A hybrid strategy on combining different optimization algorithms for hazardous gas source term estimation in field cases

Wang, Yiduo; Chen, Bin; Zhu, Zhengqiu; Wang, Rongxiao; Chen, Feiran; Zhao, Yong; Zhang, Laobing

DOI
10.1016/j.psep.2020.02.029

Publication date
2020

Document Version
Final published version

Published in
Process Safety and Environmental Protection

Process Safety and Environmental Protection 138 (2020) 27–38

Contents lists available at ScienceDirect

Process Safety and Environmental Protection

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

A hybrid strategy on combining different optimization algorithms for hazardous gas source term estimation in field cases

Yiduo Wang[a], Bin Chen[a,*], Zhengqiu Zhu[a], Rongxiao Wang[a], Feiran Chen[a], Yong Zhao[a], Laobing Zhang[b,c]

[a] College of Systems Engineering, National University of Defense Technology, 109 Deya Road, Changsha, 410073, China
[b] Faculty of Technology, Policy and Management, Safety and Security Science Group (S3G), TU Delft, 2628 BX, Delft, The Netherlands
[c] Institute of War, Academy of Military Sciences PLA China, Beijing, 100091, China

ARTICLE INFO

Article history:
Received 8 November 2019
Received in revised form 19 February 2020
Accepted 20 February 2020
Available online 26 February 2020

Keywords:
Gas emission
Source term estimation
Neural network
Hybrid strategy
Optimization algorithms

ABSTRACT

Estimating gas source terms is essential and significant for managing a gas emission accident. Optimization method, as a kind of estimation methods, is helpful to figure out the source terms by solving the inverse problem. Significantly, the performance of optimization method on source term estimation is affected by the accuracy of forward dispersion model. To enhance the estimation accuracy, previous works have demonstrated the feasibility of using Back Propagation Neural Network (BPNN) trained by actual experimental datasets as a forward dispersion model. However, the overall accuracy of source estimation is still limited by backward estimation methods. Most related studies used a single optimization algorithm to estimate source terms, which usually fails to realize the requirements of both high calculation accuracy and satisfying computational efficiency. Therefore, a hybrid strategy was proposed in this study to combine optimization algorithms with different characteristics, including particle swarm optimization, genetic algorithm and simulated annealing algorithm, to not only achieve high accuracy in global searching, but also converge to a stable result efficiently. Finally, extensive experiments are conducted to testify our proposed hybrid optimization algorithms. The Skill scores of hybrid optimization algorithms decrease obviously compared to those of single optimization algorithm. Hence, the proposed hybrid strategy is potentially useful for guiding the combination of optimization algorithms for gas source terms estimation, which further contributes to deal with a gas emission accident with satisfying calculation accuracy and computational efficiency.

1. Introduction

Hazardous gas emission accidents have brought huge damage to the society (Yi et al., 2017). For example, The Fukushima Daiichi Nuclear Power Plant (FNPP1) accident in Japan in March 2011 caused wide spread devastation (Tsuruda, 2013). Consequently, it is of vital importance and necessity to monitor hazardous gas emission and apply different methods to estimate terms of emission source. In recent years, many researchers use different algorithms and strategies to solve the problem of source term estimation. However, these methods always have their own limitations, for instance, particle swarm optimization algorithms (PSO) is easy to fall into local optimum, or genetic algorithms (GA) has low efficiency due to its poor local searching ability. Researchers have used different ways to overcome the limitations of traditional source term estimation algorithms, such as fine-tuning the parameters of a single algorithm (Chu et al., 2011) or combine different algorithms together (Vasant et al., 2019). The former approach is difficult to find optimal values of parameters with limited performance improvement, while the latter approach still can be improved. So it is necessary to combine optimization algorithms with different characteristics under the guidance of suitable hybrid strategy for better estimation of gas source.

Source terms consist of source position, source strength and so on. These parameters can be estimated by various methods based on gas dispersion observation data, including concentration data and meteorological data (Zhu et al., 2018). Except for the direct way by portable instruments or widely distributed sensors, indirect methods by computation algorithms coupled with measurement results are also used to determine the source terms of gas emissions (Ma et al., 2014). Common computation algo-

* Corresponding author.
E-mail addresses: wangyiduo14@nudt.edu.cn (Y. Wang), nudtcb9372@gmail.com (B. Chen), zhuzhengqiu12@nudt.edu.cn (Z. Zhu), wangrongxiao12@nudt.edu.cn (R. Wang), chenfeiran14@nudt.edu.c (F. Chen), Zhaoyong15@nudt.edu.cn (Y. Zhao), zhanglaobing.nudt@gmail.com (L. Zhang).

https://doi.org/10.1016/j.psep.2020.02.029
0957-5820/

rithms include Algebraic method (Debski, 2010), Bayesian method (Xue et al., 2018) and Optimization method (Chu et al., 2011). Among them, the Algebraic method obtains the source terms by figuring out the atmospheric dispersion model inversely. But due to the complexity of atmospheric dispersion, it is hard to handle the problem directly under most conditions (Debski, 2010). The goal of the Bayesian method is to obtain the posterior probability density distribution of the source terms by Bayesian inference, and then estimate the source terms by probability density (Keats et al., 2007). The Optimization method is mainly to find the optimal value of the source terms by minimizing the cost function (Vasant et al., 2020). There are various optimization algorithms, such as PSO, GA, simulated annealing algorithms (SA), firefly algorithms (Fister et al., 2013). PSO is a commonly used optimization algorithm originated in studies of synchronous bird flocking and fish schooling. A collection of particles is defined in PSO corresponding to a candidate solution to the optimization problem (Boeringer and Werner, 2004). Motivated by Darwin's theories of evolution and the concept of "survival of the fittest", genetic algorithms use processes analogous to genetic recombination and mutation to promote the evolution of a population that best satisfies a predefined goal (Khlaifi et al., 2009). SA is a stochastic optimization procedure which is widely applicable and has been found effective in several problems arising in different fields (Gibson et al., 2002). Ma (Ma et al., 2017) combined PSO algorithm with Tikhonov regularization to identify source parameters including source strength and location. The comparison results of simulation and experiment case showed that the linear Tikhonov–PSO method with transformed linear inverse model has high computation efficiency. Khlaifi et al. (Khlaifi et al., 2009) applied GA and Gaussian diffusion model for the source quantification. The study demonstrated that this method makes it possible to identify the emission parameters of the main source in the study zone. Thomson (Thomson et al., 2007) used a random search algorithm and a SA algorithm to locate the known gas diffusion sources in the desert. However, researches show that different optimization algorithms have their own advantages and limitations. For example, algorithms with fast calculation speed are often trapped in local optimum and hard to obtain global optimal solutions, like PSO algorithm. While algorithms with strong searching ability usually show a poor performance in computational efficiency, like GA algorithm. Combining algorithms with different characteristics together is an obviously suitable way to obtain better performance of estimation results. Furthermore, the accuracies of estimation results largely depend on the accuracy of the forward dispersion model that is used in backward calculation (Wang et al., 2018;Vasant et al., 2020). The error of the forward dispersion model will affect the range of the error of the inverse estimation. Therefore, it is necessary to build an accurate forward dispersion model for source term estimation.

In the study of forward atmospheric dispersion modeling, researchers used partial differential equations to describe the gas dispersion process. With the development of computer technology, people began to use computer simulation to numerically simulate and calculate gas dispersion. (Sheppard, 1962). Gaussian dispersion model (Hanna et al., 1982), Lagrange stochastic model (Wilson and Sawford, 1996) and Computational Fluid Dynamics model (CFD) (Efthimiou et al., 2017) are currently three commonly used methods. However, methods with high computational efficiency, like Gaussian dispersion model, often have low accuracy in complicated scenes. While methods with high calculation accuracy, like Lagrange stochastic model and CFD, are always computationally expensive (Qiu et al., 2018). In order to solve the problem that aforementioned atmospheric dispersion models are difficult to balance the accuracy and computational efficiency, Artificial Neural Network (ANN) has been applied to this field in recent years (Boznar et al., 1993; Chen et al., 2020). As one of the most widely used ANN models, Back Propagation Neural Network (BPNN) is a multilayer feedforward network trained by error inverse propagation algorithm. The BPNN model can approximate an arbitrary function theoretically and has strong nonlinear mapping ability. After being trained by dataset, the BPNN model can predict the dispersion of hazardous gas accurately and efficiently (Valeriy and Pandian, 2018). Many researchers have applied this method into their studies. So et al. (So et al., 2010) used neural network to estimate the release rate based on real-time sensor data. The results indicated that the proposed technique can estimate release rates effectively within seconds. Lauret et al. (Lauret et al., 2016) coupled Cellular Automata with ANN to calculate the atmospheric dispersion of methane in 2D scenario. Then, Efforts are made in reducing computation time while keeping an acceptable accuracy. Ma (Ma and Zhang, 2016) applied different machine learning algorithms, including BPNN, Radial Basis Function (RBF) network and Support Vector Machine (SVM), coupled with Gaussian parameters to predict forward atmospheric dispersion and compare the performance of different algorithms. These studies have shown that neural network has good performance in the simulation of atmospheric forward dispersion, and thus has positive effects on improving the accuracy of inverse source term estimation.

