchmark D=22 at time t=8.5s. Thanks to the Variational Bayesian inference, the normalized uncertainty contours by our model could overcome such a disadvantage. One side is that the length D of the normalized uncertainty contour is more closed to the benchmark length D compared to that of the forecasted concentration contour.

For example, the D of the normalized uncertainty contour equals to 21 very closed to the benchmark D = 22 at time t=8.5s while the D of forecasted concentration contour equals to 18. In this case, one could thereby use

the large normalized uncertainty contour as the boundary to identify the areas with and without the released gas along with analyzing the forecasted concentration contour at the same time. For example, from Fig. 16, one can obviously see the large normalized uncertainty contour, inside which the forecasted concentration values are larger than 0.01 indicating released gas exists. Outside such large normalized uncertainty contour, the concentration values are forecasted much closed to 0 indicating no released gas exists. Due to this fact, our Hybrid_PG_VBSTnn model could provide safer and more accurate spatiotemporal evolution information at the boundary for the further decision making process compared to the point-estimation deep learning model which does not consider the uncertainty.

For example, from Fig. 16, it can be seen that large normalized uncertainty values at the Area A at time t=13.0s, which indicates the released gas exists at the corresponding Area. This is same with that the released gas exists at the Area A of the benchmark contour at time t=13.0s. However, if not determining such uncertainty information, e.g., by the point-estimation deep learning model, one could see no released gas exists only by analyzing the forecasted concentration contour, which is incorrect and very danger for the further decision-making process, because such gas has the potential to cause the fire and explosion accidents once being ignited.

