IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 18, 2019, accepted February 5, 2019, date of publication February 22, 2019, date of current version March 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2900988

Research on Situation Awareness of Airport Operation Based on Petri Nets

QIAN LUO[1]*, YANRU CHEN[2]*, LUNYUE CHEN[2]*, XIAO LUO[1], HUAN XIA[1], YUANYUAN ZHANG[2], AND LIANGYIN CHEN[2,3], (Member, IEEE)

[1]Second Research Institute, General Administration of Civil Aviation of China, Chengdu 610042, China
[2]College of Computer Science, Sichuan University, Chengdu 610065, China
[3]Institute for Industrial Internet Research, Sichuan University, Chengdu 610065, China

Corresponding authors: Yuanyuan Zhang (yuanyuanzhang@stu.scu.edu.cn) and Liangyin Chen (chenliangyin@scu.edu.cn)

This work was supported in part by the NNSFC and Civil Aviation Administration of China under Grant U1533203, in part by the collaborative innovation of industrial cluster project "A New Generation of Intelligently Operating and Controlling System for Large Airport" under Grant 2016-XT00-00015-GX, in part by the National Natural Science Foundation of China under Grant 61373091, in part by the Science and Technology Department of Sichuan Province under Grant 2019YFG0184 and Grant 19CXTD0005, and in part by the National Key R&D Program of China under Grant 2018YFB1601200 and Grant 2018YFB1601201.

*Qian Luo, Yanru Chen, and Lunyue Chen contributed equally to this work.

ABSTRACT The civil aviation industry is undergoing rapid development. However, the on-time rate of airport flights and passenger service quality are not particularly satisfying. The cause of the above problems is the contradiction between the limited operational support capability and the continuous growth of passenger traffic volume. Therefore, the key to solving these problems is achieving situation awareness of airport operation. Many situation awareness algorithms, typically categorized into modeling and machine learning, have been proposed in the past years. However, existing models lack flexibility and their prediction accuracy is unstable. Machine learning's results cannot be timely and effective when external conditions are suddenly changed although some related algorithms have higher accuracy because of the retraining of artificial neural network (ANN). This paper proposes a situation awareness method based on Petri nets (PNs). This method introduces the queuing theory and perceptual parameters into the existing PN and constructs the perceptual PNs' model for general service systems so that it can quickly model different scene service systems. In combination with the ANN, this paper proposes a complete situation awareness algorithm to realize a sustained and accurate situation awareness prediction of the service system by solving point estimations of the macroscopic and microscopic situation in this model, which helps to address some challenges faced by current civil aviation airports. By experimenting on-ground support in civil aviation airports and the access of website as well as comparing the situation separately with Airport Collaborative Decision Making and stochastic PNs, the validity and accuracy of the algorithm proposed in this paper are well verified.

INDEX TERMS Artificial neural network, modeling, Petri nets, situation awareness.

I. INTRODUCTION

Air transportation is characterized by high efficiency, safety and comfort. Also, the ongoing development of the civil aviation industry is rapid. However, at the same time, airport flight support capability and passenger service quality are not particularly satisfying. Challenges are widespread in most airports, such as information interconnection, flight operation control capabilities, baggage handling, traffic control, and airport security. These make it difficult for airport operation to be long-term safe, stable, and highly efficient. The root cause of the above problems is the contradiction between the limited airport capacity and the continuous growth of passenger traffic volume. Thus, the key to solving the problem of civil aviation airport operation is achieving situation awareness of the airport operation by using the big data infrastructure. Currently, for various operational problems of civil aviation airports, scholars mainly use modeling and machine learning methods to predict some parameters. These methods can be viewed as situation awareness research.

The associate editor coordinating the review of this manuscript and approving it for publication was Shouguang Wang.

In 1995, Endsley [1] proposed a new situation awareness method which made use of available data to predict future events. Since then, situation awareness has been widely applied in many areas, such as military field [2], [3], power system [4], [5] and network security [6], [7]. Usually, for service systems, situation awareness can provide decision support for decision makers to understand the operating status of a system. At present, for common service systems, scholars mainly use modeling and machine learning to discuss systems' situation.