In this study, the state-of-the-art BPNN-based method is used to simulate the gas forward dispersion process. After trained by abundant actual experiment data, the neural network exhibits good performance in both calculation accuracy and calculation speed compared with traditional atmospheric dispersion models. The trained BPNN also ensures the accuracy and efficiency of inverse estimation result on source term. In addition, as it is the first step to investigate the feasibility of using hybrid strategy in the realm of source term estimation, we selected three most commonly used optimization algorithms (PSO, GA and SA) and combined them together based on the hybrid strategy. The estimation results on source terms of actual dataset illustrate that each step of combination overcomes several limitations of original algorithms. As for the final PSO + GA + SA hybrid algorithm, in the early stage of operation, the hybrid optimization algorithm has large randomness, ensuring strong global search ability for an accurate solution. As the number of iteration increases, the randomness of hybrid optimization algorithm reduces gradually, in order to improve the local search ability and obtain a stable solution. Experimental results show that a hybrid strategy combining three common and naïve optimization algorithms can even achieve better performance compared to traditional strategies. This implies that a hybrid strategy could be a promising and useful tool in the emergency management of gas emission accidents.

The rest of this paper is organized as follows. Section 2 describes the principle of atmospheric forward dispersion model (BPNN) and optimization algorithms, as well as the proposal of hybrid strategy. The establishment and training of BPNN based on field dataset (the Prairie Grass Project) is demonstrated and analyzed in Section 3. Section 4 gives the source estimation experiments of three optimization algorithms. The comparison and analysis of estimation results are also discussed in this section. The discussion is given in Section 4, followed by the conclusions in Section 5.

2. Models and methods

2.1. Atmospheric forward dispersion model

To solve the problem of source term estimation, it often requires the forward atmospheric dispersion model to behave well in computational accuracy and efficiency. However, existing atmospheric dispersion models, like the Gaussian model, Lagrange stochastic model and CFD model, cannot balance the requirements of high

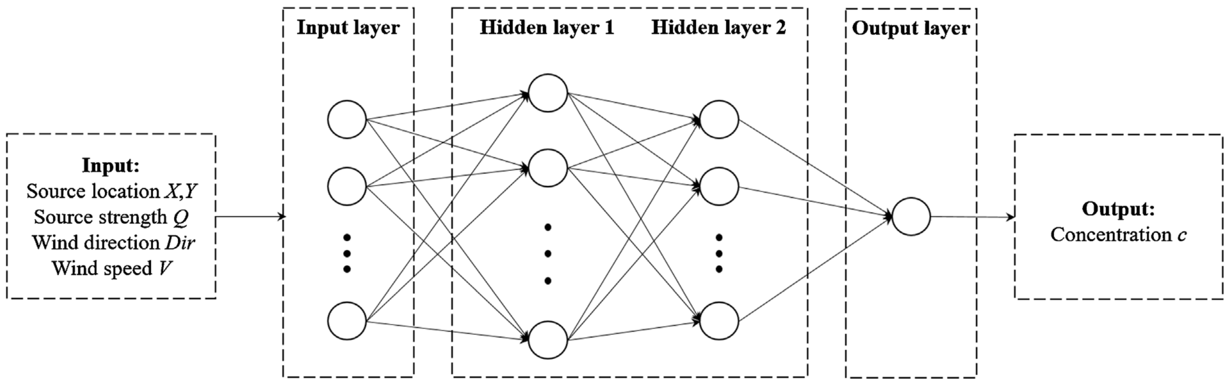

Fig. 1. Structure of the neural network.

estimation accuracy and good computational efficiency. To deal with the problem, ANN has been applied in recent years to predict the concentration distribution of gas dispersion. BPNN is a multi-layer feedforward network trained by error inverse propagation algorithm (Valeriy and Pandian, 2020a). As for computational efficiency, the computation time of BPNN is mainly cost in the neural network training process, but the training can be completed in advance (Valeriy and Pandian, 2020b). Once the training is completed, as long as input data follow the corresponding format in the inference process, BPNN can output the prediction results quickly (Thomas et al., 2020). In terms of computational accuracy, BPNN has a strong nonlinear fitting ability. Through the training of abundant data, it can fit the complex relationship between the input and the output with high precision (Pelliccioni and Tirabassi, 2006).

Therefore, this study uses BPNN model to predict the concentration distribution of gas accurately and quickly. Noticeably, the structure of our BP model is shown in Fig. 1, including an input layer, two hidden layers and an output layer. The number of input layer neurons equals the dimension of training data, matching the number of source terms to be estimated. The neuron numbers of two hidden layer are continuously debugged according to the dataset characteristics and training results. Since the goal is to predict the gas concentration distribution, the output layer has only one neuron, representing the concentration of hazardous gas.

There exist many factors affecting the gas concentration distribution in different degrees, like emission parameters, meteorological parameters and topography parameters. In this paper, five most important parameters, including source position *X*, *Y*, source strength *Q*, wind speed *V* and wind direction *Dir*, are selected as the parameters of source term estimation (Wang et al., 2018). The establishment and training process of BPNN will be described in Section 3.2.

2.2. Optimization algorithms and hybrid strategy

To our knowledge, this study is the first work using hybrid strategy in the realm of source term estimation. Therefore, the goal of this study is not using the state-of-the-art techniques and fine-tuning their parameters. Instead, we simply select three most commonly used optimization algorithms (PSO, GA, and SA) to testify the feasibility of using hybrid strategy. Moreover, the hybrid strategy is not a random combination, but a strategy with progressive relation based on the characteristics of different optimization algorithms.

2.2.1. Particle swarm optimization algorithm

PSO was first proposed by Eberhart and Kennedy in 1995 (Kennedy and Eberhart, 1995), and its basic concept stems from the study of the foraging behavior of flocks. The PSO algorithm is simple and easy to implement, with few parameters and fast convergence. The process of PSO can be described as follows. There are *N* particles in the *D*-dimensional searching space corresponding to candidate solutions to the optimization problem. The position of the particle *i* in the *k*th iteration is X_i^k, which is updated by the influence of the velocity (V_i^k) of the particle *i*. $pbest_i = (p_{i1}, p_{i2}, \ldots p_{iD})$ is the previous best position of particle *i*, and $gbest = (g_1, g_2, \ldots g_D)$ is the global best position of the swarm. After getting $pbest_i$ and *gbest*, each particle can update its position and velocity according to following formula:

$$V_i^{k+1} = \omega \bullet V_i^k + c_1 \bullet rand_1 \bullet (pbest_i - X_i^k) + c_2 \bullet rand_2 \bullet (gbest - X_i^k) \quad (1)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (2)$$

Where ω is the inertia factor, used to control the velocity of particles. i = 1, 2, 3, ... N is the particle number. c_1 and c_2 are called learning factors or acceleration coefficients. Appropriate c_1 and c_2 can not only accelerate the convergence speed but also refrain from falling into local optimum easily to some extent. In order to balance the effect of random factors, c_1 and c_2 are set equal generally. $rand_1$ and $rand_2$ are random numbers between 0 and 1. According to Eq. (1), the next iteration velocity of particle *i* is determined by its current velocity, position, $pbest_i$ and *gbest*.

Generally, there are many local optimal values distributed around the global optimal value. Due to the characteristics of PSO algorithm, once the algorithm finds a local optimal value, it will fall into it and thus fail to find out the global optimal value (Bhushan and Pillai, 2013).

2.2.2. Genetic algorithm

The Genetic Algorithm (GA) originated from computer simulation studies of biological systems. It is a random global search and optimization method that mimics the evolutionary mechanism of biological evolution in nature, drawing on Darwin's evolution theory and Mendel's genetic theory (Angeline et al., 1994). The GA use processes analogous to genetic recombination and mutation to promote the evolution of a population that best satisfies a predefined goal. It is an efficient, parallel, global search method in essence, which can automatically acquire and accumulate knowledge about the search space, and control the search process adaptively to obtain the best solution (Boeringer and Werner, 2004). It has many advantages, like parallel computing capabilities, scalability, and easy to combine with other technologies and algorithms. The biggest advantage is that it scarcely falls into the local optimization during the search process. Even in the case that defined cost function is discontinuous, irregular, or noisy, the global optimal solution can be found with high probability (Boeringer and Werner, 2004) (Fig. 2).

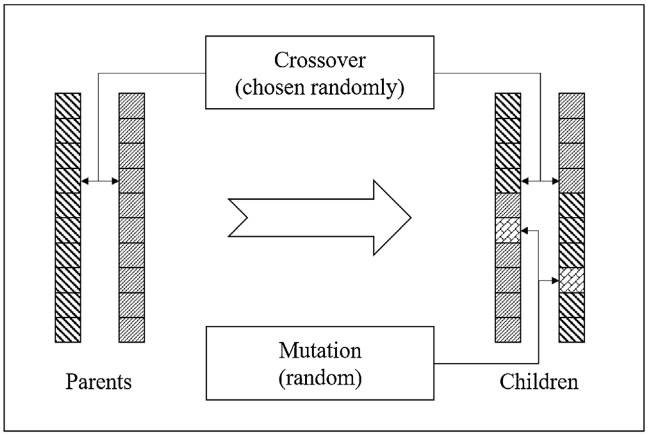

Fig. 2. Crossover and mutation in genetic algorithm.

One of the core ideas of GA is chromosome coding, which establishes the mapping between actual parameters and genes. In this paper, the selected source terms and corresponding velocity are considered as genes on different chromosomes. The core steps of the GA are crossover, mutation and selection. The crossover operation refers to the exchange of two partial chromosomes in a certain way to form two new individuals. One-point Crossover, Two-point Crossover, Multi-point Crossover, Uniform Crossover and Arithmetic Crossover are typical means of crossover. The mutation operation refers to the replacement of several genes in the individual chromosome with other alleles, thereby forming new individuals. The means of mutation usually include Simple Mutation, Uniform Mutation, Boundary Mutation, and Non-uniform Mutation. The selection operation refers to the screening of all individuals in the population according to certain strategy. There are many common selection strategies, such as Roulette Wheel Selection, Stochastic Tournament and Elitist Selection. Crossover and mutation operations increase the diversity of the population, thus expanding the parameters search space, guaranteeing the GA behave well in the global search.