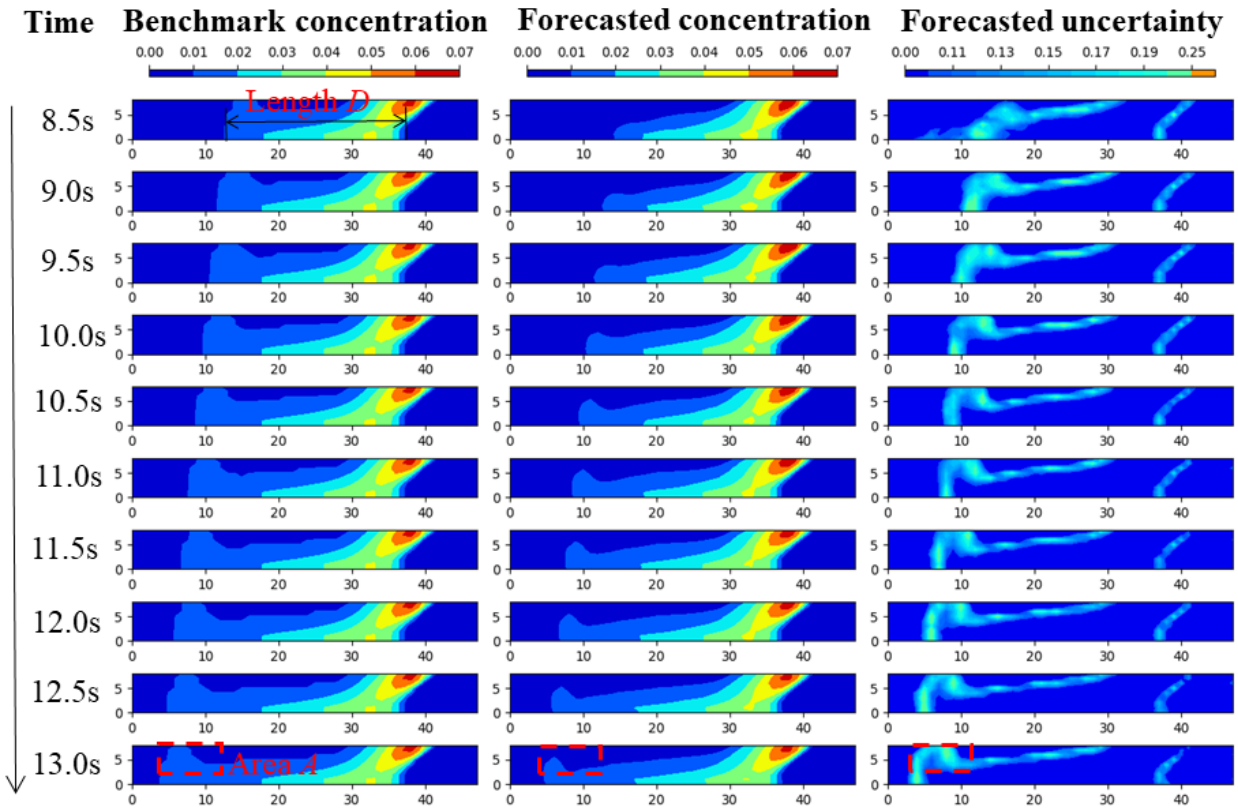


Fig. 16 Forecasted concentration spatiotemporal evolution with uncertainty by the hybrid model without considering physical constraint. (The benchmark 10-times-ahead concentration contours from t=8.5s to t=13.0s of the 1st input sequence are taken from the benchmark numerical testing dataset). Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

Fig. 17 demonstrates the forecasted spatiotemporal concentration evolution with normalized uncertainty by

the hybrid model without considering the physical constraint term. Please noting that the benchmark 10-steps-ahead concentration contours from t=8.5s to t=13.0s of the 1st input sequence are taken from the benchmark numerical testing dataset as well. As can be seen, the model exhibits competitive accuracy in terms of the spatiotemporal evolution forecasting at the area occupied by the high concentration gas. However, at the boundary, such a model exhibits poor performance. For example, one can obviously see the released gas exits at Area C from the benchmark concentration contour at 9.0s, while the forecasted concentration is 0 indicating no released gas. Furthermore, at Area B, the model forecasts the negative concentration, which does not follow the physical laws and thereby results in the larger normalized uncertainty values. One could not use the forecasted normalized uncertainty to identify the areas with and without the released gas without considering the physical constraint term, which reduces the model's accuracy at the boundary.

