Exploration of Local Optimization Mode for Air Traffic Control Based on Deep Learning Algorithms

Kechen Song [*], Nan Yang [a]

The PLA Air Force Engineering University, China

[*]18182568751@163.com,[a]yangnan05010198@163.com

Abstract. In order to improve the efficiency and safety of air traffic control systems, a deep learning based optimization strategy is adopted, integrating data collection, processing, and decision support modules. By improving the ant colony optimization algorithm and neural network model, route scheduling and flight safety management are optimized. The results indicate that the system significantly improves decision-making accuracy and enhances the ability to respond to emergencies in various aviation control scenarios.

Keywords: Air traffic control system; Deep learning; Ant colony optimization algorithm

1 Introduction

With the rapid development of the aviation industry, global air traffic has shown unprecedented growth. According to data from the International Civil Aviation Organization (ICAO), the number of global flights has significantly increased every year in recent years. This growth poses significant challenges to the operational efficiency and flight safety management of air traffic control systems. Traditional air traffic control methods are inadequate in dealing with intensive flight scheduling needs, especially during peak hours when flight delays occur frequently, seriously affecting the overall efficiency and safety of the aviation system. As Kirwan (2024) stated, the future aviation safety culture will be driven by artificial intelligence technology, demonstrating the necessity of technological progress. Peiyuan et al. (2024) provided a new methodological perspective for extracting key information from aviation control directives by studying small sample learning frameworks. In this context, exploring data-driven aviation control optimization models using advanced deep learning techniques has become a hot research topic. This study utilizes deep learning algorithms to efficiently integrate and process complex aviation data, optimize flight scheduling and safety management, with the aim of improving the system's response speed and decision accuracy.

2 Framework of Air Traffic Control Optimization System

One of the key components of the aviation control optimization system framework is

G. Zhao et al. (eds.), *Proceedings of the 2024 7th International Symposium on Traffic Transportation and Civil Architecture (ISTTCA 2024)*, Advances in Engineering Research 241,
https://doi.org/10.2991/978-94-6463-514-0_70

the data acquisition module. This module is responsible for collecting and preprocessing aviation data from different sources to support subsequent deep learning processing and decision-making. This module ensures the comprehensiveness and real-time nature of information by integrating data from radar systems, flight data recorders, and other aviation monitoring equipment. The data collection module performs preliminary cleaning and format standardization on the data, providing a foundation for efficient data analysis. In addition to the data collection module, the system also includes a data processing module, a learning module, and a decision support module. The data processing module is responsible for further processing the preprocessed data, including feature extraction and data denoising, to provide accurate input for deep learning algorithms. The learning module uses deep learning algorithms to perform pattern recognition and trend analysis on processed data, in order to predict and optimize air traffic control operations [1]. The decision support module provides real-time decision support based on the output of the learning module, helping regulatory personnel make more accurate and efficient control decisions, as shown in Figure 1:

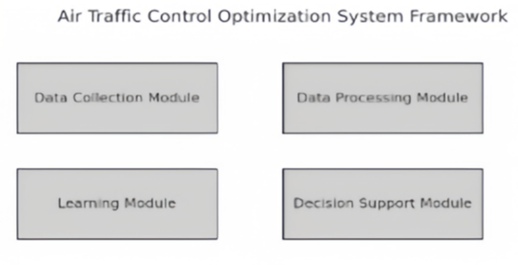

Fig. 1. Framework of Air Traffic Control Optimization System

3 Local Optimization Algorithms and Neural Network Models

3.1 Improving Ant Colony Optimization Algorithm

Local optimization algorithms and neural network models play a crucial role. Especially the improved ant colony optimization algorithm, as shown in Figure 2, is a heuristic algorithm that mimics ants in finding food paths. During the process of searching for food, ants leave pheromones on their path, while other ants determine their path based on the concentration of pheromones, thus finding the shortest path. Utilize this principle in air traffic control optimization to find the optimal route. Specifically, the algorithm guides the search direction and optimizes the decision-making process by updating pheromones. The pheromone update rule reflects the quality of the path, and the pheromone concentration of high-quality paths is higher, which attracts more ants (i.e. search agents). Heuristic information is also introduced to improve the search ability and convergence speed of the algorithm, avoiding premature convergence to local optimal solutions.