2.2.3. Simulated annealing algorithm

The earliest idea of Simulated Annealing (SA) was proposed by N. Metropolis et al. in 1953 (Metropolis et al., 1953). In 1983, S. Kirkpatrick et al. introduced annealing ideas into the field of combinatorial optimization successfully (Kirkpatrick, 1984). The SA is a random optimization algorithm based on the Monte-Carlo iterative solution strategy. According to the Metropolis criterion (Metropolis et al., 1953), the probability that particles tend to balance at temperature T is $\exp\left(-\Delta E/(kT)\right)$, where E is the internal energy at temperature T, ΔE is the change of internal energy, and k is the Boltzmann constant. Metropolis guidelines are often expressed as:

$$p = \begin{cases} 1 if E_{new} < E_{old} \\ \exp(-\dfrac{E_{new} - E_{old}}{T}) if E_{new} \geq E_{old} \end{cases} \tag{3}$$

The Metropolis criterion shows that when temperature is T, the probability of cooling with an energy difference of ΔE is $P(\Delta E)$. When the energy difference ΔE is constant, the cooling probability $P(\Delta E)$ is positively correlated with the temperature T. Since the temperature decreases gradually according to Eq. (4) during the annealing process, the cooling probability P also reaches 0 gradually.

$$T = \alpha \bullet T, \alpha \in (0, 1) \tag{4}$$

One obvious characteristic of the SA is that there is a certain decreasingly probability P in the running process to accept a point near the local optimal solution, so it is possible to jump out of the local optimum and find out the global optimal solution finally.

2.2.4. Hybrid strategy

As mentioned before, in the problem of source term estimation of hazardous gas, single optimization algorithm can hardly meet the requirements of high estimation accuracy and high computational efficiency at meantime. For example, the PSO has simple particle update step and fast calculation speed, but it is easy to fall into local optimum (Ni et al., 2013). Due to the crossover and mutation steps, the GA has strong global search ability, but it is difficult to converge to a stable value, and its calculation time is long. The local search ability and global search ability of the SA are greatly affected by initial temperature T and annealing coefficient α. If the initial temperature is high, it is highly possible for the algorithm to find out a global optimal solution successfully. When α reaches 0, the temperature drops rapidly, and thus the global search ability of the algorithm is poor. When α is close to 1, the temperature drops slowly, resulting in a longer calculation time.

Single optimization algorithm has its own limitations when solving the problem of source term estimation. Therefore, it is necessary to propose a hybrid strategy to combine different algorithms together. Thus, a hybrid algorithm which overcomes the limitations of single algorithm can be applied in the estimation of source term. The detailed hybrid strategy is exhibited in Fig. 3.

The three selected optimization algorithms (PSO, GA, SA) have different characteristics. Apart from SA, PSO and GA contain a number of particles or individuals corresponding to candidate solutions. Although PSO and GA possess similar capabilities, PSO has much simpler principle and implementation. As a result, PSO is chosen as the basic algorithm to combine with others. Single PSO is prone to fall into a local optimum and hard to obtain global optimal values. To enhance the global searching ability of single PSO, it is necessary to combine PSO with other optimization algorithms with good performance on global searching, such as GA and SA. Compared with SA, GA is more suitable for solving problems with a number of particles or individuals. Specifically, the PSO is added with the crossover, mutation and selection steps to obtain the hybrid PSO + GA algorithm as the guidance of hybrid strategy. These three steps can increase the population diversity of the particle swarm and expand the parameters search space in theory, so that the hybrid PSO + GA has stronger global search ability than single PSO. This paper chooses Arithmetic Crossover, Uniform Mutation and Elitist Selection as implementation methods in these three steps. Arithmetic Crossover refers to selecting two individuals as the male and female parents randomly, then producing one filial generation by the linear combination of these two parents. Uniform Mutation is replacing the original gene values at each chromosome with a small probability. Elitist Selection means sorting all particles in the population according to the cost function value, then selecting the individuals with small cost function value as the next generation population.

As for the PSO + GA, the mutation step which is controlled by the mutation probability, has the most significant effect on parameters search space expansion. If the mutation probability value is too small, the PSO + GA is difficult to jump out of the local optimum; while if the mutation probability value is too large, the calculation results hardly converge to a stable value. Therefore, the idea of the SA is introduced to obtain the PSO + GA + SA algorithm. The temperature T and the annealing coefficient α are used to control the variation of the mutation probability. As a result, the mutation probability decreases with the increase of iteration number. In conclusion, the hybrid strategy can ensure that the combined algorithm has a large parameters-search-space and a strong global search ability in the early stage; as the iteration goes on, the calcu-

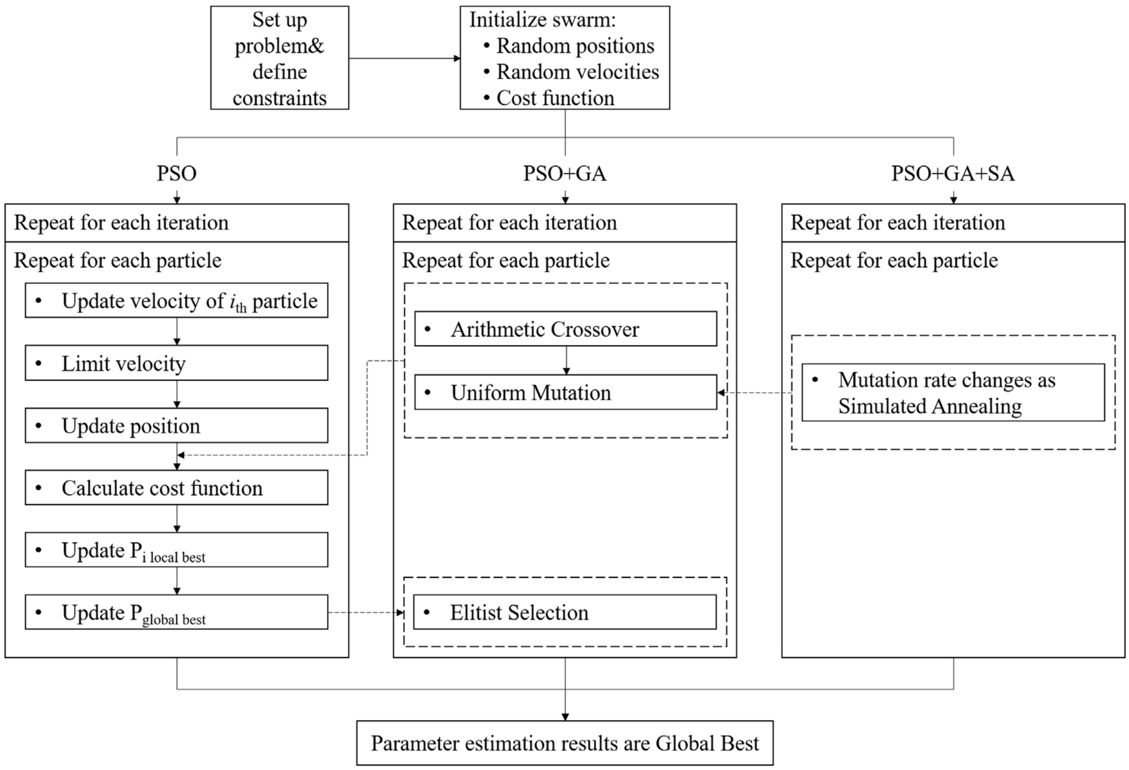

Fig. 3. Algorithm hybrid strategy and flowchart.

lation result of each step converges gradually, and a stable global optimal solution is finally obtained.

3. Gas concentration prediction based on BPNN and field dataset

3.1. Data acquisition and network construction

3.1.1. Project prairie grass dataset

The Prairie Grass Project is a well-known experiment dataset of actual field. It is a typical hazardous gas emission case with low-chimney source in flat terrain and open air conditions. The experiments were conducted in O'Neil, Nebraska (USA, 42.493 °N, 98.572 °W) from July to August 1956 (Barad, 1958). Hazardous gas (SO_2) was released from a continuous point source at a height of 0.46 m. Gas concentration data were collected by sensors at the height of 1.5 m. These sensors were centered on the gas emission source, and evenly distributed on five semi-circular arcs (located at 50, 100, 200, 400, 800 m in the downwind direction, respectively).

The spacing between sensors was two degrees on the inner four semi-circular arcs, and one degree on the outermost semi-circular arc. In terms of meteorological data, the wind direction and wind speed were measured for about 10 or 20 min by two sensors, one was 25 m directly west of the emission source, another was 450 m north and 30 m west of the source.

The dataset contains 68 groups of amount to 8246 gas concentration and meteorological data. All the data are divided into two parts randomly, some of which contains 60 groups (6954 experimental data) to train and validate the BPNN. While the other part contains 8 groups (1292 experimental data) to test the neural network and carry out the optimization algorithms experiments.

3.1.2. Establishment and training of BPNN

In this section, a neural network will be established and trained based on the Prairie Grass dataset. According to the training and prediction results of the BPNN, the parameters are adjusted continuously. Thus a suitable neural network is obtained finally.

Table 1
Neural network model parameters.