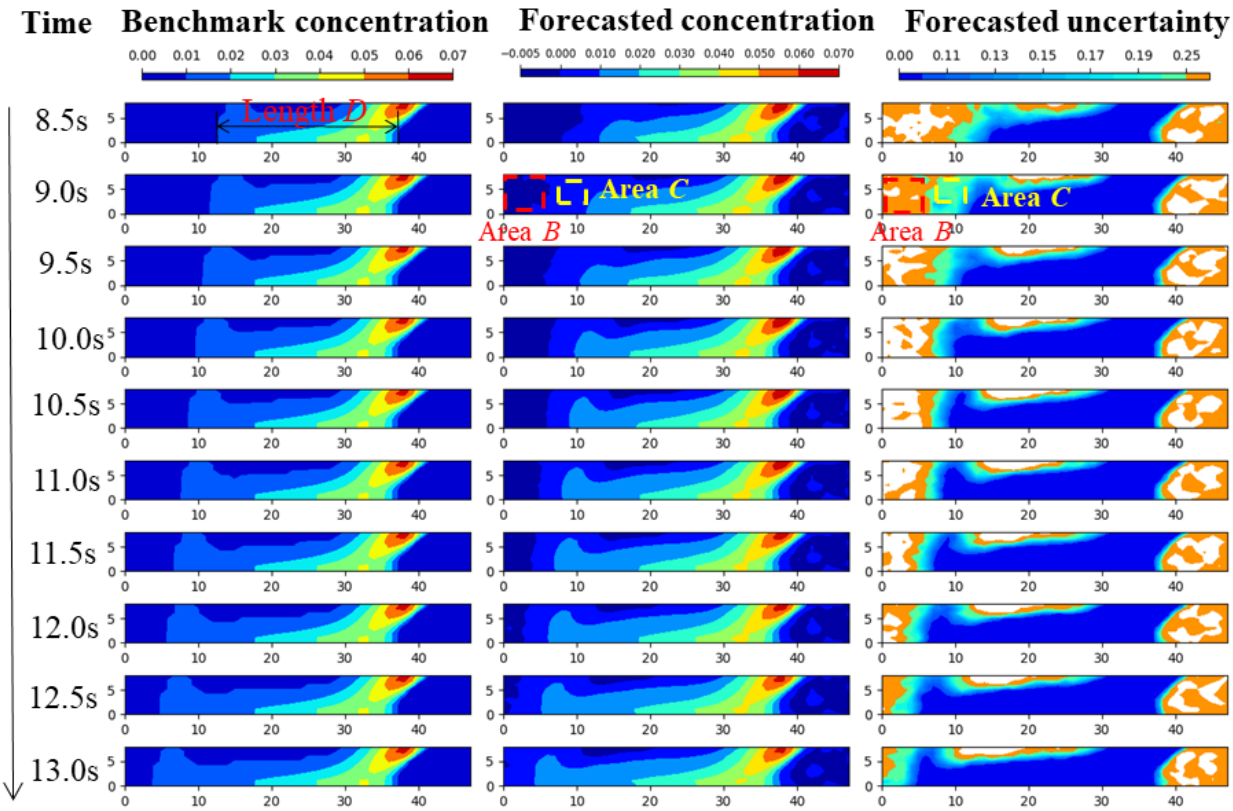


Fig. 17 Forecasted concentration spatiotemporal evolution with uncertainty by the hybrid model without the physical constraint. (The benchmark 10-steps-ahead concentration contours from t=8.5s to t=13.0s of the 1st input sequence are taken from the benchmark numerical testing dataset). Noting that 40 means the 40th grid in x axis and 8 indicates the 8th grid in y axis.

Table. 6 demonstrates the comparison among our developed hybrid model, the hybrid model without physical constraint, point estimation model in terms of the inference accuracy and speed. We calculate the correlation of determination R^2 between the forecasted concentration and benchmark concentration of the testing dataset as the criterion for the comparison of inference accuracy. And we determine the average time interval during which one spatial concentration could be outputted as the criterion for the comparison of inference speed. From it, it can be seen that R^2 value of our hybrid model equals 0.978, close to that determined by other models. This

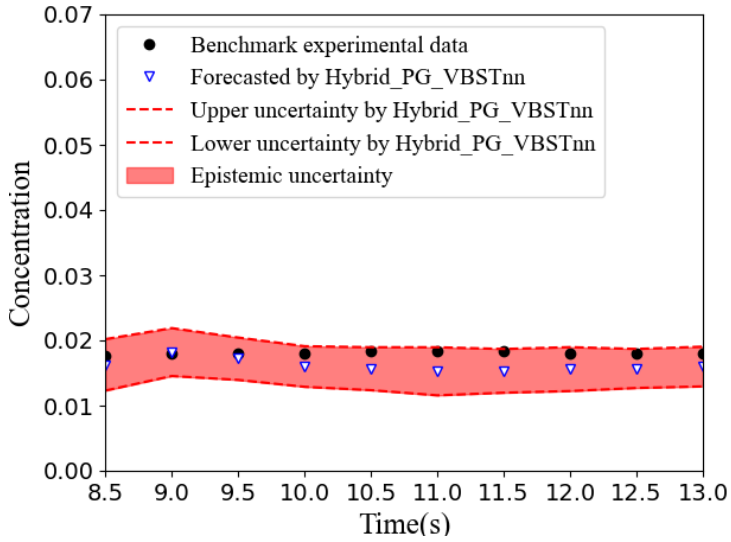
indicates hybrid model exhibits high accuracy at the area occupied by the released gas with high concentration.

Nevertheless, the additional normalized uncertainty contour, as well as the physical constraint, ensure the more accuracy of our hybrid model in terms of the spatiotemporal forecasting at the boundary compared to the other models. Also, it can be seen that the inference time of our hybrid model equals to 0.5s closed to the time interval between two benchmark concentration contours, which indicates the real-time capability of our hybrid model. Furthermore, due to the MC sampling process, our hybrid model has the relatively slower inference speed compared to that by point estimation model. Considering the considerable accuracy of our developed model at the boundary, it is acceptable to harm very limited inference speed while still ensuring the real-time capability. In addition, the difference of inference time between our hybrid model and the model without physical constraint term is very limited indicating the additional physical constraint will not increase the computational burden. At last, all the deep learning models exhibit the much more efficient inference speed compared to the CFD model.