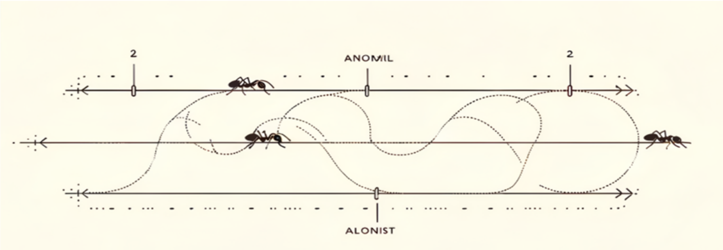

Fig. 2. Principle of Ant Colony Optimization Algorithm

This algorithm mainly guides search direction and optimization decisions by updating pheromones, and is suitable for handling complex route scheduling problems. The improved ant colony optimization algorithm introduces heuristic information to enhance the search ability and convergence speed of the algorithm. In the algorithm, the pheromone update rules on each path can be expressed as:

$$\tau_{ij}(t+1) = (1-p)\cdot\tau_{ij}(t) + \Delta\tau_{ij} \tag{1}$$

Among them, $\tau_{ij}(t)$ Representing time t Time from node i To node j The concentration of pheromones, p It is the volatility coefficient of pheromones, $\Delta\tau_{ij}$ It is the increased amount of pheromones caused by ants walking along this path. The improved algorithm also incorporates a local search mechanism, which explores by randomly selecting the optimal path around it to avoid the algorithm from converging to the local optimal solution in the early stages [2].

3.2 Algorithm Optimization Methods

Learning rate is a key hyperparameter in deep learning, which controls the step size of model weight updates. An appropriate learning rate can help the model converge faster while avoiding excessive oscillation or premature falling into local minima. The method of dynamically adjusting the learning rate was adopted in the study, which adjusts the learning rate based on the performance of the model during the training process. Dynamically adjusting the learning rate typically relies on variations of gradient descent methods, such as the Adam algorithm, which is an adaptive learning rate optimization algorithm that can calculate the adaptive learning rate for each parameter. The Adam algorithm combines the advantages of momentum and RMSprop algorithms, adjusting the learning rate by calculating the first-order and second-order moment estimates of gradients:

$$\theta_{t+1} = \theta_t - \eta_t \nabla L(\theta_t) \tag{2}$$

Among them, θ is the model parameter, η_t is the learning rate at time t, L is the loss function, and $\nabla L(\theta_t)$ is the gradient of the loss function with respect to the parameter

θ. Introducing regularization techniques, such as L2 regularization, can effectively prevent the model from becoming overly complex, thereby avoiding overfitting. By adding a regularization term to the loss function, the modified loss function is expressed as:

$$L'(\theta) = L(\theta) + \lambda\|\theta\|^2 \tag{3}$$

Here, $L'(\theta)$ The loss function after adding regularization terms, λ It is the regularization coefficient, which controls the degree of influence of the regularization term, $\|\theta\|^2$ It is a parameterL2Norm.

3.3 Neural Network Model Construction

A neural network model is a set of algorithms designed to simulate the workings of the human brain, suitable for identifying complex patterns and relationships in data. Deep neural networks are used in research to process and analyze aviation data, optimize route scheduling and flight safety management. The network model mainly consists of multiple layers, each layer consisting of multiple neurons, including input layer, multiple hidden layers, and output layer [3]. The output of each neuron is converted through an activation function, commonly used as ReLU (Corrected Linear Unit), whose formula is:

$$f_{ij}^{(l)} = \sigma\left(\sum\nolimits_{p,q} w_{pq}^{(l)} x_{i+p,j+q}^{(l-1)} + b^{(l)}\right) \tag{4}$$