Parameters name	Setting value
Maximum training number	200
Training goal	1000
Learning rate	0.005
Momentum factor	0.9
Maximum failures number	100

In the establishing and training process, the most important is the setting of various parameters. Among all parameters involved in the BPNN, some of them control the training process and affect the training results, like maximum training number, training goal, learning rate, momentum factor and maximum failures number. After a large number of experiments, these parameters are set to the values shown in Table 1.

The neuron number in two hidden layers also has obvious influence on the performance of the BPNN model. If the number is too small, the fitting and predicting performance of the network will be poor; while if the number is too large, the training time and error will arise, leading to reduction of the generalization ability of the network. Accordingly, the selection of the neuron number in hidden layers is also important. Moreover, because of the randomness of the network training in MATLAB software, different trainings will have different results even if the neural network is set to the same parameters. It is necessary to repeat training process to obtain the best network (Ma and Zhang, 2016).

3.2. Result analysis

The training results are always measured by neural network characteristic parameters, such as normalized mean squared error

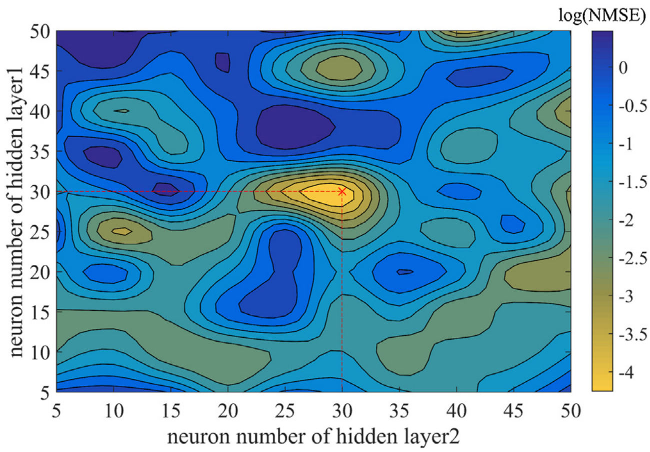

Fig. 4. The log(NMSE) of different combinations of neuron number.

(NMSE), mean squared error (MSE), R-squared (R^2) (Pelliccioni and Tirabassi, 2006). This paper determines the number of neuron in hidden layers by calculating the NMSE value of the neural network (Wang et al., 2018). For better distinction, the function log is applied to NMSE (shown in Fig. 4). The region with light yellow means smaller NMSE value compared with the region with dark blue. Small NMSE value indicates that the predicting result of neural network is highly accurate. Therefore, the combinations of the neuron number in two hidden layers corresponding to the light yellow region can obtain neural networks with good prediction performance potentially. Finally, the neuron number in two hidden layers is determined as [30, 30].

We randomly select two experimental datasets (Release 17, 22) of the Prairie Grass Project (Barad, 1958) to test the trained neural network. The gas concentration values of the experimental data and neural network output data in the two release experiment datasets are shown in the two sub-graphs of Fig. 5. It can be seen that although some prediction results of the neural network are deviated from experimental concentration values, most predicted concentrations agree well with the experimental data.

The scatterplot and linear fit of the experimental data and neural network output data are given in the two graphs of Fig. 6. The linear fit curves of the two graphs are very close to "y = x", indicating that the prediction results of the neural network are very close to the experimental data. In addition, R^2 in graph (a) is 0.9629, and graph (b) is 0.9657, which further demonstrates the great performance of the neural network in predicting gas concentration.

4. Source estimation experiments and results analysis

The flow of the hybrid strategy is shown in Fig. 3. In order to compare the influence of the hybrid strategy on the estimation of source terms, this paper compares three optimization algorithm combinations (PSO, PSO + GA, PSO + GA + SA) in MATLAB software (2015a version) through comprehensive estimation experiments. The performance of these algorithms is compared and analyzed from the aspects of accuracy and speed according to the experiment results.

4.1. Assessment parameter

One important parameter for optimization algorithms is cost function, whose purpose is measuring the error between the actual value and the predicting value during the calculation process. Since these three algorithms are based on the frame of the PSO, the def-

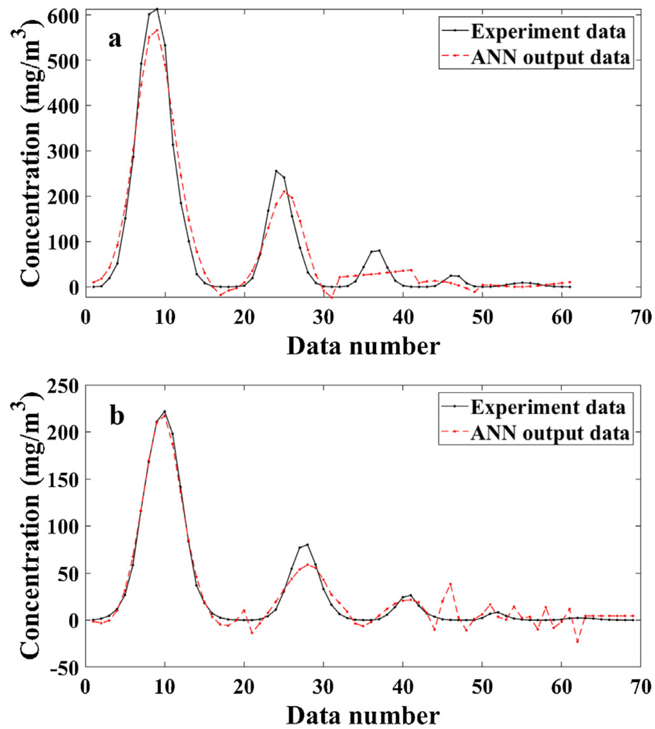

Fig. 5. Experimental data and neural network output data. (a) is the results of release 17 and (b) is the results of release 22.

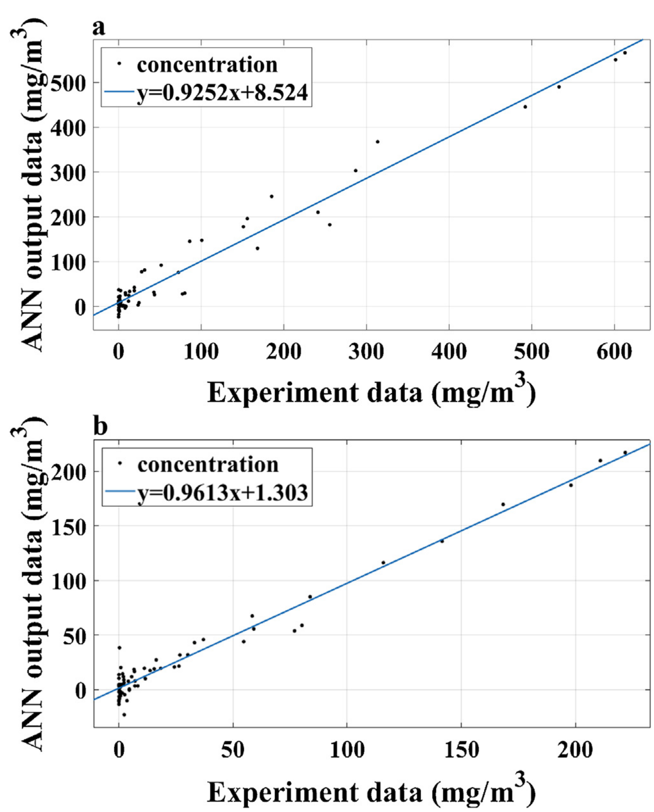

Fig. 6. Experimental data and neural network output data fitting curve. (a) is the results of release 17 and (b) is the results of release 22.

inition of cost function are similar. There are many forms of cost function (Hui and Zhang, 2017). In this paper, the MSE is chosen as the cost function, shown in Eq. (5).

$$f_i = \frac{1}{m}\sum_{j=1}^{m}\left[C_{est,i}(:) - C_{act,i}(:)\right]^2 \tag{5}$$

Where f_i is the cost function value of the particle i. $C_{est,i}(:)$ indicates the predicted gas concentration distribution value of the particle i, while $C_{act,i}(:)$ indicates the actual gas concentration distribution value. The dimension of these two parameters is m, which corresponds to the number of gas concentration data measured in different experiments. It's worth mentioning that the value of m varies in different experiments.