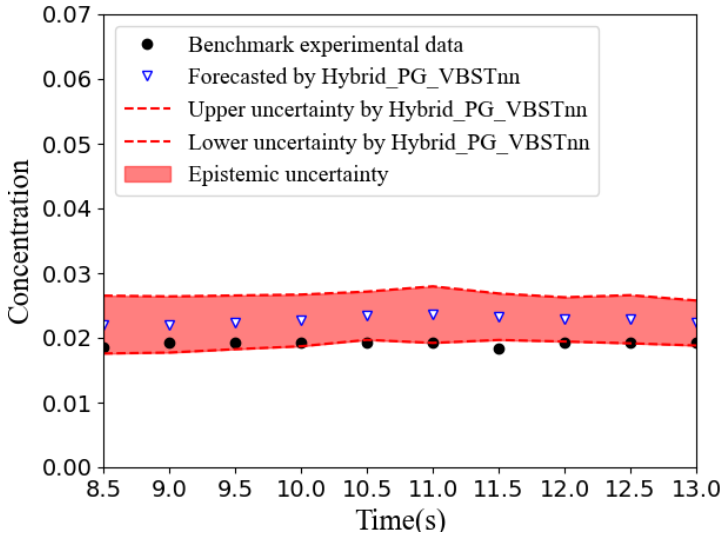
Table. 6 Comparison among our hybrid model, the model without physical constraint, point estimation model in terms of inference accuracy and speed

Model	R^2	Inference time
Our hybrid model	0.988	0.5 s
Model without physical constraint	0.985	0.5 s
Point estimation model	0.984	0.2 s
CFD model	N/A	1 hr

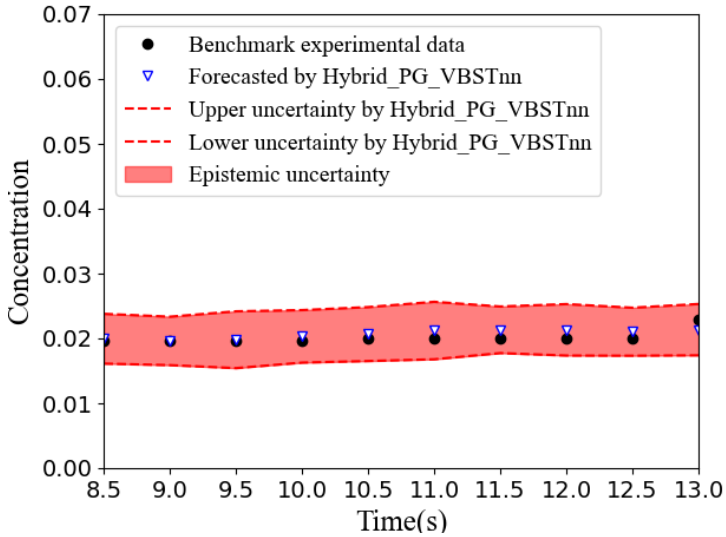
Additionally, by using the previous 10-steps-ahead numerical concentration contours (from t=3.5s to t=8s) under the same configuration with the experiment in Table. 1 as the input, the concentration values of 10-steps-ahead with uncertainty from t=8.5s to t=13s at the same position with the experimental sensor in Fig. 2 could be determined. Fig. 18 demonstrates the forecasted concentration values of 10-time-step ahead with uncertainty under various release rates, i.e., 4.7 L/min, 6.2 L/min. and 7.7 L/min by using our Hybrid_PG_VBSTnn model. Please note that the y ordinates of all input sequences are the same with that of the experimental sensor in Fig. 2. Also, such input sequences are not from the above mentioned training and testing dataset. As can be seen, the forecasted concentration values under 4.7 L/min and 7.7 L/min are very close to the benchmark experimental data. Despite relatively large differences between the forecasted values and experimental ones under 6.2 L/min, all the forecasted values except the concentration at t=11.5s are located between the upper and lower uncertainty curves thanks to the Variational Bayesian inference of our Hybrid_PG_VBSTnn model. This also indicates our model could forecast the more robust time-series concentration evolution compared to the 'over-confident' point estimation model which could not output the additional uncertainty information.