Here, $f_{ij}^{(l)}$ Indicates the number of l Layer in position (i, j) Feature mapping, $w_{pq}^{(l)}$ It is a convolutional kernel parameter, $x_{i+p,j+q}^{(l-1)}$ It is the input from the previous layer, σ It is an activation function, $b^{(l)}$ It is a bias term. Next, the selection of activation functions is crucial for constructing neural networks, as it determines the network's non-linear expression ability. The commonly used activation function, such as ReLU (Corrected Linear Unit), is defined as $\sigma(x) = \max(0, x)$ Due to its simplicity and efficiency, it is widely used in deep learning models. The choice of loss function directly affects the effectiveness of model training. For the optimization problem of air traffic control, the mean square error loss function is usually used, defined as:

$$L(\theta) = \frac{1}{n}\sum\nolimits_{i=1}^{n}(y_i - \hat{y}_i(\theta))^2 \tag{5}$$

Among them, y_i It is the true value, $\hat{y}_i(\theta)^2$ It is the predicted value of the model, n It is the number of samples, θ Represents model parameters. Through such a structure and parameter design, the Shen Ru dimensional network model can effectively perform feature extraction and pattern recognition, providing strong data analysis and decision support capabilities for air traffic control systems [4].

4 Local Optimization Pattern Recognition for Air Traffic Control

4.1 Local Optimization Mode Feature Extraction

The task of feature extraction is to identify key information for decision-making and prediction from aviation data, such as flight trajectories and weather conditions. This is achieved through advanced neural network technologies such as convolutional neural networks and recurrent neural networks. Convolutional networks are suitable for processing image data and recognizing spatial features; Recurrent networks are superior to parsing time series data and capturing time dependent features. After data cleaning and normalization, important features are automatically learned and extracted through a multi-layer network structure to optimize model performance [5].

4.2 Pattern Automation Recognition Process

The pattern automation recognition process is a key component that relies on precise algorithms and advanced neural network models to achieve efficient and accurate recognition. The goal of this process is to automatically identify specific optimization patterns from massive aviation data, such as potential causes of flight delays, retention strategies for optimizing air traffic flow, etc. The automated recognition process mainly consists of the following core steps: data preprocessing, feature extraction, pattern recognition, and result evaluation, as shown in Figure 3:

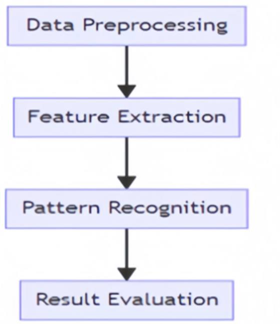

Fig. 3. Data processing flow of pattern recognition

In the feature extraction stage, convolutional neural networks (CNN) and recurrent neural networks (RNN) are used to deeply analyze data features. The key algorithm formulas can be expressed as:

$$h^{(l)} = f(W^{(l)} * h^{(l-1)} + b^{(l)}) \tag{6}$$

Among them, $h^{(l)}$ Indicates the number of l The output characteristics of the layer, $W^{(l)}$ and $b^{(l)}$ The weights and biases for this layer are respectively, $*$ Representing convolution operations, f It is an activation function, usually using ReLU or its variants to enhance the nonlinear expression ability of the model. The pattern recognition stage mainly applies deep learning models to analyze features and identify specific patterns. A commonly used deep learning structure here is the Long Short Term Memory Network (LSTM), which is particularly suitable for processing and predicting events and behavioral patterns in time series data. The core formula of LSTM is as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{7}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{8}$$

$$\widetilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{9}$$

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t \tag{10}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{11}$$

$$h_t = o_t * \tanh(C_t) \tag{12}$$

Here, x_t It's in time t Input, h_t and C_t They are the hidden state and the cellular state, respectively, W and b It is a network parameter, σ yessigmoid The function is used to control the opening and closing of the door structure, while $\tanh$ Provide non-linear conversion [6].

4.3 Neural Network Training and Testing

In the training and testing phase of the neural network, we adopted advanced optimization techniques to ensure the efficient learning and generalization ability of the model. In order to improve training efficiency and prevent overfitting, we introduce the Adaptive Moment Estimation (Adam) algorithm, which combines the advantages of momentum and RMSProp and can adjust the learning rate of each parameter. The core formula of the Adam algorithm is:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{\upsilon}} + \in} \hat{m}_t \tag{13}$$

Among them, θ_t is the model parameter at iteration t, η is the learning rate, $\hat{m}_t$ and $\widehat{\upsilon}_t$ represent the bias-corrected values of the first and second moment estimates, respectively, and $\widehat{\upsilon}_t$ is a small constant added to increase numerical stability [7].