As for measuring the results of source term estimation, there exit many approaches used by researchers. In this paper, there are five source terms to be estimated in total. It is necessary to measure the accuracy of the estimation of all source terms. Fortunately, Skill scores can be constructed to quantify the closeness of each estimation to the real source terms (Long et al., 2010; Ma et al., 2017). The Skill score combines individual component equations, one to quantify the accuracy of each source term (i.e. source location, source strength, wind direction and wind speed) as determined by different optimization algorithms.

$$S_x = \max\left(\frac{x_{act} - x_{est}}{x_{act} - x_{\min}}, \frac{x_{est} - x_{act}}{x_{\max} - x_{act}}\right) \tag{6}$$

$$S_y = \max\left(\frac{y_{act} - y_{est}}{y_{act} - y_{\min}}, \frac{y_{est} - y_{act}}{y_{\max} - y_{act}}\right) \tag{7}$$

S_x and S_y are the individual Skill score of estimating x and y coordinate, respectively. Eq. (6) and Eq. (7) are similar, the subscript "*act*" indicates the actual value, and the subscript "*est*" indicates the predicted value. The subscript "min" means the minimum value of this parameter in the dataset, and the subscript "max" means the maximum value. Taking Eq. (6) as example, if $x_{act} > x_{est}$, thus $x_{act} - x_{est} > 0$ and $x_{est} - x_{act} < 0$, S_x is calculated by the former formula. While if $x_{act} < x_{est}$, S_x is calculated by the latter formula.

$$S_q = \frac{\left|Q_{act} - Q_{est}\right|}{Q_{act}} \tag{8}$$

Different from x and y coordinate, source strength Q is an one-dimensional variable. Thus the expression of Eq. (8) is simpler than Eq. (6) and Eq. (7). S_q is the relative error of the actual value and the predicted value.

$$S_{dir} = \min\left(\left|Dir_{act} - Dir_{est}\right|, 360 - \left|Dir_{act} - Dir_{est}\right|\right)/180 \tag{9}$$

The range of wind direction is [0,360), thus the absolute value of difference between the actual and predicted direction may be calculated in two ways. The denominator is set to 180, so as S_{dir} can be normalized to the internal [0, 1.0].

$$S_v = \frac{\left|V_{act} - V_{est}\right|}{V_{act}} \tag{10}$$

Similar to source strength Q, wind direction V is also an one-dimensional variable. The expression of Eq. (10) means the relative error of the actual value and the predicted value of V. On most occasions, the score value of each term in Eq. (6)–(10) is normalized to the interval [0, 1.0]. If the calculated value is bigger than 1, it is set to 1.

$$Skillscore = (\omega_1 S_x + \omega_2 S_y + \omega_3 S_q + \omega_4 S_{dir} + \omega_5 S_v)/5 \tag{11}$$

The Skill score is the combination of individual component equations. However, the importance of each source term is different. In Eq. (11),ω_i, $i = 1, 2, 3, 4, 5$ is the weight of above five terms score. According to article (Hui and Zhang, 2017), the five weights are set to 5.0, 5.0, 2.0, 1.0 and 2.0 respectively. These weights reflect the importance level of these terms in different scenarios. Among these terms, the most important is the coordinates of the gas emission source, so ω_1 and ω_2 are both large values (5.0). Source strength Q and wind speed V are also very important parameters and are difficult to estimate, so the weights ω_3 and ω_5 values are set to 2.0. Obviously, lower Skill score means the estimation results are more accurate.

Table 2
PSO algorithm initialization parameters.

C_1	C_2	Iteration number	Particle number	Parameter dimension
2	2	100	100	5

4.2. Source terms estimation experiments

4.2.1. Estimation experiment of single PSO

Firstly, single PSO uses the trained network as the forward atmospheric model to perform source terms estimation experiment. According to the discussion before, the pseudocode of the experiment is presented as Algorithm 1.

The algorithm takes Release 17 or 22 of the Prairie Grass Dataset as the input. The output is the estimations of source term (source position, source strength, wind direction and wind speed) and cost function value. In the Initialization process, key parameters of PSO are set as Table 2. The position and velocity of each particle are assigned randomly in the problem hyperspace. In each iteration, all particles repeat step 4-8. The velocity of each particle is updated according to Eq. (1). The limit of velocity is also considered. Then, the position of each particle is updated according to Eq. (2). The value of position corresponds to source terms, which are the input of BPNN. Based on the predicted concentration of BPNN, the cost function of every particle can be evaluated by Eq. (5). Calculate and compare the cost function value of all particles, the $pbest_i$ of each particle and the $gbest$ in every iteration can be updated.

Algorithm 1: PSO

Input: Release 17 or 22 of field dataset
Output: Estimations of source term, cost function value

1 Initialization: Key parameters, position X, velocity V
2 Repeat k=1: Iteration number
3 Repeat i=1: Particle number
4 $V_i^{k+1} = \omega * V_i^k + c_1 * rand_1 * (pbest_i - X_i^k) + c_2 * rand_2 * (gbest - X_i^k)$;
5 $X_i^{k+1} = X_i^k + V_i^{k+1}$;
6 $C_{est,i}(:) \leftarrow predict(\ )$;
7 $f_i = \frac{1}{m}\sum_{j=1}^{m}\left[C_{est,i}(:) - C_{act,i}(:)\right]^2$;
8 Update $pbest_i$, $gbest$;
9 End
10 End

Continuous cost function value can exhibit the calculation process of estimation experiment. Fig. 7 shows the change of cost function value with iteration number varying in three experiments of PSO algorithm in two cases (release 17 and release 22). It can be seen from the plot that although the initial cost function value in different experiments differs greatly, the cost function values all decrease successively and eventually stabilize as the iteration number increases. However, the final stable cost function values are slightly different in various cases. To be specific, the final cost function value in release 22 is smaller than the value in release 17, indicating that the source terms in release 22 are estimated more accurate as a whole.

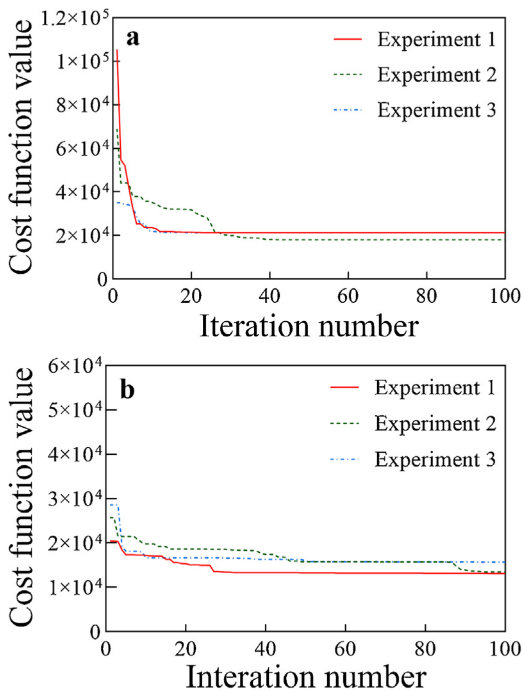

Fig. 7. Cost function value changes with iteration number varying in three different PSO experiments. (a) is the results of release 17 and (b) is the results of release 22.

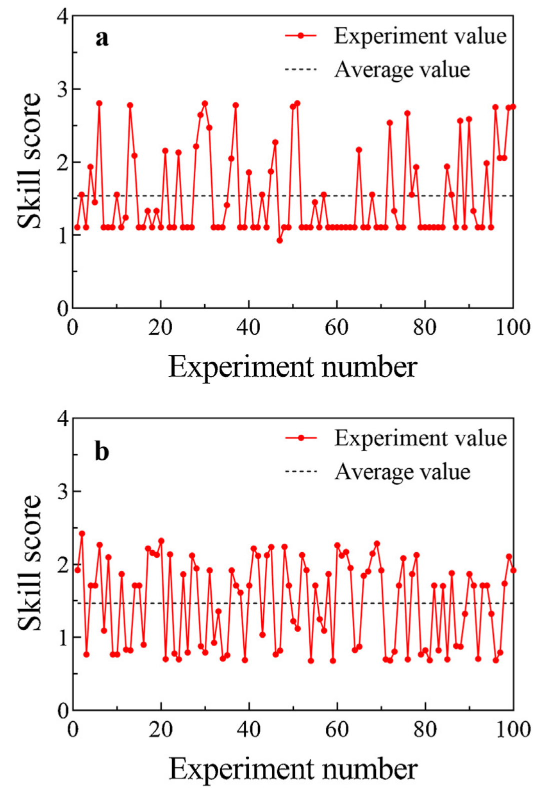

Fig. 8. The Skill score of 100 PSO experiments. (a) is the results of release 17 and (b) is the results of release 22.

In order to demonstrate the performance of the PSO, this paper repeated the experiments 100 times in release 17 and release 22. The Skill score of experimental results was calculated by Eq. (6)–(11), shown in Fig. 8. It can be seen from the plot that the average Skill score of 100 experimental results of release 17 is 1.5368, and all Skill scores are within the range of 0.9280 ~2.806. However, many points (54) have same Skill score (1.1079), which is bigger than the minimum value. These Skill scores are due to the fact that the PSO algorithm is highly possible falling into the local optimal value, and cannot search for better results (lower Skill scores). Moreover, there exist some Skill scores whose values are bigger than 2, indicating that these locally optimal estimation results are inaccurate. In order to reduce the probability that the algorithm falls into the local optimum and improve the global search ability, single PSO algorithm needs to be combined with other algorithms.

4.2.2. Estimation experiment of PSO + GA algorithm

According to Fig. 3, based on the PSO algorithm, three steps (crossover, mutation and selection) are added by referring to the idea of GA. The pseudocode of the experiment is illustrated as Algorithm 2.

In Algorithm 2, the crossover and mutation steps are performed with a certain probability. In this paper, the crossover probability is set to 0.9 and the mutation probability is set to 0.03 after hundreds of attempts. When the generated random number is less than the crossover probability, the crossover step is performed to obtain the velocity and position of the progeny particles. When the generated random number is smaller than the mutation probability, the mutation step is performed to initialize the velocity and position of the particle directly, replacing the original particle parameters. After performing the crossover step to generate another 50 individuals, the number of particles in the entire population will increase to 150. In order to maintain the stability of the particles number in each generation, the Elitist Selection strategy is adopted when the selection step is performed, and 100 individuals with smaller cost function value are selected as the next generation population.