a) 4.7 L/min

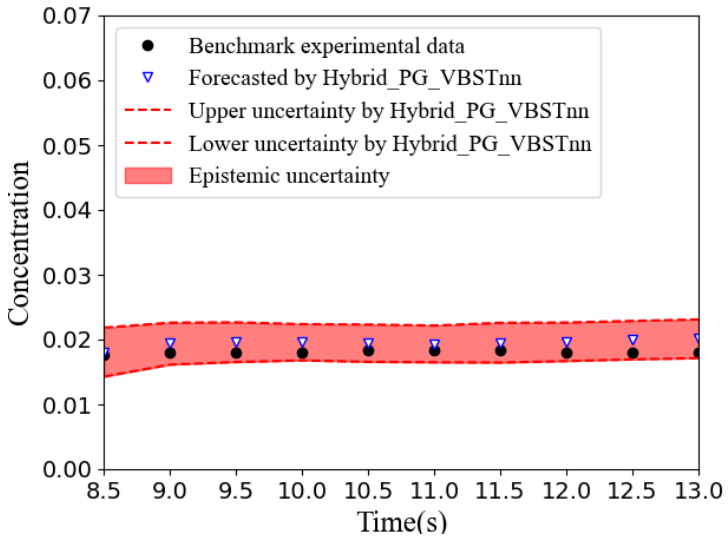


b) 6.2 L/min

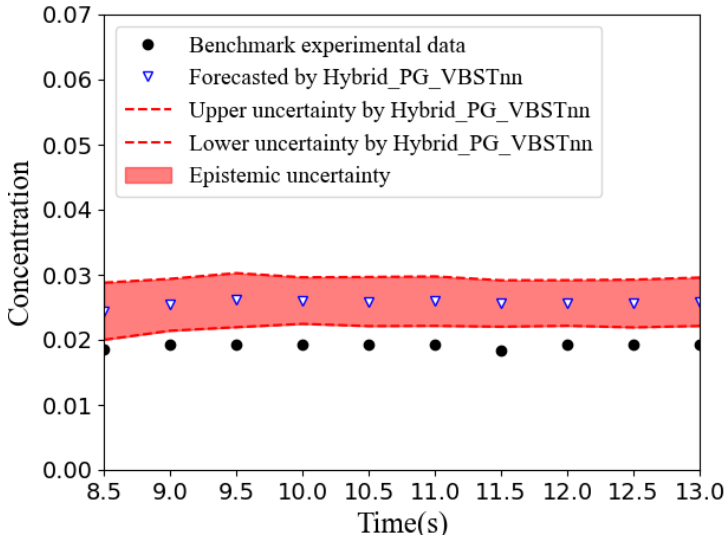


c) 7.7 L/min

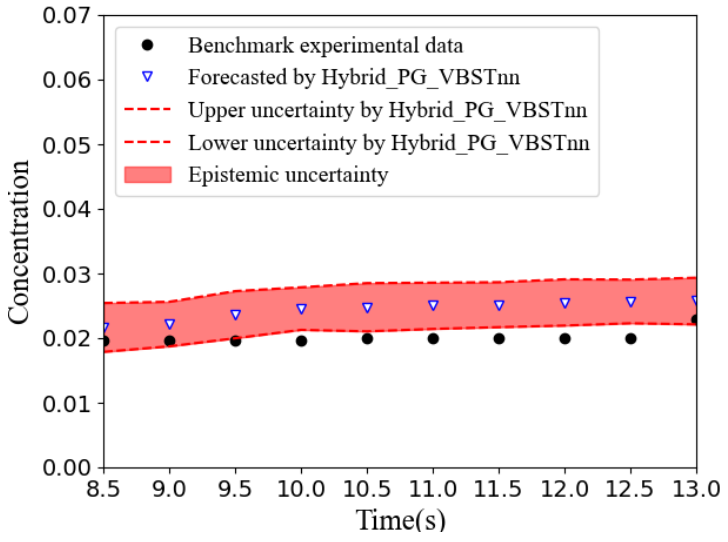
Fig. 18 Concentration values of 10-time-step ahead with uncertainty from t=8.5s to t=13s by using our Hybrid_PG_VBSTnn model



a) 4.7 L/min



b) 6.2 L/min



c) 7.7 L/min

Fig. 19 Concentration values of 10-time-step ahead with uncertainty from t=8.5s to t=13s by the deep learning model without physical constraint