Davidrajuh and Lin [8] put forward a modified modeling and simulation tool, named GPenSIM for discrete service systems, which is based on Petri nets (PN). They studied the traffic capacity of the Norwegian Harstad/Narvik Airport and provided reasonable suggestions for the operation of this airport based on their results. Mori [9] developed an aircraft ground-taxiing model using cellular automata to analyze the traffic conditions of Tokyo International Airport. This model considered the taxiing speed and the time histories of taxiing; also, the distribution of aircrafts on each runway and relative sliding time could be well obtained through simulation experiment. Nagatani [10] used the nonlinear-map to study the dynamic characteristics and control problems of aircrafts between airports. Chen and Xie [11] used the Markov model to make a grey prediction of airport's energy consumption. Although modeling methods have achieved important results in the study of situation awareness, they also have disadvantages, such as relatively low prediction accuracy, lack of flexibility, difficulty in analysis, etc., when external conditions are complicated.

Regarding the methods of machine learning, Kim et al. [12] built a prediction model, called LSTM-RNN, to predict aircraft delays. By applying deep LSTM-RNN architecture to the prediction model, a reliable delay status of a single day could be acquired. Also, the most accurate delay states for individual flights were acquired by feeding the delay status of a day to the individual flight delay model. Li et al. [13] used clustering neural networks to study the prediction of airport congestion. Wu et al. [14] employed Bayesian network technology to study the dynamic characteristics of passenger flow in the airport. Actually, methods based on machine learning technology have relatively high prediction accuracy. However, this kind of algorithm lacks flexibility, and it cannot change with external conditions. For this reason, machine learning methods cannot provide effective decision support in urgent and extreme situations.

Based on the actual conditions of airport operations and existing technology, we add machine learning to modeling technology. The improved algorithm combines the advantages of these two kinds of algorithms, and has good versatility, compatibility and accuracy. Our main contributions are as follows:

1) By integrating and improving several existing PN theories, introducing queuing theory and perceptual parameters, we constructed the Perceptual Petri nets (PPN) model, which is meaningful for general service systems.
2) We discuss a solving method of the PPN model and propose the situation awareness algorithm for general service systems based on artificial neural network (ANN).

On the basis of the above contributions, we used the PPN model and its algorithm to carry out situation awareness experiment on ground service systems in civil aviation airports and compared it with Airport Collaborative Decision Making (A-CDM), which is widely used in reality. The conclusion that is reached is that PPN greatly improves the prediction precision. Compared with A-CDM, PPN shows excellent performance advantages. The supplementary experiment on the access of website also proves our conclusion.

The rest of the paper is organized as follows: We first introduce the motivation of our method and the background knowledge of our constructed model (PPN) in Section II. Then, in Section III, we illustrate the reasons why we added queuing theory and perceptual parameters to our PPN. Section IV and Section V separately present the establishment and solution of our model, and performance evaluation results of PPN are given in Section VI. Finally, we conclude this work and discuss our future work in Section VII.

II. MOTIVATION AND MODEL BACKGROUND OF PPN

Currently, some airports and air traffic management bureaus have begun to adopt A-CDM. A-CDM [15] integrates airports, air traffic management bureaus, airlines and other related parties to achieve coordinated decision-making during airport operations management. This system integrates and shares all parties' data to realize the rational allocation of resources and improve the overall operational efficiency to ensure that flights are on time. Therefore, it improves the predictability of events and optimizes the utilization of resources.

A-CDM realizes the calculation and prediction of specific times in the flight process, but its situation awareness is still relatively weak. Firstly, its various estimated specific times still need to be given by professional dispatchers; hence, it's not effective. Secondly, the update of its forecast results depends on the exchange of information between airlines, airports and air traffic management bureaus. It is more like a plan than a forecast. In addition, when external conditions change, it is necessary to manually correct each estimated specific time. Obviously, the accuracy is not particularly satisfying. The above shows that the A-CDM system only initially achieves situation awareness of operation of civil aviation airport and cannot completely solve the operational problems currently faced.

As mentioned above, the key to solving the problem of civil aviation airport operation is achieving situation awareness of airport operation by using big data infrastructure. On this basis, we decided to conduct a research on situation awareness of airport operations. Since PN have the ability to describe the static structures and analyze the dynamic behaviors of systems, we chose PN as the basis for our work. Therefore, in this section, we first introduce the main idea of basic PN. Second, the motivation of our method is presented accordingly.

