

REGULAR PAPER

Departure flight delay prediction due to ground delay program using Multilayer Perceptron with improved sparrow search algorithm

X. Dong , X. Zhu  and J. Zhang 

State Key Laboratory of Air Traffic Management System, Nanjing University of Aeronautics and Astronautics, College of Civil Aviation, Nanjing, China

Corresponding author: X. Dong; Email: dongxiangning@nuaa.edu.cn

Received: 10 May 2023; **Revised:** 10 July 2023; **Accepted:** 22 August 2023

Keywords: ground delay programs; departure flight delay; improved sparrow search algorithm; machine learning

Abstract

The ground delay program (GDP) is a commonly used tool in air traffic management. Developing a departure flight delay prediction model based on GDP can aid airlines and control authorities in better flight planning and adjusting air traffic control strategies. A model that combines the improved sparrow search algorithm (ISSA) and Multilayer Perceptron (MLP) has been proposed to minimise prediction errors. The ISSA uses tent chaotic mapping, dynamic adaptive weights, and Levy flight strategy to enhance the algorithm's accuracy for the sparrow search algorithm (SSA). The MLP model's hyperparameters are optimised using the ISSA to improve the model's prediction accuracy and generalisation performance. Experiments were performed using actual GDP-generated departure flight delay data and compared with other machine learning techniques and optimisation algorithms. The results of the experiments show that the mean absolute error (MAE) and root mean square error (RMSE) of the ISSA-MLP model are 16.8 and 24.2, respectively. These values are 5.61%, 6.3% and 1.8% higher in MAE and 4.4%, 5.1% and 2.5% higher in RMSE compared to SSA, particle swarm optimisation (PSO) and grey wolf optimisation (GWO). The ISSA-MLP model has been verified to have good predictive and practical value.

Nomenclature

<i>STD</i>	standard deviation
<i>GDP</i>	ground delay program
<i>TFM</i>	traffic flow management
<i>ETD</i>	estimated time of departure
<i>CTD</i>	controlled time of departure
<i>ATC</i>	air traffic control
<i>TAF</i>	terminal aerodrome forecasts
<i>METAR</i>	meteorological aerodrome report
<i>SVM</i>	support vector machine
<i>ELM</i>	extreme learning machine
<i>XGboost</i>	extreme gradient boosting tree
<i>AAR</i>	arrival aircraft rate
<i>MARL</i>	multi-agent reinforcement learning
<i>DQN</i>	Q-learning network
<i>RMSE</i>	root mean square error
<i>MAE</i>	mean absolute error

<i>SSA</i>	sparrow search algorithm
<i>ISSA</i>	improved sparrow search algorithm
<i>MLP</i>	Multilayer Perceptrona
<i>GWO</i>	grey wolf optimisation
<i>PSO</i>	particle swarm optimisation
<i>SHAP</i>	Shapley additive explanations

Greek symbol

α	random number between (0, 1]
β	normally distributed random number obeying a mean of 0 and a variance of 1
μ, ν	normal distribution
τ	constant value of 1.5
ε	very small constants

1.0 Introduction

In contemporary times, as the aviation industry steadily expands, flight delays have emerged as a primary concern impacting both passenger travel experience and airline operational efficiency. In unfavourable weather or airspace congestion scenarios, air traffic controllers typically implement ground delay programs (GDP), which are among the principal contributors to flight delays [1, 2]. Thus, investigating the ramifications of GDP on departure flight delays can furnish significant insights to the concerned authorities, enabling them to formulate more effective strategies for airport flight management.

Several research endeavors have explored GDP from various angles, encompassing modeling simulation techniques and machine learning approaches. The primary objective of the modeling simulation methodology is to mitigate aviation disruptions [3] by employing an integer planning model to determine the optimal amalgamation of ground waiting and rerouting strategies. Avijit Mukherjee et al. [4] introduced an algorithm for ascertaining flight departure delays using probabilistic airport capacity forecasts, which employs a static stochastic ground waiting model to optimise the number of scheduled incoming flights in multiple stages. This approach is simpler to implement than earlier stochastic dynamic optimisation models and presents a fresh perspective on the ground waiting issue. Yan et al. [5] formulated a comprehensive platform for simulating flight operations during GDP and proposed an algorithm to resolve the route recovery problem in such scenarios. Jacquillat [6] introduced a new passenger-centric approach ground delay programs (GDP-PAX) optimisation approach that devised a large-scale integer optimisation model and reported considerable reductions in passenger delays at a slight increase in flight delay costs. Liu et al. [7] proposed a framework for joint optimisation of GDP parameters under uncertain airport capacity, resulting in significant reduction of delay times and improvement of operational efficiency, while maintaining the acceptable level of air traffic control (ATC) safety risk. These models all focus on different perspectives in solving the GDP problem by adding more constraints. The advantage is that optimal solutions can be obtained under certain conditions, but for large-scale integer planning problems, planning methods are often very time-consuming. They cannot even be solved in a finite amount of time, making it difficult to meet real-time traffic management needs.

In light of the escalating data storage capacity and enhanced computational capabilities witnessed in recent years, research endeavors have increasingly harnessed the potential of data mining techniques. This development has, in turn, stimulated investigations into the conundrum of ground delay programs employing machine learning methodologies [8]. Prominent research methods that are commonly utilised include Bayesian networks, neural networks, support vector machines, reinforcement learning and random forests [9, 10]. Smith et al. [11] implemented support vector machine (SVM) algorithms and terminal forecast (TAF) data to predict the arrival aircraft rate (AAR), which was employed to determine the duration of GDP. Mangortey [12] evaluated a range of machine learning algorithms to combine fused weather and flight time data and identified that the random forest model was the most

precise model for prediction. Liu et al. [13] utilised SVM to examine the relationship between convective weather and GDP to obtain a score, analysed multiple airports to create a GDP duration prediction model, and compared the best prediction results among seven regression models, reporting that the elastic network model was the best. Chen et al. [14] applied multi-agent reinforcement learning (MARL) to simulate the use of GDP to address demand and capacity balancing issues in high-density situations in the pre-tactical phase. The MARL approach utilising a double Q-learning network (DQN) has the potential to significantly reduce the number of delayed flights and the average delay duration. Dong et al. [15] conducted a preliminary analysis of flight delay prediction due to GDP and concluded that the best predictions were achieved using a decision tree model. Yu et al. [16] proposed a novel deep belief network approach to mine the intrinsic patterns of flight delays, proposed an effective flight delay prediction model, and fused SVM with DBN to implement supervised fine-tuning within the prediction architecture. Micha Zoutendijk et al. [17] used two probabilistic prediction algorithms, hybrid density networks and random forest regression to forecast individual flight delays, both of which estimated the delay distribution of arriving and departing flights well, with an average absolute error of fewer than 15 minutes. Ehsan Esmailzadeh et al. [18] used an SVM model to explore the non-linear relationship between flight delay outcomes, and the results showed that pushback delay was the most crucial cause. Mokhtarimousavi et al. [19] used two methods to investigate the relationship between significant variables and flight delays, firstly by a random parameter logit model to underly usingstand the potential significant variables of flight delays and then by an SVM model trained by the artificial bee colony (ABC) algorithm to explore the non-linear relationship between flight delay outcomes and causes.