Fig. 19 shows the forecasted concentration values of 10-steps-ahead with uncertainty under various release rates, i.e., 4.7 L/min, 6.2 L/min, and 7.7 L/min by the deep learning model without physical constraint term. From Fig. 19a), it can be seen that forecasted concentration values are closed to the benchmark experimental data under release rate of 4.7 L/min. However, as can be seen in both Fig. 19b) and Fig. 19c), most of the forecasted concentration values are not located between the upper and lower uncertainty curves, which indicates the relatively poorer accuracy of the model without the physical constraint term compared to our Hybrid_PG_VBSTnn model.

Overall, from the above analysis, our developed Hybrid_PG_VBSTnn model exhibits the competitive accuracy in terms of the area with high concentration released gas. The additionally normalized uncertainty contour, as well as the physical constraint term, improve the accuracy of our hybrid model to identify the boundary between the areas with and without the released gas. Also, our hybrid model could determine the uncertainty interval in terms of the time-series concentration evolution forecasting at one specific position, which ensures the robustness of our hybrid model. All the forecasted atmospheric carbon concentration and uncertainty results provide more accurate and comprehensive information for the real-time atmospheric carbon mitigation decision-making process regarding the natural gas release from oil and gas facility compared to state-of-the-art deep learning models.

6. Conclusions

This study proposed the advanced hybrid deep learning model by integrating variation inference and physical constraint with deep learning backbone to more accurately and reliably forecast spatiotemporal concentration evolution of natural gas release at plume area with high concentration as well as plume boundary compared to the point-estimation deep learning model. In addition, Optimal pre-defined parameters such as Monte Carlo sampling number m=100, penalty factor λ=0.1, and drop probability p=0.1 were determined to ensure the accuracy with R^2=0.988 as well as the real-time capability with a tiny delay time of 0.5s.

Overall, this proposed model is a reliable technique to enable real-time mitigation of the negative impact of natural gas release from oil and gas facilities and ensure the cleaner production of natural gas. In addition, our proposed integration tragedy provides new alternatives to enhance the spatiotemporal forecasting performance of state-of-the-art deep learning model for other cleaner production applications. Further work is also expected to provide the interpretability of spatiotemporal concentration evolution forecasting by fully coupling deep learning algorithm with Navier-Stokes equations.

Acknowledgment

This study was supported by National Key R&D Program of China [grant number 2021YFB4000901-03]. National Natural Science Foundation of China (Project No.: 52101341). Natural Science Foundation of Shandong Province (Project No.: ZR2020KF018). China Postdoctoral Science Foundation Funded Project

(Project No.: 2019M662469). Qingdao Science and Technology Plan (Project No.: 203412nsh). Key Project of Natural Science Foundation of Shandong Province (Project No.: ZR2020KF018). The authors would like to acknowledge partially support of the Hong Kong Research Grants Council (T22-505/19-N).

References

[1] C. Gürsan, V.J.R. de Gooyert, S.E. Reviews, The systemic impact of a transition fuel: Does natural gas help or hinder the energy transition?, Renewable and Sustainable Energy Reviews, 138 (2021) 110552.

[2] C. Liu, T. Lao, W.-Z. Wu, W. Xie, H. Zhu, An optimized nonlinear grey Bernoulli prediction model and its application in natural gas production, Expert Systems with Applications, 194 (2022) 116448.

[3] J. Zhang, H. Meerman, R. Benders, A. Faaij, Potential role of natural gas infrastructure in China to supply low-carbon gases during 2020–2050, Applied Energy, 306 (2022) 117989.

[4] A.S. Anđelković, D. Bajatović, Integration of weather forecast and artificial intelligence for a short-term city-scale natural gas consumption prediction, Journal of Cleaner Production, 266 (2020) 122096.

[5] D. Bajatović, A.S. Anđelković, I. Ćosić, R. Maksimović, Application of predictive models for natural gas needs-current state and future trends review, Tehnical Gazette, 27 (2020) 648-655.

[6] C. Liu, W.-Z. Wu, W. Xie, T. Zhang, J. Zhang, Forecasting natural gas consumption of China by using a novel fractional grey model with time power term, Energy Reports, 7 (2021) 788-797.