A. PN AND ITS TRANSITION FIRING RULE

PN were first proposed in 1962 by former German scientist Petri [16], and they can model and evaluate Discrete Event Systems (DES) [17]. DES (such as manufacturing systems or information networks) are highly parallel and distributed, and they need to be evaluated from a qualitative point of view as well as from a quantitative point of view. Since the complexity of these systems increases due to many factors, both qualitative and quantitative analyses are used more and more frequently. Proth claims that PN are the most powerful set of tools that can support the functional specification as well as these two types of analysis. Because PN have the characteristics of being concurrent, asynchronous, distributed, parallel, nondeterministic and stochastic, Murata [18] believes that PN are graphical and mathematical modeling tools applicable to many systems. As graphical tools, PN can be used as a visual-communication aid (like block diagrams and flow charts), and tokens are used to simulate the dynamic and concurrent activities of systems. Besides, as mathematical tools, PN can establish mathematical models that govern the behavior of systems (such as state equations and algebraic equations). Regarding our situation, PN can be used to facilitate the establishment of general service system models, and they are easy to analyze by relevant mathematical tools and simulation software. PN already have many applications and researches in some fields, such as deadlock control [19]–[21], fault diagnosis [22]–[25], forbidden state problem [26], [27] and other modeling analyses and applications [28]–[32].

A basic PN [16] is defined as follows:

Definition 1: A triad $BN = (S,T,F)$ satisfying the following conditions is called a prototype PN:
(a)$S \cup T \neq \emptyset$;
(b)$S \cap T = \emptyset$;
(c)$F \subseteq (S \times T) \cup (T \times S)$

In the above definition, elements are called *places* in set S, *transitions* in set T, and *flow relations* in set F. For $x \in S \cup T$, pre-set is defined as $^{\bullet}x = \{y|y \in S \cup T \wedge (y,x) \in F\}$ and post-set is defined as $x^{\bullet} = \{y|y \in S \cup T \wedge (x,y) \in F\}$. A simple prototype PN is shown in Fig.1, in which arrows represent the elements of the set F. This definition describes the static structure of a system. For general service system models, element S stands for the set of service nodes in the system, T for the set of service processes, and F for the set of partial ordering relation.

Definition 2: Supposing $PN = (BN,M)$ is a prototype PN system (M is called a marking of BN and is a mapping from S to N^+), its transition firing rules are as follows:
(a)For $t \in T$, $\forall s \in S$, if $s \in {^{\bullet}t} \rightarrow M(s) \geq 1$; t can transit, and M is denoted as $M[t\rangle$;
(b)M will transform into M' when t happens, denoted as $M[t\rangle M'$. We define M' as follows:

$$M'(s) = \begin{cases} M(s) - 1, s \in {^{\bullet}t} - t^{\bullet} \\ M(s) + 1, s \in t^{\bullet} - {^{\bullet}t} \\ M(s), \textit{others} \end{cases}$$

There is a simple example of this transition. For the prototype PN in Fig.1, t_1 can transit when $M_0 : \{s_0, s_1\} \rightarrow \{0, 3\}$. Then, the marking of this system is $M_1 : \{s_0, s_1\} \rightarrow \{1, 2\}$. Transition firing rule describes the state of a system and its changes. For general service systems, M describes the distribution of the service object between the various service nodes of the system, and the transition rule describes the process of receiving services. According to the above definition, M and its transition rules describe the situation in a service system.

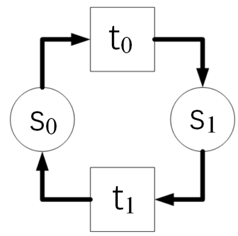


FIGURE 1. A simple prototype PN.

B. EPN AND ITS TRANSITION FIRING RULE

Definitions 1 and 2 show that the prototype PN can fully model general service systems. However, in order to describe some elements more easily, such as the capacity of service nodes and the preference of service object flowing between nodes, we need to further extend this prototype PN. We used function K to represent capacity and W to represent weights.