To summarise, while operational research methods can identify the optimal timing or geographic range for GDP release [7], they do not provide information on when delays are likely to happen, which is essential for airlines and passengers. Furthermore, most existing research has focused on GDP-related feature processing, with little attention given to machine learning models and their parameters, leading to poor generalisation of the models [20]. Therefore, this study further investigates the flight delays that will be generated under the implementation of GDP, and proposes a departure flight delay prediction model based on MLP and an ISSA for hyperparameter search. The model is tested using actual departure flight delay data, and the results show that the ISSA-MLP model can accurately predict delays caused by GDP, thus demonstrating its reliability and stability.

2.0 Improved sparrow search algorithm

2.1 Sparrow search algorithm

The SSA is a recently developed swarm intelligence optimisation technique proposed by Xue et al. [21] in 2020. The algorithm models the foraging and anti-predatory behaviour of sparrows and categorises them into three types: finders, joiners and detectors, each with their corresponding behavioural rules. The algorithm assumes that the search space has d dimensions and there are n sparrows in the swarm.

The position of the finder is updated as shown in Equation (1):

$$X_{ij}^{t+1} = \begin{cases} X_{ij}^t \cdot \exp\left(\frac{-i}{\alpha \cdot T}\right), R_2 < ST \\ X_{ij}^t + Q \cdot L, R_2 \geq ST \end{cases} \quad (1)$$

where t is the current number of iterations; T is the maximum number of iterations; X_{ij}^t is the current sparrow position; α is a random number between (0, 1]; Q is a random number that fits the normal distribution; L is the unit matrix of $1 \times d$; R_2 is the alert value; and ST is the safety value.

The position of the joiner is updated as shown in Equation (2):

$$X_{ij}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{\text{worst}} - X_{ij}^t}{i^2}\right), & i > \frac{n}{2} \\ X_p^{t+1} + |X_{ij}^t - X_p^{t+1}| \cdot A^+ \cdot L, & \text{otherwise} \end{cases} \quad (2)$$

where X_{worst} denotes the global worst position; X_p denotes the best position occupied by the finder; A denotes a matrix of $1 \times d$ where each element is randomly assigned a value of 1 or -1 and $A^+ = A^T(AA^T)^{-1}$.

The detectors make up 10%–20% of the total number of sparrows and their position is updated as shown in Equation (3):

$$X_{ij}^{t+1} = \begin{cases} X_{\text{best}}^t + \beta \cdot |X_{ij}^t - X_{\text{best}}^t|, f_i > f_g \\ X_{ij}^t + K \left(\frac{|X_{ij}^t - X_{\text{worst}}^t|}{(f_i - f_w) + \varepsilon} \right), f_i = f_g \end{cases} \quad (3)$$

where X_{best}^t denotes the current global optimum position; β is the step control parameter for a normally distributed random number obeying a mean of 0 and a variance of 1; K is a random number between -1 and 1; f_i denotes the current sparrow’s fitness value; f_g and f_w denote the current global optimum fitness and worst fitness, respectively; and ε is the smallest constant to avoid zero in the denominator.

2.2 Improved approach for sparrow search algorithm

The limitation of the SSA algorithm in tackling complex optimisation problems is its low convergence accuracy and the tendency to converge to a local optimum. To counteract these drawbacks, ISSA was introduced to enhance the global optimisation capabilities of the algorithm. Three key improvements were made to the SSA algorithm: (1) the initial population was optimised using the tent chaotic mapping; (2) dynamic adaptive weights were integrated into the finder to balance the algorithm’s global search and local exploitation ability; (3) the detector was enhanced with a Levy flight strategy to disrupt the current optimal solution and reinforce the local search ability.

2.2.1 Tent chaos map

Chaotic mapping is a term used to describe the random motion that arises from a deterministic equation. This motion exhibits both periodicity and inherent randomness within a phase space. By introducing chaotic mapping to an algorithm, the diversity of the initial population can be increased, leading to improved optimisation capabilities. Compared to logistic chaos, the tent chaotic map demonstrates more pronounced chaotic features [22]. Therefore, the tent chaotic map was utilised in this study to initialise the population, generating an initial population with high diversity. The equation for the tent chaotic map is provided in Equation (4):

$$y_{i+1} = \begin{cases} \frac{y_i}{a}, 0 \leq y_i \leq a \\ \frac{1 - y_i}{1 - a}, a < y_i \leq 1 \end{cases} \quad (4)$$

where $a \in (0, 1)$; i is the number of iterations.

In the tent mapping formula, a is generally taken to be 0.5. The tent mapping is used to generate a chaotic sequence matrix to initialise the sparrow population, as shown in Equation (5):

$$X_i = X_{lb} + (X_{lb} - X_{ub}) \times y_i \quad (5)$$

where X_{lb} and X_{ub} are the upper and lower bounds for each individual in each dimension and X_i is the mapped individual.

2.2.2 Dynamic adaptive weights

This study introduces dynamic adaptive weights for the finder to enhance the global search capability and convergence rate of SSA. This is done to mitigate the likelihood of an individual falling into a locally

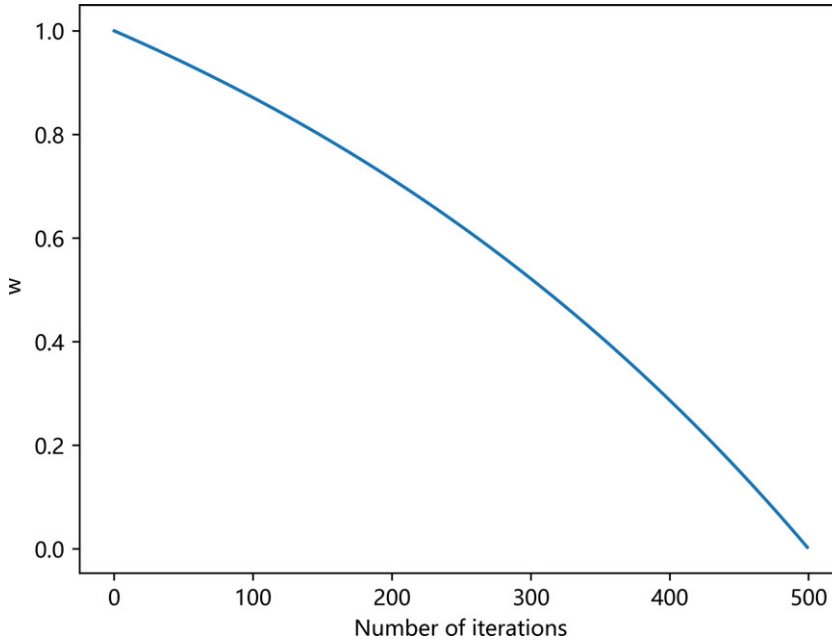


Figure 1. Dynamic adaptive weight change curves.

optimal solution due to insufficient search capability when it is near the optimal solution. The dynamic adaptive weights are represented mathematically as w and are illustrated in Equation (6):

$$X_{ij}^{t+1} = \begin{cases} X_{ij}^t \cdot w \cdot \exp\left(\frac{-i}{\alpha \cdot T}\right), R_2 < ST \\ X_{ij}^t + Q \cdot L, R_2 \geq ST \end{cases} \tag{6}$$

The dynamic adaptive weights w equation is shown in Equation (7):

$$w = 1 - \frac{e^{(t/T_{\max})} - 1}{e - 1} \tag{7}$$

where t is the current number of iterations and T_{\max} is the maximum number of iterations.

Implementing dynamic adaptive weights enables adaptive regulation of the finder’s position. As depicted in Fig. 1, higher initial values of w correspond to a larger search range for the algorithm. Towards the end of the iteration, lower values of w facilitate the algorithm’s local development.