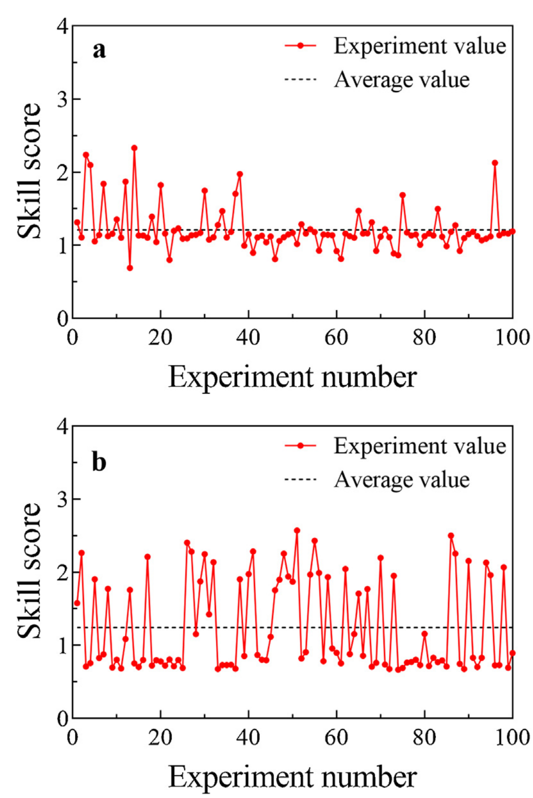

Fig. 9. The Skill score of 100 PSO + GA experiments. (a) is the results of release 17 and (b) is the results of release 22.

The core of the PSO + GA algorithm is the introduction of crossover and mutation steps. These steps make particles not move directly to the direction determined by the global optimal value and the individual optimal value, but with some deviation based on this direction. In some situations, particles which fall into the local optimal values may even jump out completely and perform random initialization. Therefore, the hybrid algorithm reduces the probability of falling into local optimum in the estimation process, and improves the global search ability.

Algorithm 2: PSO+GA

Input: Release 17 or 22 of field dataset
Output: Estimations of source term, cost function value

1 Initialization: Key parameters (adding new ones), position X, velocity V
2 Repeat k=1: Iteration number
3 Repeat i=1: Particle number
4 $V_i^{k+1} = \omega \cdot V_i^k + c_1 \cdot rand_1 \cdot (pbest_i - X_i^k) + c_2 \cdot rand_2 \cdot (gbest - X_i^k)$;
5 $X_i^{k+1} = X_i^k + V_i^{k+1}$;
6 End
7 While j< progeny particles number
8 $PX_j^{k+1}, PV_j^{k+1} \leftarrow crossover(\ )$;
9 $PX_j^{k+1}, PV_j^{k+1} \leftarrow mutation(\ )$;
10 End
11 Repeat i=1:Particle number + progeny particles number
12 $C_{est,i}(:) \leftarrow predict(\ )$;
13 $f_i = \frac{1}{m}\sum_{j=1}^{m}\left[C_{est,i}(:) - C_{act,i}(:)\right]^2$;
14 Update $pbest_i$, $gbest$;
15 $Section(\ )$;
16 End
17 End

Fig. 9 shows the results of 100 PSO + GA algorithm experiments in two cases. It can be seen that for release 17, the average value in these experiments is 1.2131. Although most values are smaller than the average value, there still exist a few Skill scores of experimental results larger than 2. These large values have a negative effect on the performance of the algorithm. The Skill scores of 100 experiments in release 22 have the similar characteristics, but the distribution of Skill score is different from that in release 17.

Comparing the corresponding graphs in Fig. 8 and Fig. 9, it can be seen that the average values in Fig. 9(a) and Fig. 9(b) are smaller in contrast with those in Fig. 8(a) and Fig. 8(b). This phenomenon shows that after combining the PSO with the GA, the global search ability of hybrid algorithm is improved. However, the introduction of the mutation step leads to new problems. Specifically, a large mutation probability makes it difficult to converge to a stable value, and a small mutation probability does not make an obvious effect to improve the global search ability. After abundant trials, the mutation probability is set to 0.03. Although this value can balance the problem of convergence to a stable value and global search ability to some extent, there still exist some estimation results which occur large jump at the end of the iteration calculation, leading to high Skill scores, such as the values of the 3rd, 4th, 14th and 96th experiments in Fig. 9(a), and the 26th, 51 st and 86th experiments in Fig. 9(b).

4.2.3. Estimation experiments of PSO + GA + SA algorithm

In order to improve the performance of PSO + GA algorithm, the idea of SA algorithm is used to control the mutation probability by the temperature value in the simulated annealing process. The flow of PSO + GA + SA is similar to PSO + GA. Except for adding some new parameters (e.g. initial temperature T and coefficient α) and step 8, the pseudocode of PSO + GA + SA algorithm is same with Algorithm 2. At the beginning of PSO + GA + SA, the temperature value and the corresponding mutation probability value are both high. There is a great likelihood of performing the mutation step, which improves the algorithm's global search ability. As the number of iteration increases, the temperature gradually decreases, the corresponding mutation probability value also goes down, and the probability that the particle performs the mutation step is also reduced, so that the estimation result of source terms is easy to converge.

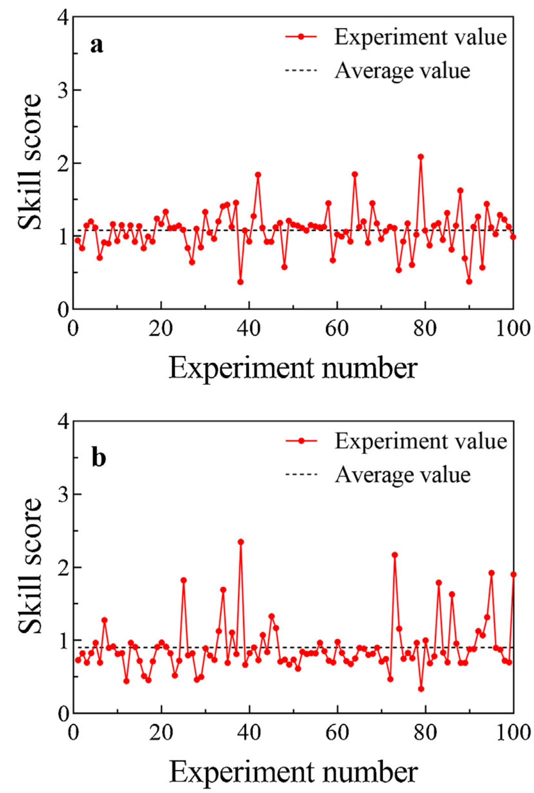

Fig. 10. The Skill score of 100 PSO + GA + SA experiments. (a) is the results of release 17 and (b) is the results of release 22.

As shown in Eq. (4), T represents the temperature in the SA algorithm, and each step is attenuated by the coefficient α ($\alpha < 1$), which in turn leads to a decrease in the probability of mutation.

The initial temperature T is set to 8000 K, the coefficient α is 0.7, and the results of 100 experiments in two cases are shown in Fig. 10. In release 17, the Skill scores of the PSO + GA + SA algorithm are all in the interval of 0.474~1.863, and the average value is 1.091. Compared with PSO + GA and PSO algorithm (Fig. 8(a) and Fig. 9(a)), the values decrease significantly, indicating that the hybrid algorithm is more accurate. The experiment results in release 22 show the similar phenomenon.

4.3. Comparison and analysis of estimation experiment results

The Skill scores of three optimization algorithms are shown in Fig. 8–10. The comparison and analysis of total Skill score indicate the improvement of estimation accuracy for hybrid optimization algorithms. However, total Skill score cannot demonstrate the change of individual Skill score of each source term. Fig. 11 shows the mean Skill score of five source terms in 100 estimation experiments for release 17.

Comparing the individual Skill score of PSO + GA with that of PSO, it can be seen that scores of all source terms drop off obviously. The score values of five source terms decrease by 21.10 %, 23.11 %,

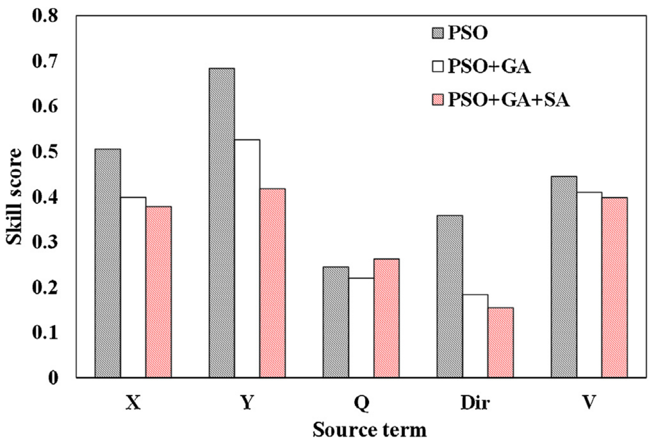

Fig. 11. Skill score of each source term in three optimization algorithms.

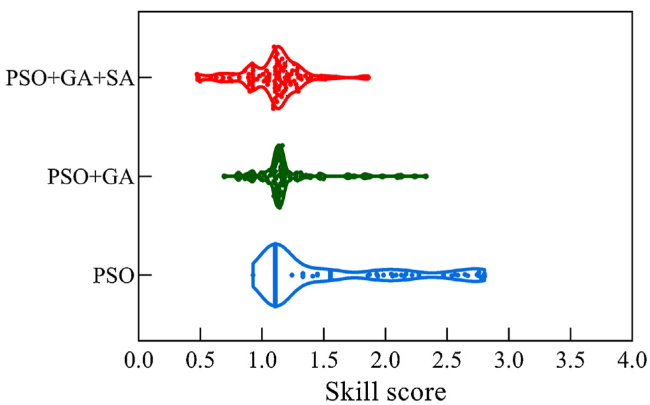

Fig. 12. Skill score of three optimization algorithms.