Definition 3: A 6-tuple $EPN = (S,T,F,K,W,M)$ satisfying the following conditions is called an extended PN:
(a)$S \cup T \neq \emptyset$;
(b)$S \cap T = \emptyset$;
(c)$F \subseteq (S \times T) \cup (T \times S)$;
(d)$W : F \rightarrow N^+$;
(e)$K : S \rightarrow N^+$;
(f)$M : S \rightarrow N^+$

At the same time, EPN needs to meet the following transition firing rules:
(g)For $t \in T$, the condition of $M[t\rangle$ is

$$\begin{cases} \forall s \in {^{\bullet}t} : M(s) \geq \sum\limits_{s_i \in {^{\bullet}t}} W(s_i, t) \\ \forall s \in t^{\bullet} - {^{\bullet}t} : M(s) + W(t,s) \leq K(s) \\ \forall s \in t^{\bullet} \cap {^{\bullet}t} : M(s) + W(t,s) - W(s,t) \leq K(s) \end{cases}$$

(h)If $M[t\rangle M'$, $\forall s \in S$,

$$M'(s) = \begin{cases} M(s) - W(s,t), s \in {^{\bullet}t} - {^{\bullet}t} \\ M(s) + W(t,s), s \in t^{\bullet} - {^{\bullet}t} \\ M(s) + W(t,s) - W(s,t), s \in {^{\bullet}t} \cap t^{\bullet} \\ M(s), \textit{others} \end{cases}$$

This extended PN is on the basis of prototype PN. We added the weighted function W, representing the relative proportion of F, and the capacity function K, representing the capacity of places. Due to the limitation of W and K, the occurrence of transition depends not only on its pre-set but also on W and K in the post-set. It would be more convenient to describe the preference of the flow of service objects in service systems and the capacity of each service node by EPN.

As shown above, EPN can better describe the structure, status and changes of general service systems. However, for an actual service system, the existing PN are not enough.

1) In actual service systems, service nodes often have different service processes, such as the length of the service time and the number of entities providing services. Existing PN, including Time PN theories, Timed PN and Stochastic Petri nets (SPN), cannot describe various service processes comprehensively and effectively. They are unable to provide enough theoretical models and lack versatility.
2) In traditional PN, parameters are constant. However, things in the real world are universally connected, and the nature of actual service systems will change due to changes in external parameters. PN with constant parameters can only provide support for static analysis and simulation, and it is difficult to analyze dynamically and predict in real time on the basis of changes in the external environment.

In this paper, we propose a novel perceptual PPN model, which combines queuing theory and perceptual parameters, to solve the problem of dynamic analysis and sustained prediction. It can effectively predict situation awareness.

III. OUR PPN: INTRODUCING QUEUING THEORY AND PERCEPTUAL PARAMETERS

We introduced queuing theory and perceptual parameters to reinforce existing PN model. As discussed in the Section II, in actual service systems, the service nodes of different systems often have different service processes, and existing PN theories are not sufficient to describe these processes. To solve this problem, we introduced queuing theory. Queuing theory can effectively establish models for the system with respect to random clustering phenomenon and random service process. In addition, in our discussion, the properties of the existing PN tend to be steady. When the parameters of a model do not change with external conditions, the state space generated by the model gets larger and larger, and the error between the estimated value of output and actual value is more and more obvious with derivation of the model. At the same time, the operational situation awareness of airports also requires the correlation analysis of multi-source data. In view of the lack of dynamic characteristics in the existing PN theories, we propose the concept of perceptual parameters. We will describe the queuing theory and the perceptual parameters separately.

A. QUEUING MODELS

The queuing theory was created by a Danish mathematician Erlang [33]. It was developed for computing and optimizing the efficiency of any system that achieves its objectives by consuming multiple resources optimally [34]. In our situation, it can describe a variety of service processes of a general service system. At present, the application of queuing theory is extensive. Load balancing [35], urban traffic [36], [37] and resource utilization [38] all involve queuing theory. A queuing model is generally expressed as a 6-tuple: $X/Y/Z/A/B/C$. In the above formula, X is the time interval at which customers arrive at the system; Y is the distribution of service time; Z is the number of service stations; A is the system capacity limit; B is the number of customer sources; and C is the service rule.

In civil aviation airport support systems, we can establish suitable queuing models during these three processes. Regarding the process in which aircrafts are queued for takeoff, the time aircrafts arrive at airports obeys negative index distribution and is expressed by M. Combined with the actual situation of airports, this process can be described as a queuing theory model $M/0/Z/A/\infty/FCFS$. With respect to the process of aircrafts receiving services, the customer arrival time distribution and capacity depend on the guarantee process and the service process of the upper queuing process. Therefore, X and B in the model are recorded as the determined value D. Since the data set of airport service time is large enough, a normal distribution can be used to estimate the time to accept the service, and the normal distribution is denoted as N. Thus, this process can be described as a queuing theory model $D/N/Z/A/D/FCFS$. Moreover, there are two situations for the process of aircrafts leaving the service nodes. One is the process of aircrafts receiving other different services after the previous service is completed, and it is related to the weight of the flow relations in PN. The other is the process of aircrafts taking off from the airport, and it can be described as $D/0/\infty/\infty/D/FCFS$ after analysis. Adding the queuing theory can effectively respond to the operation of the system.