2.2.3 Levy flight strategy

Levy flight [23] is a unique random wandering model that characterises motion in terms of random step lengths and directions. In Levy flight, these step lengths and directions are random and possess a long-tailed distribution, occasionally including extreme values. Adding the Levy flight strategy to the detector can now perturb the optimal solution and prevent getting trapped in a locally optimal solution. The revised formulation of the detector is given by Equation (8):

$$X_{ij}^{t+1} = \begin{cases} Levy(\lambda) \cdot X_{best}^t + \beta \cdot |X_{ij}^t - Levy(\lambda) \cdot X_{best}^t|, f_i > f_g \\ X_{ij}^t + K \left(\frac{|X_{ij}^t - X_{worst}^t|}{(f_i - f_w) + \varepsilon} \right), f_i = f_g \end{cases} \tag{8}$$

Algorithm 1 The framework of the ISSA.

Initialisation:

Maximum number of iterations is T

Number of finders is PD

Number of detectors is SD

The alert value is R_2

Number of sparrows is n

Initialise the position of n sparrows according to Equation (5)

Iteration:

- 1: **while** ($t < T$)
 - 2: Rank the fitness values to identify the current best and worst individuals
 - 3: $R_2 = \text{rand}(1)$
 - 4: **for** $i = 1 : PD$
 - 5: Update the position of the finder using the Equation (6)
 - 6: **end for**
 - 7: **for** $i = (PD + 1) : n$
 - 8: Update the position of the joiner using the Equation (2)
 - 9: **end for**
 - 10: **for** $i = 1 : SD$
 - 11: Update the position of the detector using the Equation (8)
 - 12: **end for**
 - 13: Get current update position
 - 14: If the new position is better than the previous one, update it
 - 15: $t = t + 1$
 - 16: **end while**
 - 17: Output the best solution
-

The symbol $Levy(\lambda)$ denotes the random number that follows the Levy distribution. To calculate $Levy(\lambda)$, the Mantegna algorithm can be used, as shown in Equation (9).

$$S = \frac{\mu}{|v|^{1/\tau}} \quad (9)$$

where s is the Levy flight stage length; $\tau = 1.5$; μ and v obey a normal distribution, $\mu \sim N(0, \sigma_\mu^2)$, $v \sim N(0, \sigma_v^2)$, $\sigma_v^2 = 1$; σ_μ^2 can be calculated from Equation (10).

$$\sigma_\mu = \left\{ \frac{\Gamma(1 + \tau) \cdot \sin(\pi \cdot \tau/2)}{\Gamma[(1 + \tau)/2] \cdot \tau \cdot 2^{(\tau-1)/2}} \right\} \quad (10)$$

where Γ is the gamma function.

The detailed steps of ISSA are presented in pseudocode form in Algorithm 1.

2.3 Benchmark function experiments

To evaluate the effectiveness of ISSA in solving the optimisation problems presented in this paper, six test functions were simulated and are listed in Table 1. Among them, F_1 and F_2 were unimodal functions, F_3 and F_4 were multimodal functions, and F_5 and F_6 were fixed-dimensional functions. The performance of ISSA was compared with that of standard SSA, GWO and PSO using a population size of 30 and 300 iterations for all algorithms.

Table 1. Benchmarking functions

Function	Dimension	Rank	Best value
$F_1(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	[-10,10]	0
$F_2(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)$	30	[-100,100]	0
$F_3(x) = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	[-5.12,5.12]	0
$F_4(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	[-600,600]	0
$F_5(x) = 4x_1^2 + 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[-5,5]	-1.0316
$F_6(x) = -\left(e^{\left 100 - \frac{\sqrt{x_1^2 + x_2^2}}{\pi} \right } \left \sin(x_1) \sin(x_2) \right + 1 \right)^{-0.1}$	2	[-10,10]	-1

Table 2. Algorithm parameter setting

Algorithm	Parameter
ISSA	$PD = 70\%$; $SD = 20\%$; $R_2 = 0.8$
SSA	$PD = 70\%$; $SD = 20\%$; $R_2 = 0.8$
GWO	a decreases linearly from [2, 0]
PSO	$w = 0.8$; $C_1 = 0.5$; $C_2 = 0.5$

The simulation experiments were conducted in Python 3.9. The parameters of the four algorithms were set as presented in Table 2, and 30 independent experiments were performed for each function to eliminate any chance bias. The optimal values, average values and standard deviation (STD) of the results are presented in Table 3. To better visualise the optimisation performance of the algorithms, the convergence curves of ISSA, SSA, GWO and PSO for the six tested functions are included in Fig. 2.

The analysis presented in Table 3 indicates that the ISSA algorithm outperforms other algorithms regarding the optimal solutions for unimodal and multimodal functions. For the unimodal functions, the optimal solutions obtained by the ISSA algorithm are all zero, whereas other algorithms produce larger values. Similarly, the ISSA algorithm's optimal solutions for the multimodal functions are also all zero, while other algorithms produce larger values. This demonstrates that the ISSA algorithm has higher accuracy and more robust global search capability. Regarding fixed dimensional functions, the ISSA algorithm performs comparably to other algorithms, but its mean and standard deviation are relatively small, suggesting better stability. The ISSA algorithm performs well in solving optimisation problems and exhibits significant advantages, especially in unimodal and multimodal functions.

In Fig. 2, the convergence curves for the six test functions are displayed. It is evident that ISSA outperforms the other optimisation algorithms in terms of accuracy and convergence speed, except for F_5 . The superiority of ISSA is not as apparent in F_5 because the test functions are not very intricate, and thus all the algorithms can find optimal solutions. Nonetheless, ISSA is still more efficient in this case.

3.0 ISSA-MLP Model for predicting delays in departing flights due to ground delay program

3.1 Multilayer Perceptron

The Multilayer Perceptron [24] is a feed-forward neural network architecture, as depicted in Fig. 3. Comprising an input layer, multiple hidden layers and an output layer, it acquires the mapping association

Table 3. Benchmark function simulation results

Type	Function	Statistics	ISSA	SSA	GWO	PSO
Unimodal benchmark functions	F_1	Best	0	1.28e-300	8.23e-18	67.04
		Average	2.92e-13	7.44e-06	1.42e-16	113.02
		STD	1.57e-12	2.38e-05	1.65e-16	24.77
	F_2	Best	0	2.95e-134	2.87e-27	6.05×10^5
		Average	1.94e-25	2.40e-09	4.69e-24	7.24×10^5
		STD	7.44e-25	9.19e-09	2.25e-23	3.01×10^4
Multimodal benchmark functions	F_3	Best	0	0	0	116.04
		Average	0	8.36e-12	4.73e-16	228.50
		STD	0	3.11e-11	1.64e-15	46.93
	F_4	Best	0	0	0	90.03
		Average	0	4.62e-13	5.18e-17	289.49
		STD	0	1.77e-12	1.93e-16	111.14
Fixed dimensional functions	F_5	Best	-1.0316	-1.0316	-1.0316	-1.0316
		Average	-1.0310	-1.0309	-1.0267	-1.0316
		STD	0.00068	0.0010	0.01233	4.27e-16
	F_6	Best	-1	-1.0	-1.0	-0.08
		Average	-0.96	-0.07	-0.96	-0.05
		STD	0.14	0.24	0.17	0.03

between the input and output signals by applying a nonlinear transformation to the input signal. The formula for its forward propagation is expressed as Equation (11).

$$h^{(l)} = f(W^{(l)}h^{(l-1)} + b^{(l)}) \tag{11}$$

where $h^{(l)}$ denotes the output of the layer l neuron, $W^{(l)}$ denotes the connection weight of the layer l neuron, and $b^{(l)}$ denotes the bias of the layer l neuron. f denotes the nonlinear activation function.