[7] Z.D. Weller, S.P. Hamburg, J.C. von Fischer, A national estimate of methane leakage from pipeline mains in natural gas local distribution systems, Environmental science & technology, 54 (2020) 8958-8967.

[8] J. Wang, L.P. Tchapmi, A.P. Ravikumar, M. McGuire, C.S. Bell, D. Zimmerle, S. Savarese, A.R. Brandt, Machine vision for natural gas methane emissions detection using an infrared camera, Applied Energy, 257 (2020) 113998.

[9] Y. Zhang, R. Gautam, S. Pandey, M. Omara, J.D. Maasakkers, P. Sadavarte, D. Lyon, H. Nesser, M.P. Sulprizio, D.J.J.S.a. Varon, Quantifying methane emissions from the largest oil-producing basin in the United States from space, Science advances, 6 (2020) eaaz5120.

[10] A. Safari, N. Das, O. Langhelle, J. Roy, M. Assadi, Natural gas: A transition fuel for sustainable energy system transformation?, Energy Science & Engineering, 7 (2019) 1075-1094.

[11] E.P.A. (EPA), Overview of Greenhouse gas., url:https://www.epa.gov/ghgemissions/overview-greenhouse-gases.

[12] S. Conley, G. Franco, I. Faloona, D.R. Blake, J. Peischl, T.J.S. Ryerson, Methane emissions from the 2015 Aliso Canyon blowout in Los Angeles, CA, Science, 351 (2016) 1317-1320.

[13] S. Pandey, R. Gautam, S. Houweling, H. Denier van der Gon, P. Sadavarte, T. Borsdorff, O. Hasekamp, J. Landgraf, P. Tol, T. van Kempen, R. Hoogeveen, R. van Hees, S.P. Hamburg, J.D. Maasakkers, I. Aben, Satellite observations reveal extreme methane leakage from a natural gas well blowout, Proc Natl Acad Sci U S A, 116 (52) (2019) 26376-26381.

[14] I. Irakulis-Loitxate, L. Guanter, Y.-N. Liu, D.J. Varon, J.D. Maasakkers, Y. Zhang, A. Chulakadabba, S.C. Wofsy, A.K. Thorpe, R.M. Duren, Satellite-based survey of extreme methane emissions in the Permian basin, Science Advances, 7 (2021) 4507.

[15] E.B. Kujawinski, C.M. Reddy, R.P. Rodgers, J.C. Thrash, D.L. Valentine, H.K. White, The first decade of scientific insights from the Deepwater Horizon oil release, Nature Reviews Earth & Environment, 1 (2020) 237-250.

[16] FrackCheckWV.net, Blast and fire at blue racer processing plant in Ohio valley kills worker, http://www.frackcheckwv.net/2014/11/14/blast-and-fire-at-blue-racer-processing-plant-in-ohio-valley-kills-worker/, (2019).

[17] M. Saunois, R. Jackson, P. Bousquet, B. Poulter, J.J.E.R.L. Canadell, The growing role of methane in anthropogenic climate change, Environmental Research Letters, 11 (2016) 120207.

[18] M. Xie, S. Yan, L. Wu, L. Liu, Y. Bai, L. Liu, Y.J.J.o.C.P. Tong, A novel robust reweighted multivariate grey model for forecasting the greenhouse gas emissions, Journal of Cleaner Production, 292 (2021) 126001.

[19] N. Xu, S. Ding, Y. Gong, J.J.E. Bai, Forecasting Chinese greenhouse gas emissions from energy consumption using a novel grey rolling model, Energy, 175 (2019) 218-227.