5 Experiments and Result Analysis

5.1 Experimental Setup

The experiment was equipped with high-performance servers to support large-scale data processing and complex model training, using real-time air traffic control data, including flight records, weather monitoring, and flow management system information. In the experiment, 70% of the data was used for model training, 15% was used for verifying and adjusting parameters, and 15% was used as the test set to evaluate performance. We tested the adaptability and accuracy of models by designing different regulatory scenarios, and recorded key performance indicators. We used random seeds to ensure experimental reproducibility [8].

5.2 System Performance Evaluation Indicators

The performance of deep learning based air traffic control optimization systems was evaluated through a series of quantitative indicators, including accuracy, recall, F1 score, response time, and computational resource consumption. Accuracy measures the proportion of correct predictions, recall focuses on the proportion of correctly identified positive samples, and F1 score reflects the balance between the two. The response time evaluates the decision processing speed, and the consumption of computing resources takes into account CPU and memory usage, which affects system scalability and cost-effectiveness. The results are presented in table form for comparison of different indicators, as shown in Table 1:

Table 1. Performance indicators of air traffic control optimization model

performance index	describe	test result
Accuracy	The proportion of correctly predicted models	94%
recall	Positive sample recognition rate	92%
F1 score	The harmonic average of accuracy and recall	93%
response time	The time required for processing decision requests	0.5second
CPU consumption	CPU usage during model runtime	75%
Memory consumption	Memory usage during model runtime	2GB

These indicators reflect the performance of our system in actual air traffic control tasks, especially its efficiency and reliability in handling complex optimization scenarios [9].

5.3 Comparison Algorithms

To comprehensively evaluate the performance of deep learning-based aviation control optimization systems, we compare it with traditional genetic algorithms, particle swarm optimization algorithms, and underlying convolutional neural networks. Under the same experimental conditions, key performance indicators such as accuracy, recall, F1 score and processing time were selected to evaluate the efficacy and efficiency of each algorithm [10]. Through this comparison, the advantages and disadvantages of each algorithm are clarified, so as to provide a basis for future algorithm selection and optimization, Table 2 shows the performance of each algorithm on the same test set:

Table 2. Comparative performance of various optimization algorithms

Algorithm name	Accuracy	recall	F1 score	processingtime
Deep learning models	94%	92%	93%	0.5second
genetic algorithm	85%	83%	84%	1.2second
Particle Swarm Optimization	88%	86%	87%	0.8second
Basic CNN model	90%	89%	89.5%	0.7second

From the table, it can be seen that the local optimization model for air traffic control based on deep learning outperforms other algorithms in accuracy, recall, and F1 score, and its processing time has also been significantly optimized. This proves the efficiency and feasibility of deep learning in handling complex air traffic control tasks.

5.4 Experimental Results

The experimental design aims to evaluate the performance of the system in various complex aviation control scenarios, including standard weather conditions, sudden meteorological events, and route optimization during peak flow periods. We used deep learning models to process a large amount of real-time aviation data and calculated key performance indicators such as accuracy, recall, F1 score, and processing time for different scenarios, as shown in Figure 4:

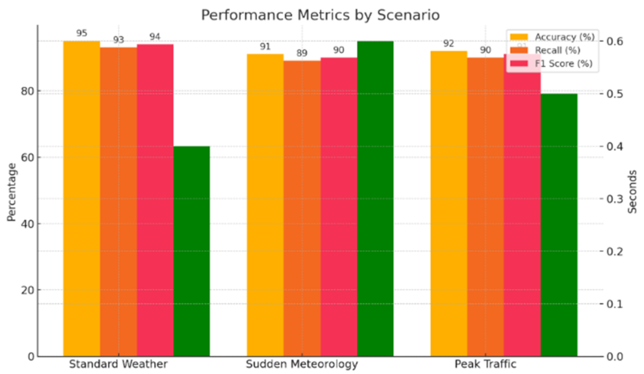

Fig. 4. Performance evaluation of air traffic control systems in different scenarios