Every neuron within the MLP is a fundamental processing unit that obtains input from a neuron in the previous layer and produces an output value. The interconnections between these neurons are characterised by connection weights, which may be deemed parameters within the learning model. During MLP training, a backpropagation algorithm is utilised to determine the gradient of each connection weight, thereby reducing the loss function of the training sample. The loss function, represented by Equation (12), is minimised. Theoretically, a neural network can conform to any function [25]. However, the neural network’s architecture significantly affects the model, and the hyperparameters to be adjusted are the number of neurons per layer, the learning rate and the batch_size.

$$L(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n l(y^{(i)}, \hat{y}^{(i)}) \tag{12}$$

where l represents the loss value of each sample, $y^{(i)}$ is the true value and $\hat{y}^{(i)}$ is the predicted value.

3.2 ISSA-MLP prediction model

When creating a neural network model, selecting appropriate hyperparameters is essential as they directly impact the model’s predictive performance. Conventional methods of hyperparameter search, such as grid search or random search, are typically employed to test different combinations of hyperparameters. However, these methods have several limitations, including a vast search space and high

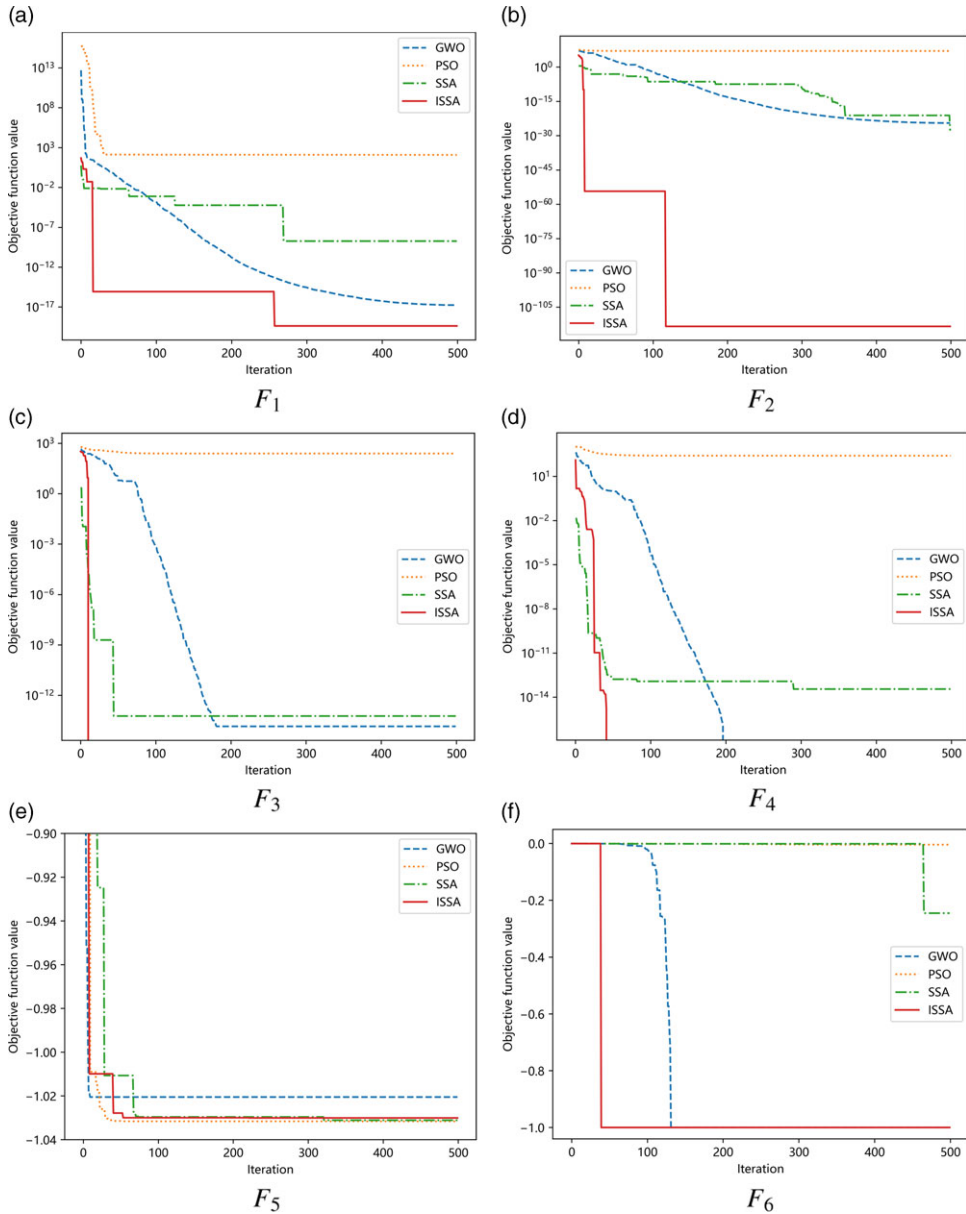


Figure 2. Benchmark function convergence curves.

computational expense. To enhance the prediction accuracy of MLP, the ISSA algorithm is utilised to explore the hyperparameters of the MLP model, which consist of the number of neurons in the first and second layers, the probability of randomly dropping neurons (dropout), and the data block size for batch training (batch_size). The flow diagram for the constructed ISSA-MLP model is presented in Fig. 4.

4.0 GDP Example analysis

4.1 Ground delay programs

The GDP initiation process is presented in Fig. 5. Before GDP commencement, traffic flow management (TFM) issues an initial GDP plan to the affected airport based on GDP operating parameters such as

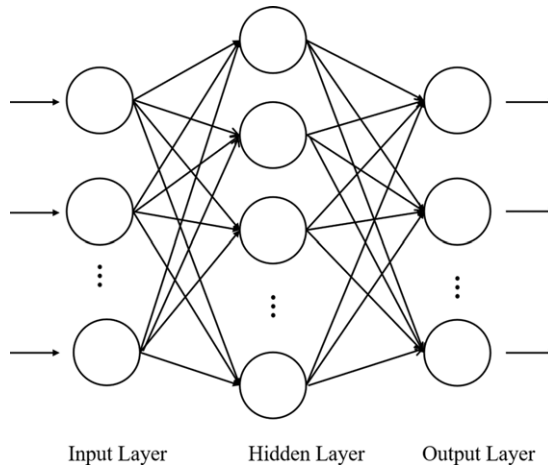


Figure 3. Multilayer Perceptron model.

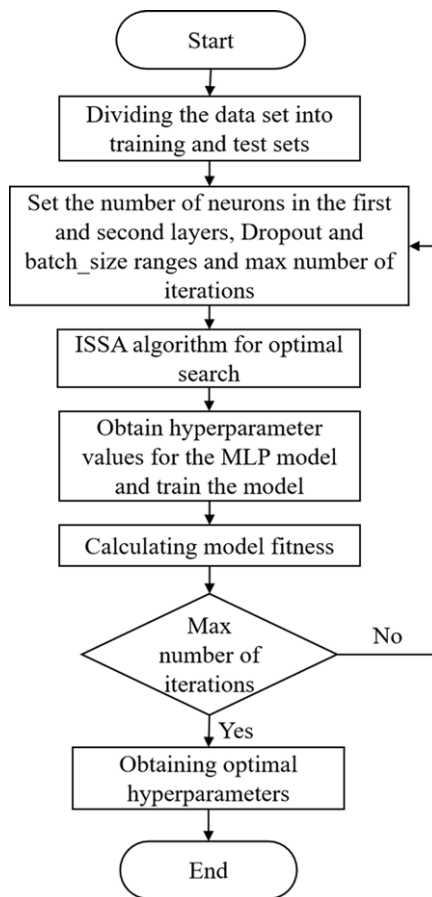
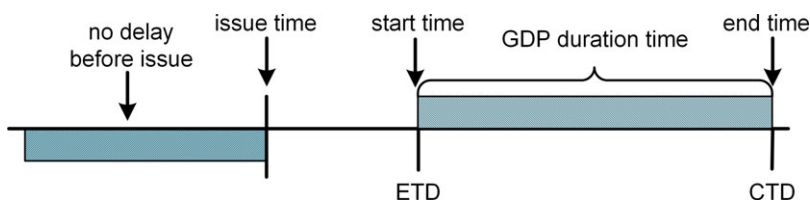


Figure 4. Flow chart of the ISSA-MLP prediction model.