[20] K. Leerbeck, P. Bacher, R.G. Junker, G. Goranović, O. Corradi, R. Ebrahimy, A. Tveit, H. Madsen, Short-term forecasting of CO2 emission intensity in power grids by machine learning, Applied Energy, 277 (2020).
[21] J. Ofosu-Adarkwa, N. Xie, S.A. Javed, Forecasting CO2 emissions of China's cement industry using a hybrid Verhulst-GM(1,N) model and emissions' technical conversion, Renewable and Sustainable Energy Reviews, 130 (2020).
[22] H. Wang, X. Li, D. Wang, J. Zhao, H. He, Z. Peng, Regional prediction of ground-level ozone using a hybrid sequence-to-sequence deep learning approach, Journal of Cleaner Production, 253 (2020).
[23] M.S. Bakay, Ü.J.J.o.C.P. Ağbulut, Electricity production based forecasting of greenhouse gas emissions in Turkey with deep learning, support vector machine and artificial neural network algorithms, 285 (2021) 125324.
[24] F. Ren, D.J.J.o.C.P. Long, Carbon emission forecasting and scenario analysis in Guangdong Province based on optimized Fast Learning Network, 317 (2021) 128408.
[25] J. Shi, Y. Chang, C. Xu, F. Khan, G. Chen, C. Li, Real-time leak detection using an infrared camera and Faster R-CNN technique, Computers & Chemical Engineering, 135 (2020).
[26] J. Wang, J. Ji, A.P. Ravikumar, S. Savarese, A.R. Brandt, VideoGasNet: Deep learning for natural gas methane leak classification using an infrared camera, ENERGY, 238 (2022) 121516.
[27] S. Jongaramrungruang, A.K. Thorpe, G. Matheou, C. Frankenberg, MethaNet – An AI-driven approach to quantifying methane point-source emission from high-resolution 2-D plume imagery, Remote Sensing of Environment, 269 (2022).
[28] J. Na, K. Jeon, W.B. Lee, Toxic gas release modeling for real-time analysis using variational autoencoder with convolutional neural networks, Chem. Eng. Sci., 181 (2018) 68-78.
[29] D. Song, K. Lee, C. Phark, S. Jung, Spatiotemporal and layout-adaptive prediction of leak gas dispersion by encoding-prediction neural network, Process Safety and Environmental Protection, 151 (2021) 365-372.
[30] A. Daw, A. Karpatne, W. Watkins, J. Read, V. Kumar, Physics-guided neural networks (pgnn): An application in lake temperature modeling, arXiv:.11431, (2017).
[31] D.M. Blei, A. Kucukelbir, J.D. McAuliffe, Variational inference: A review for statisticians, Journal of the American statistical Association, 112 (2017) 859-877.
[32] Y. Gal, Z. Ghahramani, Dropout as a bayesian approximation: Representing model uncertainty in deep learning, International conference on machine learning, 2016, pp. 1050-1059.
[33] Y. Liu, H. Qin, Z. Zhang, S. Pei, Z. Jiang, Z. Feng, J. Zhou, Probabilistic spatiotemporal wind speed forecasting based on a variational Bayesian deep learning model, Applied Energy, 260 (2020) 114259.
[34] P. Ni, J. Li, H. Hao, Q. Han, X. Du, Probabilistic model updating via variational Bayesian inference and adaptive Gaussian process modeling, Computer Methods in Applied Mechanics and Engineering, 383 (2021) 113915.
[35] W. Li, M.Z. Bazant, J. Zhu, A physics-guided neural network framework for elastic plates: Comparison of governing equations-based and energy-based approaches, Computer Methods in Applied Mechanics and Engineering, 383 (2021) 113933.
[36] J. Zhang, X. Zhao, Spatiotemporal wind field prediction based on physics-informed deep learning and LIDAR measurements, Applied Energy, 288 (2021) 116641.
[37] S. Horii, MLE, MAP and Bayesian Inference, url: https://towardsdatascience.com/mle-map-and-bayesian-inference-3407b2d6d4d9, (2019).
[38] C. Andrieu, N. De Freitas, A. Doucet, M.I. Jordan, An introduction to MCMC for machine learning, Machine learning, 50 (2003) 5-43.
[39] R.M. Neal, Bayesian learning for neural networks, Springer2012.
[40] C.M. Bishop, N.M. Nasrabadi, Pattern recognition and machine learning, Springer2006.
[41] J. Shi, J. Li, Y. Zhu, H. Hao, G. Chen, B. Xie, A simplified statistic-based procedure for gas dispersion prediction of fixed offshore platform, Process Safety and Environmental Protection, 114 (2018) 48-63.
[42] Y. He, H. Li, Probability density forecasting of wind power using quantile regression neural network and kernel density estimation, Energy conversion and management, 164 (2018) 374-384.