10.08 %, 48.91 %, and 7.90 %, respectively. As we can see, the Skill score of other terms in PSO + GA + SA all decrease in contrast with those in PSO + GA except for *Q*. These values are declined by 5.12 %, 20.64 %, 15.79 %, and 2.88 %, respectively. The Skill score of *Q* in PSO + GA + SA increases by 19.27 % than the score in PSO + GA, and is also a little higher than the score in PSO. As mentioned before, the estimation results are more accurate with lower Skill score on source term. In most occasions, the estimation accuracy of every source term increases as the combination of algorithms.

Fig. 12 shows the results of source terms estimation experiments of three algorithms. In the violin diagram, each dot represents an experiment result. The width of the violin outline indicates the number of experiment results within the range. The violin outline will be thicker when more experiment results fall within this Skill score range. The analysis results of Skill score are shown in Table 3.

The median and 25 % percentile Skill score of the PSO are both 1.108. In addition, there are many results whose Skill scores are high. These phenomena indicate that the accuracy of the estimation is not accurate enough and the algorithm is easy to fall into local optimum. After combining with the GA, in addition to the median (1.141), the mean (1.213), the 25 % percentile (1.099), the 75 % percentile (1.196), the minimum (0.692) and the maximum value (2.331) of the PSO + GA algorithm experimental results are lower than the mean (1.537), 25 % percentile (1.108), 75 % percentile (1.974), minimum value (0.928), and maximum value (2.806) corresponding to the PSO algorithm experimental results, indicating that the PSO + GA algorithm is more accurate than the PSO algorithm. Moreover, from the outline of the graph, the difference between the 75 % percentile and the 25 % percentile of the PSO + GA algorithm is 0.097, which is much smaller than 0.866 of the PSO algorithm, demonstrating that the results of the PSO + GA algorithm are more concentrated and stable. Compared with the results of above two algorithms, the mean of the scores (1.091), the median (1.120), the 25 % percentile (0.928), the 75 % percentile (1.202), the minimum (0.474) and maximum (1.863) of the PSO + GA + SA algorithm are all smaller, revealing that the accuracy of the algorithm has been further improved. In addition, the difference between the 75 % percentile and the 25 % percentile of the PSO + GA + SA algorithm is 0.274, which is slightly larger than 0.097 of PSO + GA algorithm and less than 0.866 of PSO algorithm. The difference between the maximum and minimum is 1.389, which is less than 1.639 of PSO + GA algorithm and 1.878 of PSO algorithm. These results illustrate that the Skill scores of PSO + GA + SA algorithm are more concentrated overall.

Table 4
Time cost of three optimization algorithms.

Algorithm	PSO	PSO + GA	PSO + GA + SA
Time(s)	571.05	924.07	949.82

Table 4 shows the time-cost of different algorithms on source terms estimation experiments. The processing dataset and the hardware condition (DESKTOP-6A30060, 3.10 GHz Intel(R) Core(TM) i7−8809 G, 16GB) are same. The iteration number and particle number are set as Table 2, and the experiment of each algorithm repeats 10 times. From the results, the time-cost of PSO + GA algorithm is increased by 61.8 % compared with PSO algorithm. The main reason is that when the crossover, mutation and selection steps are introduced, each generation of population needs to be crossed to generate 50 progeny particles. So there are 150 particles in the new population to perform a mutation step. After selection step, 100 particles are selected as the new generation population finally. Since the number of particles calculated by the PSO + GA algorithm is increased by 50 % compared with the PSO algorithm actually, the time-cost has increased greatly. Compared with the PSO + GA algorithm, the time-cost of the PSO + GA + SA algorithm is only increased by 2.8 %, which is not obvious. According to the results of source term estimation experiments, the run-time complexity of three optimization algorithms are O(*MN*), O(*1.5 MN*) and O(*1.5 MN*), respectively. *M* represents the Iteration number of the algorithms, while *N* represents the Particle number in each iteration. In another word, the PSO + GA + SA hybrid algorithm does not raise the calculation time rapidly when improving the calculation accuracy obviously.

5. Discussion

Fig. 8(a), Fig. 9(a) and Fig. 10(a) show the Skill score values obtained by running the three optimization algorithms 100 times in release 17. A larger Skill score indicates a more inaccurate estimation result. As we can see in Fig. 8(a), there are many Skill scores of points are higher than 2. Even more, 54 points have same Skill score. These points are due to the fact that the PSO algorithm falls into local optimum during the calculation process, and cannot continue to search for a more accurate estimation result. In Fig. 9(a), after the introduction of the GA, this problem is settled in some extent. The average value of Skill score in Fig. 9(a) is lower than that in Fig. 8(a), showing the PSO + GA hybrid algorithm behaves better in global search. However, because of the randomness of the PSO + GA algorithm, there are still many Skill scores have high values, indicating the PSO + GA algorithm is not stable enough. It can be seen from Fig. 10(a) that the number of Skill scores bigger than 2 is smaller than Fig. 9(a) and Fig. 8(a). Moreover, the average value also decreases. These results indicate the hybrid algorithm is robust. Fig. 8 (b), Fig. 9 (b) and Fig. 10 (b) corresponding to release

Table 3
Analysis results of Skill score of three optimization algorithms.

	Mean	Median	25 % percentile	75 % percentile	Minimum	Maximum
PSO	1.537	1.108	1.108	1.974	0.928	2.806
PSO + GA	1.213	1.141	1.099	1.196	0.692	2.331
PSO + GA + SA	1.091	1.120	0.928	1.202	0.474	1.863

22 also show similar phenomenon, wherein the average value and the number of points with large Skill score are both decreasing. In addition, in the horizontal comparison of the two plots (a) and (b) in Fig. 8~10, the average value of Skill score in (b) is slightly lower than that in (a). These phenomena indicate that the method can be applied to the scenario of release 22 with higher estimation accuracy.

As mentioned before, the accuracy of source term estimation often depends on the accuracy of the forward dispersion model (Wang et al., 2018). In this paper, through a large number of experiments, BPNN with 2 hidden layers is used as the forward dispersion model. However, from the comparison between the Skill score in two scenarios, it can be seen from various experiment scenarios that the difference in prediction accuracy of the BPNN model will lead to the change of calculation results eventually. Therefore, improving the prediction accuracy of the neural network model in various scenes is especially important for the source term estimation problem. Most existing neural network models attempt to enhance model expression by increasing model complexity, so it may be useful to try BPNN with more hidden layers. However, the complex network structure will lead to a long time-cost in computation. Due to the existence of massive iteration steps in the algorithm, when adopting the network for inference, a small increase of time-cost in one iteration step will prolong the overall calculation time of the algorithm obviously. In recent years, there have been many studies focusing on the pruning of complex structures of neural networks for calculation speedup. Networks like MobileNet (Chen and Su, 2017) and ShuffleNet (Zhang et al., 2017) are used to reduce the network parameters by designing elaborate convolution structures while maintaining accuracy. Next study is trying to apply these neural networks to model forward atmospheric dispersion process.

In this paper, to investigate the feasibility of using hybrid strategy in the realm of source term estimation, three most commonly used optimization algorithms (PSO, GA and SA) are combined together based on the hybrid strategy. However, some key parameters in the hybrid algorithms has a significant impact on the parameter estimation results, such as crossover probability and mutation probability in PSO + GA algorithm, initial temperature and annealing formula in PSO + GA + SA algorithm. Although the values of these parameters in this paper are a high-quality result after abundant attempts, there may exist other values which can make the corresponding hybrid algorithms have better performance in source term estimation. How to determine the optimal values of key parameters in an algorithm under specific metrics quickly and automatically is also a problem worth studying. In addition to the PSO, GA and SA applied in this paper, there are many other optimization algorithms, such as Ant Colony Optimization algorithm, Artificial Bee Colony algorithm, Differential Evolution algorithm. Studying these algorithms and combining them will be very suitable for solving the problem of source terms estimation.

6. Conclusion

In order to solve the source term estimation problem of hazardous gas emission accidents, this paper used a hybrid strategy as the backward estimation method with a state-of-the-art BPNN as the gas forward dispersion model. In terms of the BPNN-based forward model, the neuron number in hidden layers was finally determined after vast experiments. Trained and tested by mass of actual data, the network shows great performance on prediction accuracy and calculation speed in different scenarios. Aiming at the characteristics of different optimization algorithms, this paper proposed a hybrid strategy to combine PSO algorithm, GA algorithm and SA algorithm step by step. To overcome the shortcomings of PSO algorithm, which is easy to fall into local optimization, the steps of crossover, mutation and selection are adopted to improve the global search ability of the algorithm. Because of the randomness of PSO + GA hybrid algorithm, the SA algorithm was combined to control the variation of the mutation probability. The comparison of Skill score among three algorithms indicates that the PSO + GA + SA hybrid algorithm has higher calculation accuracy with acceptable computational efficiency. In the end, the hybrid strategy proposed in this paper has a positive impact in practice. Under the guidance of the hybrid strategy, combining different optimization algorithms, the hybrid algorithm shows better performance in calculation accuracy and computational efficiency for hazardous gas source term estimation, which has important implications to hazard emergency management.

Future researches can focus on studying optimal values of parameter combinations, leading to better algorithm performance in source term estimation. Moreover, in another direction, we can focus on designing a generic framework of hybrid strategy for combining different algorithms. Meanwhile, putting the hybrid algorithm presented in this paper into application can protect the environment form the threat of hazardous gas.