Table 4. Weather and traffic variables description

Category	Attribute name	Attribute description
Weather elements	Visibility	Surface visibility(miles)
	Celling	Cloud height (miles)
	Wind direction	0 if tailwind, 1 otherwise
	Wind magnitude	Surface wind magnitude (m/s)
Traffic features	Scheduled arrivals	Hourly scheduled arrival counts (aircraft/h)
	Scheduled departures	Hourly scheduled departure counts (aircraft/h)
	Actual arrivals	Hourly actual arrival counts (aircraft/h)
	Actual departures	Hourly actual departures counts (aircraft/h)
	Hour	Hour of GDP modeling time
	Month	Month of GDP modeling time

**Figure 5.** GDP occurrence process.

GDP start time, end time and geographical scope. During a GDP, the ground waiting time corresponds to the duration between the estimated time of departure (ETD) and the controlled time of departure (CTD).

There are three primary reasons for delays caused by GDP, the first being airspace congestion. When airspace is congested, flights may need to wait a certain period before entering the take-off and landing sequence. Secondly, weather conditions may result in delayed or canceled take-offs and landings, leading to further delays. Finally, ATC restrictions may lead to postponed take-off and flight landing times.

4.2 Data set

This study used flight operation data and meteorological aerodrome report (METAR) messages obtained from the International Nanjing Lukou Airport (ICAO four-character code: ZSNJ) between January and June 2021. Each flight data included details on the scheduled departure time, actual departure time and planned landing time. Based on the literature [13], the number of scheduled arrival flights and the number of scheduled departure flights were selected as traffic characteristics. Furthermore, considering that the actual take-off and landing flights are also essential factors for GDP implementation, the actual number of arriving and departing flights was also included in the traffic characteristics.

The GDP records hourly are sourced from the logbooks of the Jiangsu ATC branch. These records furnish information on the GDP commencement and cessation times, based on which the instances of departure flight delays during GDP implementation are tabulated. The ultimate data features constructed are explicitly outlined in Table 4. This paper employs a partitioning scheme to allocate 20% of the initial dataset as a distinct test dataset. The remaining data is then divided into training and validation datasets utilising a five-fold cross-validation method.

We acquired 117,467 flight data and tallied the hourly departure flight delays during the GDP implementation period. The delay time distribution is illustrated in Fig. 6, indicating that most delays are within 25 minutes.

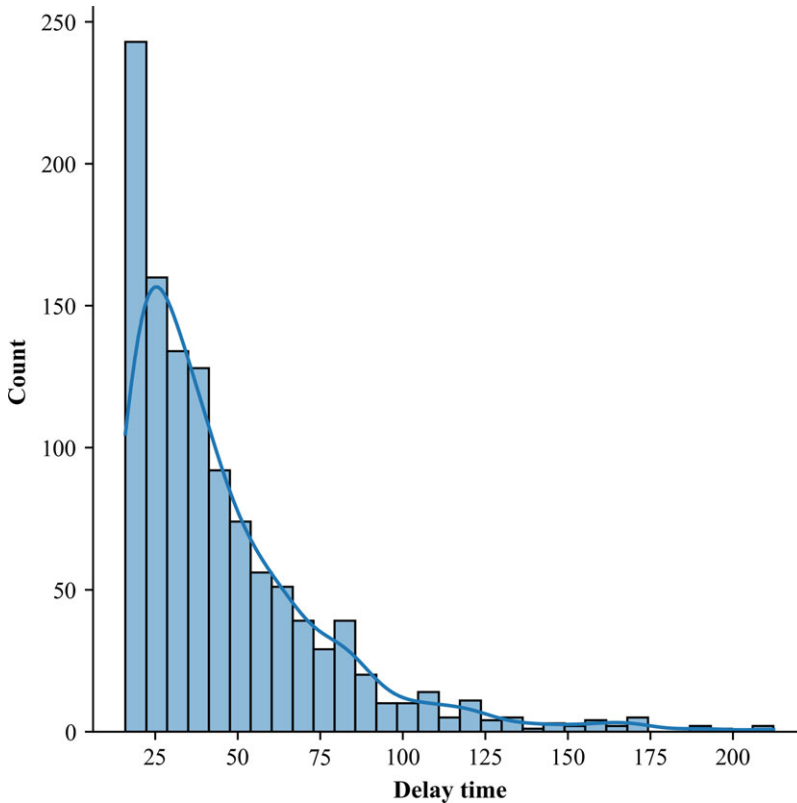


Figure 6. Delay distribution.

4.3 Data standardisation

Table 4 presents the data features used in this study. However, due to the differences in unit magnitudes among the features and the neural network’s sensitivity to data values, direct analysis of the raw data is not possible. Therefore, we normalised the data using Equation (13) to transform the data into the range of [0,1] during the model training process.

$$x^* = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{13}$$

4.4 Performance measures

In assessing the model’s prediction performance, mean absolute error (MAE) and root mean squared error (RMSE) are used to assess the overall prediction accuracy of the model. The equations for these two metrics are shown in Equations (14) and (15).

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i| \tag{14}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \tag{15}$$

where \hat{y}_i is the predicted value, y_i is the true value and n is the length of the data.

Table 5. Grid search range

Model	Parameter	Parameter Search Range	Steps	Optimal Values
SVM	C	[0,1]	0.1	0.1
	γ	[0,1]	0.01	0.03
	kernel	[linear, poly, rbf, sigmoid]	1	rbf
ELM	layer	[0,50]	10	30
XGBoost	n_estimators	[10,300]	10	30
	γ	[0,10]	0.1	0.4
	learning_rate	[0.05,0.5]	0.01	0.07
	max_depth	[3,30]	3	6
	min_child_weight	[0,10]	1	2
MLP	layer1	[0,200]	10	90
	layer2	[0,200]	10	100
	dropout	[0,1]	0.1	0.5
	batch_size	[0,30]	1	8

Table 6. Prediction performances of different models

Model	MAE		RMSE	
	Train	Test	Train	Test
SVR	20.13	21.65	31.23	32.48
ELM	21.54	22.59	29.65	31.01
XGBoost	15.61	18.95	22.79	27.59
MLP	12.34	18.50	19.97	26.71

5.0 Result and discussion

5.1 Comparison of prediction results of different models

To analyse MLP's performance in predicting departure flight delays, we compared it with other models such as support vector regression (SVR) [26], extreme learning machine (ELM) [24] and extreme gradient boosting tree (XGBoost) [27]. We used MAE and RMSE as the performance evaluation metrics for all models. To ensure fairness, we used grid search [28] to select hyperparameters for all models, and the grid search results are shown in Table 5.