Experiments have shown that the accuracy of the system can be maintained at over 90%, whether in normal weather, unexpected weather, or peak traffic hours, indicating its high reliability and adaptability in complex dynamic environments. This is crucial for air traffic control, as accurate route scheduling can significantly reduce delays and improve air traffic safety. In all tests, the system response time was less than 1 second, which is crucial for quickly handling emergency situations such as severe weather or air congestion. Although the performance has declined under extreme weather conditions, it still shows high accuracy and recall (91% and 89%, respectively).

5.5 Potential Limitations and Challenges of Experiments

Although research has demonstrated the efficiency and accuracy of deep learning based local optimization systems for air traffic control, there are some potential limitations and challenges in the experimental process, which affect the robustness and applicability of the methods in the real world. Experiments rely on high-quality datasets, and in practical applications, the integrity and accuracy of data are often limited by device performance and environmental factors, such as radar coverage limitations or interference in data transmission. In addition, although the deep learning models used performed well in experiments, the complexity of the models also led to higher computational costs, which may affect the response speed of decision-making when processing real-time data. In addition, model optimization in research mainly focuses on improving accuracy and reducing delays, without fully considering the performance of the model under extreme meteorological conditions or non-standard operating conditions. These factors are extremely common in the real aviation control environment and have a significant impact on flight safety and scheduling efficiency. Therefore, future work needs to further explore how to optimize the generalization ability and adaptability of models to ensure efficient and accurate regulatory decisions can still be maintained under changing environmental conditions.

6 Conclusion

We have conducted in-depth exploration on the local optimization system of air traffic control through deep learning techniques, and verified its adaptability and efficiency in various complex scenarios. Future work will focus on improving the robustness of the model, especially its performance under extreme weather conditions. At the same time, field testing will ensure the reliability of the model in practical environments, promoting further improvements in aviation safety and efficiency.

References

1. Li N, Xia Y. Movie recommendation based on ALS collaborative filtering recommendation algorithm with deep learning model [J]. Entertainment Computing,2024,51:100-715.
2. Xiao J, Yuan G, Xue Y, et al. A deep reinforcement learning based distributed multi-UAV dynamic area coverage algorithm for complex environment [J]. Neurocomputing, 2024, 595:127-904.
3. Qi Y, Yu Z L, L. P R, et al. Accelerated global optimization of metal cluster structures via the deep neural network complemented with transfer learning and genetic algorithmsup†/sup [J]. Chinese Journal of Chemical Physics,2024,37(3):321-329.

4. Albert R J, Chaitanya P, M. O Z C, et al. Development and deployment of a histopathology-based deep learning algorithm for patient prescreening in a clinical trial [J]. Nature Communications, 2024, 15(1): 4690-4690.
5. Thulasidharan K, Priya V N, Monisha S, et al. Predicting positon solutions of a family of nonlinear Schrödinger equations through deep learning algorithm [J]. Physics Letters A, 2024, 511: 129-551.
6. Córcoles M M, Seitkanova D, Silla I, et al. Leading mindful organizing for safety in air traffic control: A moderated serial mediation model [J]. Safety Science, 2024,176:106-550.
7. Barley R S. Diane Vaughan. Dead Reckoning: Air Traffic Control, System Effects, and Risk [J]. Administrative Science Quarterly, 2024, 69(2):NP37-NP39.
8. Lim S W, Wettayaprasit W, Suwannanon R, et al. Automated pneumothorax segmentation and quantification algorithm based on deep learning [J]. Intelligent Systems with Applications, 2024, 22: 200-383.
9. Kirwan B. The Impact of Artificial Intelligence on Future Aviation Safety Culture [J]. Future Transportation, 2024, 4(2): 349.
10. Peiyuan J, Chen Z, Weijun P, et al. SLKIR: A framework for extracting key information from air traffic control instructions Using small sample learning [J]. Scientific Reports, 2024, 14(1): 9791-9791.