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.

Acknowledgments

This study is supported by National Key Research & Development (R&D) Plan under Grant No. 2018YFC0806900 and the National Natural Science Foundation of China under Grant Nos. 71673292, 21808181, 61673388, 71673294 and National Social Science Foundation of China under Grant No.17CGL047 and Guangdong Key Laboratory for Big Data Analysis and Simulation of Public Opinion.

References

Angeline, P.J., Saunders, G.M., Pollack, J.B., 1994. 'An evolutionary algorithm that constructs recurrent neural networks'. IEEE Trans. Neural Netw. 5, 54–65.
Barad, Morton L., 1958. Project prairie grass, a field program in diffusion. AIR FORCE CAMBRIDGE RESEARCH LABS HANSCOM AFB MA, Vol. 1.
Bhushan, B., Pillai, S.S., 2013. Particle swarm optimization and firefly algorithm: performance analysis. Advance Computing Conference.
Boeringer, D.W., Werner, D.H., 2004. 'Particle swarm optimization versus genetic algorithms for phased array synthesis'. IEEE Trans. Antennas Propag. 52, 771–779.
Boznar, Marija, Lesjak, Martin, Mlakar, Primoz, 1993. A neural network-based method for short-term predictions of ambient SO2 concentrations in highly polluted industrial areas of complex terrain. Atmospheric Environment Part B Urban Atmosphere 27, 221–230.
Chen, Hong Yen, Su, Chung Yen, 2017. 'The Enhanced Hybrid MobileNet'.

Chen, Bin, Wang, Yiduo, Wang, Rongxiao, Zhu, Zhengqiu, Ma, Liang, Qiu, Xiaogang, Weihui, Dai., 2020. 'The Gray-Box Based Modeling Approach Integrating Both Mechanism-Model and Data-Model: The Case of Atmospheric Contaminant Dispersion'. Symmetry 12, 254.

Chu, Shu Chuan, Huang, Hsiang Cheh, Roddick, John F., Pan, Jeng Shyang, 2011. Overview of algorithms for swarm intelligence. International Conference on Computational Collective Intelligence: Technologies & Applications.

Debski, Wojciech., 2010. 'Probabilistic inverse theory'. Adv. Geophys. 52, 1–102.

Efthimiou, George C., Kovalets, Ivan V., Venetsanos, Alexandros, Andronopoulos, Spyros, Argyropoulos, Christos D., Kakosimos, Konstantinos, 2017. 'An optimized inverse modelling method for determining the location and strength of a point source releasing airborne material in urban environment'. Atmos. Environ. 170, 118–129.

Fister, Iztok, Fister, Iztok, Yang, Xin-She, Brest, Janez, 2013. 'A comprehensive review of firefly algorithms'. Swarm Evol. Comput. 13, 34–46.

Gibson, Graham, Monk, Stephen, Padgett, Miles, 2002. 'A field-portable, laser-diode spectrometer for the ultra-sensitive detection of hydrocarbon gases'. J. Mod. Opt. 49, 769–776.

Hanna, Steven R., Briggs, Gary A., Hosker Jr., Rayford P., 1982. 'Handbook on Atmospheric Diffusion'. in. National Oceanic and Atmospheric Administration. Atmospheric Turbulence and Diffusion Lab., Oak Ridge, TN, USA.

Hui, Li, Zhang, Jianwen, 2017. 'Fast source term estimation using the PGA-NM hybrid method'. Eng. Appl. Artif. Intell. 62, 68–79.

Keats, Andrew, Yee, Eugene, Lien, Fue Sang, 2007. 'Bayesian inference for source determination with applications to a complex urban environment'. Atmos. Environ. 41, 465–479.

Kennedy, James, Eberhart, Russell, 1995. Particle swarm optimization. Icnn95-International Conference on Neural Networks.

Khlaifi, Anis, Ionescu, Anda, Candau, Yves, 2009. 'Pollution source identification using a coupled diffusion model with a genetic algorithm'. Math. Comput. Simul. 79, 3500–3510.

Kirkpatrick, Scott., 1984. 'Optimization by simulated annealing: quantitative studies'. J. Stat. Phys. 34, 975–986.

Lauret, Pierre, Heymes, Frédéric, Aprin, Laurent, Johannet, Anne, 2016. 'Atmospheric dispersion modeling using artificial neural network based cellular automata'. Environ. Model. Softw. 85, 56–69.

Long, Kerrie J., Haupt, Sue Ellen, Young, George S., 2010. 'Assessing sensitivity of source term estimation'. Atmos. Environ. 44, 1558–1567.

Ma, Denglong, Wang, Simin, Zhang, Zaoxiao, 2014. 'Hybrid algorithm of minimum relative entropy-particle swarm optimization with adjustment parameters for gas source term identification in atmosphere'. Atmos. Environ. 94, 637–646.

Ma, D., Tan, W., Zhang, Z., Hu, J., 2017. 'Parameter identification for continuous point emission source based on Tikhonov regularization method coupled with particle swarm optimization algorithm'. J. Hazard. Mater. 325, 239–250.

Ma, D., Zhang, Z., 2016. 'Contaminant dispersion prediction and source estimation with integrated Gaussian-machine learning network model for point source emission in atmosphere'. J. Hazard. Mater. 311, 237–245.

Metropolis, Nicholas, Rosenbluth, Arianna W., Rosenbluth, Marshall N., Teller, Augusta H., Teller, Edward, 1953. 'Equation of state calculations by fast computing machines'. J. Chem. Phys., 21.

Ni, Lei, Jiang, Juncheng, Yong, Pan., 2013. 'Leak location of pipelines based on transient model and PSO-SVM'. J. Loss Prev. Process Ind. 26, 1085–1093.

Pelliccioni, A., Tirabassi, T., 2006. 'Air dispersion model and neural network: a new perspective for integrated models in the simulation of complex situations'. Environ. Model. Softw. 21, 539–546.

Qiu, Sihang, Chen, Bin, Wang, Rongxiao, Zhu, Zhengqiu, Wang, Yuan, Xiaogang, Qiu., 2018. 'Atmospheric dispersion prediction and source estimation of hazardous gas using artificial neural network, particle swarm optimization and expectation maximization'. Atmos. Environ. 178, 158–163.

Sheppard, P.A., 1962. 'Atmospheric Diffusion'. Nature 193, 1117–1118.

So, Won, Koo, Jamin, Shin, Dongil, En, Sup Yoon, 2010. 'The estimation of hazardous gas release rate using optical sensor and neural network. 20th European Symposium on Computer Aided Process Engineering.

Thomas, J.Joshua, Pinar, Karagoz, Bazeer Ahamed, B., Pandian, Vasant (Eds.), 2020. Deep Learning Techniques and Optimization Strategies in Big Data Analytics. IGI Global, Hershey, PA, USA.

Thomson, Laura C., Hirst, Bill, Gibson, Graham, Gillespie, Steve, Jonathan, Philip, Skeldon, Kenneth D., Padgett, Miles J., 2007. 'An improved algorithm for locating a gas source using inverse methods'. Atmos. Environ. 41, 1128–1134.

Tsuruda, Takashi., 2013. 'Nuclear power plant explosions at fukushima-daiichi'. Procedia Eng. 62, 71–77.

Valeriy, Kharchenko, Pandian, Vasant (Eds.), 2018. Handbook of Research on Renewable Energy and Electric Resources for Sustainable Rural Development. IGI Global, Hershey, PA, USA.

Valeriy, Kharchenko, Pandian, Vasant (Eds.), 2020a. Handbook of Research on Energy-Saving Technologies for Environmentally-Friendly Agricultural Development. IGI Global, Hershey, PA, USA.

Valeriy, Kharchenko, Pandian, Vasant (Eds.), 2020b. Handbook of Research on Smart Computing for Renewable Energy and Agro-Engineering. IGI Global, Hershey, PA, USA.

Vasant, Pandian, Zelinka, Ivan, Weber, Gerhard-Wilhelm (Eds.), 2019. Intelligent Computing & Optimization. Springer International Publishing, Switzerland.

Vasant, Pandian, Zelinka, Ivan, Weber, Gerhard-Wilhelm (Eds.), 2020. Intelligent Computing and Optimization. Springer International Publishing, Switzerland.

Wang, R., Chen, B., Qiu, S., Zhu, Z., Wang, Y., Wang, Y., Qiu, X., 2018. 'Comparison of machine learning models for hazardous gas dispersion prediction in field cases'. Int. J. Environ. Res. Public Health, 15.

Wilson, John D., Sawford, Brian L., 1996. 'ReView of Lagrangian stochastic models for trajectories in the turbulent atmosphere'. Boundary. Meteorol. 78, 191–210.

Xue, F., Kikumoto, H., Li, X., Ooka, R., 2018. 'Bayesian source term estimation of atmospheric releases in urban areas using LES approach'. J. Hazard. Mater. 349, 68–78.

Yi, Liu, Shunjiang, Ni, Wenguo, Weng, 2017. 'Development of the public safety system and a security-guaranteed society'. Strategic Study of Chinese Academy of Engineering 19, 118–123.

Zhang, Xiangyu, Zhou, Xinyu, Lin, Mengxiao, Jian, Sun, 2017. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices.

Zhu, Zhengqiu, Chen, Bin, Qiu, Sihang, Wang, Rongxiao, Wang, Yiping, Ma, Liang, Xiaogang, Qiu., 2018. A data-driven approach for optimal design of integrated air quality monitoring network in a chemical cluster. R. Soc. Open Sci.