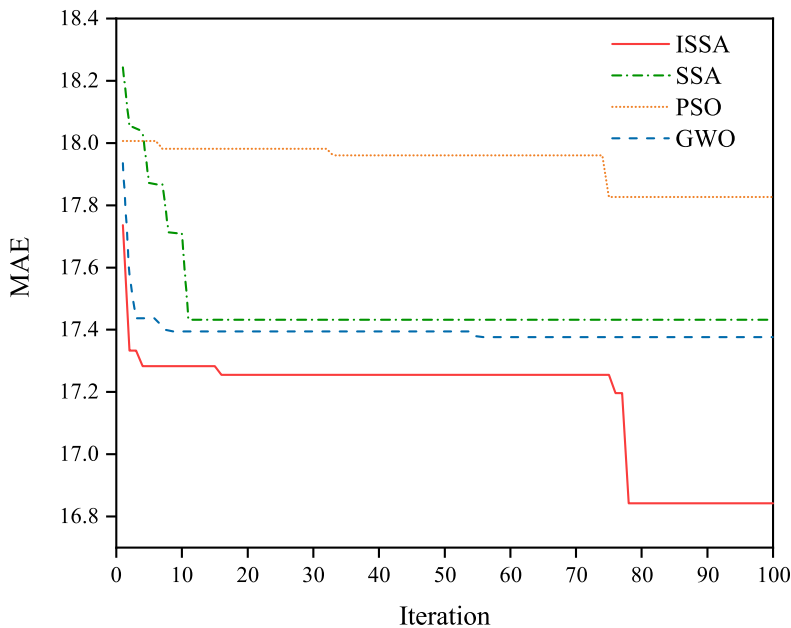
According to the optimal hyperparameters specified in Table 5, the model's training outcomes are depicted in Table 6. Upon examination of Table 6, it becomes evident that the MLP model exhibits superior predictive capabilities compared to the alternative regression models, as evidenced by its minimal MAE and RMSE values on both the training and test datasets. Following the MLP model, XGBoost and ELM models demonstrate relatively good performance. In contrast, the SVR model exhibits inferior results on both the training and test datasets, with higher MAE and RMSE values than the other models. This discrepancy indicates the SVR model's inadequacy in effectively fitting the provided data.

5.2 Comparison of hyperparametric merit search

The Python 3.9.0 programming language was utilised to write the simulation environment in this study, operating under the win10 64-bit system, and implemented the Keras 2.11.0 deep learning framework. The computer had an Intel core 710,700 processor, 2.92 Hz central frequency, and 16GB of RAM. ISSA was utilised to optimise the hyperparameters of the MLP model [29] by defining a search range for each

Table 7. Range of adjustment parameters and results of the optimisation algorithm

Hyperparameter	Rank	ISSA	SSA	GWO	PSO
Number of neurons in the first layer	[0,200]	98	96	162	100
Number of neurons in the second layer	[0,200]	104	80	123	100
Dropout	[0,1]	0.6	0.7	0.8	0.8
Batch_size	[0,30]	6	5	12	21

**Figure 7.** Fitness change curves.

hyperparameter to cover the hyperparameter search space thoroughly. The fitness of the MLP model was assessed using the MAE metric, the Adam algorithm was used as an optimiser, the learning rate was set to 0.001 and the number of model iterations was 300. Table 7 depicts the optimisation range of model hyperparameters and the four optimisation algorithms acquired for optimal hyperparameters.

The PSO, GWO and SSA optimisation outcomes were subjected to a comparative analysis, as depicted in Fig. 7. The PSO exhibited a sluggish convergence rate and produced the least optimal fitness value compared to the other three optimisation algorithms. On the other hand, the SSA algorithm displayed the initiation of convergence after the 13th generation. In contrast, the GWO algorithm exhibited a faster convergence rate than the SSA and achieved a better fitness value. Notably, the ISSA exhibited the most optimal fitness value despite having a comparatively slower convergence point, thus underscoring the superior prediction accuracy of the ISSA-MLP model.

The results presented in Fig. 8 indicate that the ISSA-optimised MLP model outperforms the models optimised using PSO, GWO and standard SSA algorithms in terms of MAE and RMSE. Specifically, the MAE and RMSE of the ISSA-MLP model are 16.8 and 24.2, respectively, the smallest among the four models. The ISSA algorithm is more effective in optimising the MLP model than the standard SSA algorithm, as it improves the accuracy and speed of the population search. Overall, the ISSA-MLP model can provide a better decision basis for the relevant stakeholders in predicting the departure flight delays generated by GDP.

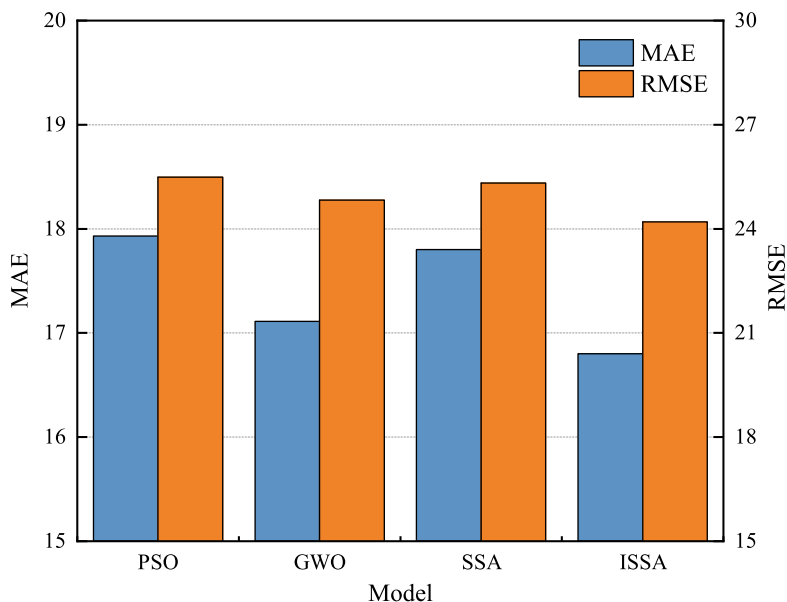


Figure 8. Prediction errors for four optimisation methods.

To assess the significance of the predictive performance disparity between ISSA-MLP and MLP, the Wilcoxon rank sum test was employed by comparing the rank sums of the two data sets. The evaluation focused on RMSE and MAE values obtained through five-fold cross-validation of the models. The significance level for this experiment was set at $\alpha = 0.05$. If $P < 0.05$, it indicates a statistically significant difference between the experimental outcomes of the two algorithms; conversely, if $P > 0.05$, the disparity is deemed insignificant.

Our analysis calculated the P -values for RMSE and MAE as 0.043 and 0.039, respectively. Since both P -values are less than 0.05, it demonstrates a statistically significant difference between ISSA-MLP and MLP in terms of predictive performance.

5.3 Feature importance analysis

The intricate nature of MLP models hinders the ability to comprehend the relationship between input features and output outcomes, resulting in a lack of explanatory insights for prediction results. These models are commonly referred to as “black box” models. To address this limitation, we employ SHAP (Shapley additive explanations) to elucidate the prediction outcomes concerning the variables.

SHAP, proposed by Lundberg et al. in 2017 [30], leverages the concept of Shapley values derived from game theory. It serves as a method to explain the prediction results of machine learning models by evaluating the individual contributions of each feature towards the model’s prediction outcomes.

The significance of a feature can be determined by calculating the sum of the absolute SHAP values for each feature and then averaging them. Figure 9 illustrates the importance of the features based on this methodology. The results indicate that time-related features, such as the day of the month and the hour of the day, hold the highest importance in delay prediction. This observation suggests a strong correlation between flight delays and temporal factors. Subsequently, the weather feature of cloud base height emerges as the next most influential, as it impacts visibility and can lead to prolonged flight delays under reduced to severe visibility conditions. All the remaining variables contribute to the predicted outcome to a similar degree.

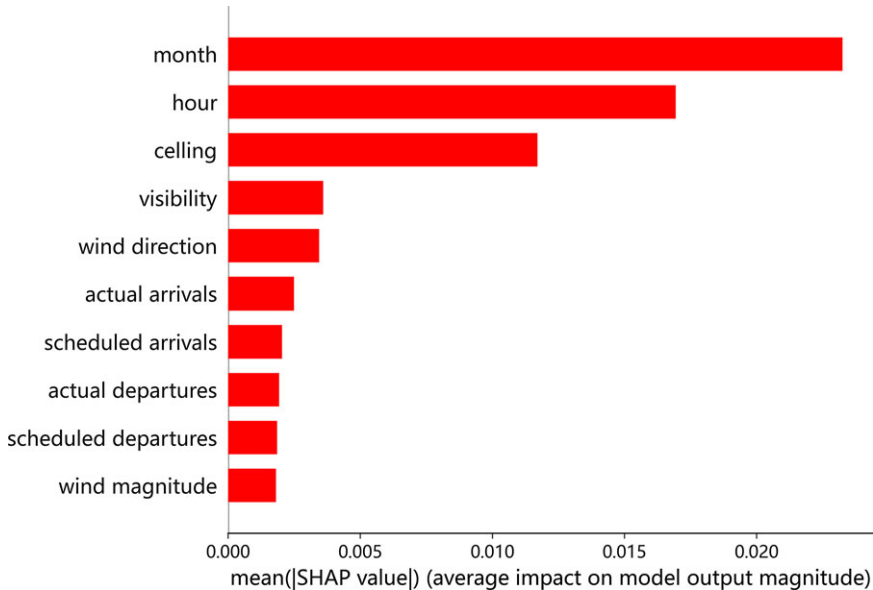


Figure 9. SHAP feature importance.

5.4 Analysis of ISSA-MLP prediction results

The optimal hyperparameters for the MLP model were obtained from Section 5.2 and subsequently integrated into the model for training. Figure 10 depicts the loss variation on the training and test sets during the MLP's training process. We carried out 300 epochs of experimentation and observed that the loss on the test set attained a state of convergence at the 100th epoch, followed by stable fluctuations in the subsequent stages, ultimately stabilising between 0.09 and 0.1. The loss gradually achieved convergence in the training set, reaching a stable state at the 200th epoch. These results indicate that the iterative training process of the MLP is rational and does not exhibit any significant overfitting.

As per Table 5, the ISSA-MLP model exhibits the highest prediction accuracy, with MAE and RMSE values of 16.8 and 24.2, respectively. To visually demonstrate the predictive performance of the ISSA-MLP model, Fig. 11 portrays the predictive performance of select test sets. From the figure, it can be discerned that the prediction errors for most points fall within an acceptable range, albeit for some points with significant changes, the prediction effect falls short of the ideal. The probable reason for this situation is twofold. Firstly, the number of extreme test points is relatively small, and thus the model cannot fully capture the coupling relationship and change characteristics of variables. Secondly, the reasons for flight delays are multifarious, and the features discussed in this paper need to be more comprehensive, which negatively affects the model's learning and prediction performance. Therefore, enhancing the model's generalisation ability will be the focal point of the following research stage.

6.0 Conclusion

This paper presents a novel ISSA-MLP-based prediction model to address the flight delay prediction problem associated with GDP. The model tackles the hyperparameter optimisation problem of deep learning neural network models by conducting a hyperparameter search using the ISSA algorithm, thereby enhancing the model's prediction accuracy. A comprehensive comparative analysis with other machine learning models is also performed, and the primary conclusions are outlined as follows:

- (1) The standard SSA algorithm has been enhanced through several improvements. Firstly, tent chaotic mapping has been incorporated to augment the diversity of the initial population.

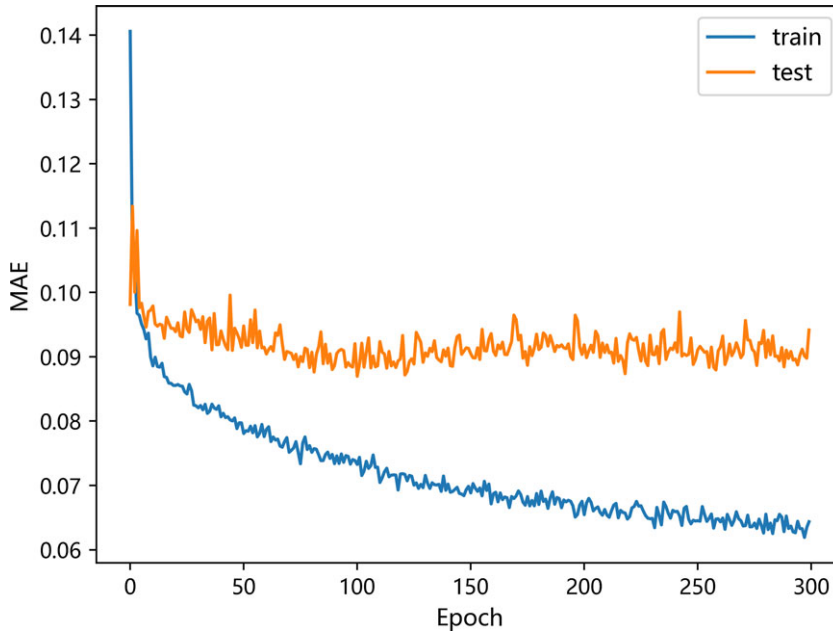


Figure 10. Changes in training loss for ISSA-MLP.

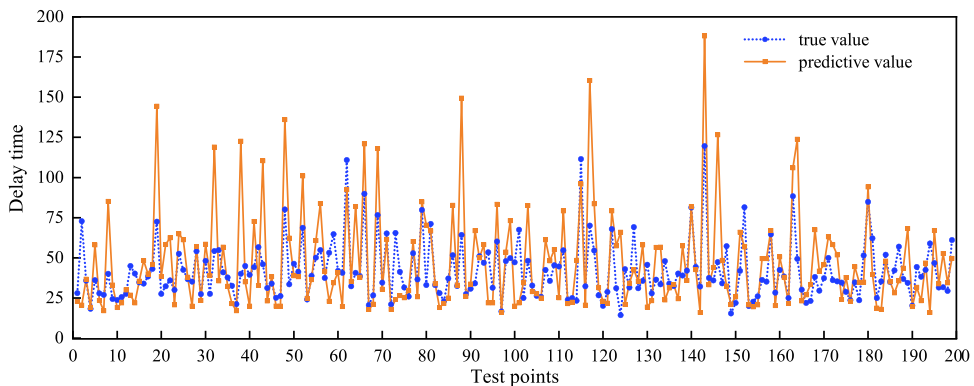


Figure 11. Comparison between predicted and true values of ISSA-MLP model.

Secondly, dynamic adaptive weights have been introduced to the finder to enhance the global search capability of SSA. Lastly, a Levy flight strategy has been integrated with the detector to perturb the optimal solution and prevent local optima. Experimental evaluations using benchmark functions demonstrate that the enhanced SSA algorithm exhibits faster convergence and higher accuracy.

- (2) Compared to other machine learning models, the MLP model exhibits more minor evaluation errors, both in the training and test sets, thus indicating that the MLP model generates more precise and consistent prediction outcomes.
- (3) This study introduces an ISSA-MLP prediction model to forecast departing flight delays due to GDP. The hyperparameters of the MLP model were optimised using ISSA and compared against PSO, SSA, and GWO. The results indicate that the ISSA-MLP model yields predictions with a MAE of 16.8 and RMSE of 24.2, outperforming the other three optimisation algorithms as

confirmed by the Wilcoxon rank sum test. This demonstrates the model's robustness in predicting departing flight delays. Furthermore, we have employed the SHAP model to enhance the interpretability analysis of the MLP model. This approach aims to deepen our understanding of flight delay characteristics, elucidate the influence of flight delay causes, and augment the overall interpretability of the model.

The research presented in this paper could benefit airlines and air traffic control authorities in designing more effective flight scheduling and air traffic control strategies. Given that this study focuses solely on the departure delays resulting from GDP at a single airport, it would be worthwhile to investigate the propagation of GDP-induced delays across a network of airports within a synergistic paradigm in future research.

Acknowledgements. This work was supported by the National Key R&D Program of China (No. 2022YFB2602403) and National Natural Science Foundation of China, grant number [U2033203, 52272333].

Competing interests. 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.

References

- [1] Grabbe S., Sridhar B. and Mukherjee A. Clustering days and hours with similar airport traffic and weather conditions. *J. Aerosp. Inf. Syst.*, 2014, **11**, pp 751–763.
- [2] Bao J., Yang Z. and Zeng W. Graph to sequence learning with attention mechanism for network-wide multi-step-ahead flight delay prediction. *Transp. Res. Part C Emerg Technol.*, 2021, **130**, p 103323.
- [3] Kuhn K.D. Ground delay program planning: Delay, equity, and computational complexity. *Transp. Res. Part C Emerg Technol.*, 2013, **35**, pp 193–203.
- [4] Mukherjee A., Hansen M. and Grabbe S. Ground delay program planning under uncertainty in airport capacity. *Transp. Plan. Technol.*, 2012, **35**, pp 611–628.
- [5] Yan C., Vaze V. and Barnhart C. Airline-driven ground delay programs: A benefits assessment. *Transp. Res. Part C Emerg Technol.*, 2018, **89**, pp 268–288.
- [6] Jacquillat A. Predictive and prescriptive analytics toward passenger-centric ground delay programs. *Transp. Sci.*, 2022, **56**, pp 265–298.
- [7] Liu J., Li K. and Yin M. Optimizing key parameters of ground delay program with uncertain airport capacity. *J. Adv. Transp.*, 2017, **2017**, pp 1–9.
- [8] Liu Y., Hansen M. and Zhang D. Modeling ground delay program incidence using convective and local weather information. In *12th USA/Europe Air Traffic Management Research and Development Seminar 2017*. Seattle, Washington, USA: The European Organisation for the Safety of Air Navigation (EUROCONTROL), 2020, pp 1–8.
- [9] Yang Z., Chen Y. and Hu J. Departure delay prediction and analysis based on node sequence data of ground support services for transit flights. *Transp. Res. Part C Emerg Technol.*, 2023, **153**, p 104217.
- [10] Wang Z., Liao C. and Hang X. Distribution prediction of strategic flight delays via machine learning methods. *Sustainability*, 2022, **14**, p 15180.
- [11] Smith D.A. and Sherry L. Decision support tool for predicting aircraft arrival rates from weather forecasts[C]. In *2008 Integrated Communications, Navigation and Surveillance Conference*. Bethesda, MD, USA: IEEE, 2008, pp 1–12.
- [12] Mangortey E. *A Dissertation Presented to The Academic Faculty*. Atlanta, Georgia, USA: Georgia Institute of Technology, 2019.
- [13] Liu Y., Liu Y. and Hansen M. Using machine learning to analyze air traffic management actions: Ground delay program case study. *Transp. Res. Part E Logist. Transp. Rev.*, 2019, **131**, pp 80–95.
- [14] Chen Y., Xu Y. and Hu M. Demand and capacity balancing technology based on multi-agent reinforcement learning. In *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*. San Antonio, TX, USA: Institute of Electrical and Electronics Engineers (IEEE), 2021, pp 1–9.
- [15] Dong X., Zhu X. and Hu M. A methodology for predicting ground delay program incidence through machine learning. *Sustainability*, 2023, **15**, p 6883.
- [16] Yu B., Guo Z., Asian S., et al. Flight delay prediction for commercial air transport: A deep learning approach. *Transp. Res. Part E Logist. Transp. Rev.*, 2019, **125**, pp 203–221.
- [17] Zoutendijk M. and Mitici M. Probabilistic flight delay predictions using machine learning and applications to the flight-to-gate assignment problem. *Aerospace*, 2021, **8**, p 152.
- [18] Esmaeilzadeh E. and Mokhtarimousavi S. Machine learning approach for flight departure delay prediction and analysis. *Transp. Res. Rec. J. Transp. Res. Board*, 2020, **2674**, pp 145–159.
- [19] Mokhtarimousavi S. and Mehrabi A. Flight delay causality: Machine learning technique in conjunction with random parameter statistical analysis. *Int. J. Transp. Sci. Technol.*, 2023, **12**, pp 230–244.

- [20] Mangortey E., Pinon-Fischer O.J. and Puranik T.G. *Predicting the Occurrence of Weather And Volume Related Ground Delay Programs, AIAA Aviation 2019 Forum*. Dallas, TX: American Institute of Aeronautics and Astronautics, 2019.
- [21] Xue J. and Shen B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control Eng.*, 2020, **8**, pp 22–34.
- [22] Jiang L., Yang L. and Huang Y. COD optimization prediction model based on CAWOA-ELM in water ecological environment. *J. Chem.*, 2021, **2021**, pp 1–9.
- [23] Deepa R. and Venkataraman R. Enhancing whale optimization algorithm with levy flight for coverage optimization in wireless sensor networks. *Comput. Electr. Eng.*, 2021, **94**, p 107359.
- [24] Zhou G., Moayedi H. and Bahiraei M. Employing artificial bee colony and particle swarm techniques for optimizing a neural network in prediction of heating and cooling loads of residential buildings. *J. Clean. Product.*, 2020, **254**, p 120082.
- [25] Hashem Samadi S., Ghobadian B., Nosrati M., et al. Investigation of factors affecting performance of a downdraft fixed bed Gasifier using optimized MLP neural networks approach. *Fuel*, 2023, **333**, p 126249.
- [26] Hazarika B.B., Gupta D., and Natarajan N. Wavelet kernel least square twin support vector regression for wind speed prediction. *Environ. Sci. Pollut. Res.*, 2022, **29**, pp 86320–86336.
- [27] Chen T. and Guestrin C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Francisco California USA: ACM, 2016, pp 785–794.
- [28] Fayed H.A. and Atiya A.F. Speed up grid_search for parameter selection of support vector machines. *Appl. Soft Comput.*, 2019, **80**, pp 202–210.
- [29] Zhang X., Liu S. and Yan J. Fixed-time cooperative trajectory optimisation strategy for multiple hypersonic gliding vehicles based on neural network and ABC algorithm. *Aeronaut. J.*, 2023, pp 1–15.
- [30] Lundberg S.M. and Lee S.I. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Long Beach, CA, USA: Curran Associates Inc., 2017, pp 4–9.