B. PERCEPTUAL PARAMETERS

Perceptual parameters reflect the changes in the external conditions. For instance, for the ground support business of the aircrafts, the factors affecting the support time are mainly the date (whether it is a holiday), weather conditions and aircraft types. We define perceptual parameter set as SA, and each parameter in this set is denoted as sa. Also, sa is the mapping of the external environment E to a parameter in the model. When we deduced the model, we re-measured the parameters of the current system's state and external conditions after every other time interval and recalculated the parameter values of the model by using SA.

The problem of situation awareness in general service systems is to find out the future trend of the distribution of the service object and the congestion of the service node in the current condition. In order to make the results to be in real time and to be understood easily by decision-makers, our algorithm will output point estimations in the end. What we should do in the case of $SM(t_0)$, E is to find the point estimations of $SM(t_i)$. SM denotes the situation of the service objects waiting to accept service in various nodes; t_0 is the current time, and t_i is a moment of future. E is a set of values of external conditions that could impact system. Point estimations of situation depend directly on perceptual parameters. The specific usage and significance will be given in Section V. With the rapid development of deep learning technology in recent years, ANN has excellent performance in most fields, such as game playing system [39], computer graphics [40], [41] and natural language processing [42], [43]. In this paper, we use ANN to solve perceptual parameters.

Specifically, the macroscopic situation awareness algorithm of a general service system is shown as Algorithm 1. What differentiates microscopic and macroscopic situation awareness algorithms is that their initial state and output values are different.

Algorithm 1 The Iteration Part of Our Algorithm

```
for general service system do
    Model it with PPN;
    Use history model parameters of system as tags,
    a number of external conditions which have
    significant influence on the system as input, to train
    ANN which is solving perceptual parameters;
    repeat
        Collect current system states and external
        conditions;
        Use ANN to obtain values of perceptual
        parameters in the current system model;
        Use formulas ((1-16) mentioned in Section V) to
        calculate point estimations of macroscopic and
        microscopic situation as results;
        if (Congestion has occurred) then
            Use simulation to obtain system's situation
            values;
        end
    until Historical data accumulates to a certain
    amount, train this ANN again;
end
```

IV. PPN MODEL ESTABLISHMENT

In this section, we introduce PPN model in details. Our proposed model is defined as follows:

Definition 4:

A 8-tuple $SAPN = (SS, MS, ST, MT, F, W, M)$ satisfying the following conditions is called a PPN:

(a)$SS \cup MS = S$;

(b)$ST \cup MT = T$;

(c)$S \cup T \neq \emptyset$;

(d)$S \cap T = \emptyset$;

(e)$t \in ST : E \to Q$;

(f)$Q : (X, Y, Z, A, B, C)$;

(g)$F \subseteq (S \times T) \cup (T \times S)$;

(h)$W : (MT \times SS) \to w(E)$;

(i)$0 \leq w(E) \in R^+ \leq 1$,

$\sum_{ss_i \in mt_i^{\bullet}} W(mt_i \times ss_i) = 1$;

(j)$M : S \to R^+$

In the above definition, (a), (b), (c) and (d) describe the *place* and the *transition*. Based on the analysis of Section II and Section III, in order to describe nodes' service process more clearly and make a good combination with queuing theory, we made some improvements compared with the traditional PN theory. We classified the *places* into two categories: One is called *service place* in set SS, which contains service objects waiting to be serviced. The other is *movement place* in set MS, which contains service objects that already have received service and are preparing to go to other nodes. In PPN, tokens are expressed as service objects in SS and MS. Similarly, *transitions* are classified into two categories: One is called *service transition* in set ST, which describes the service process of nodes specifically. The other is called *movement transition* in set MT, and service objects in it have already finished receiving services and are preparing to leave the nodes. By introducing the queuing theory, (e) and (f) describe the transition, which is a mapping from the external environment E to the queuing model $Q : (X, Y, Z, A, B, C)$. (g), (h) and (i) describe the *flow relations* of net and there is a weight W during MT transiting to SS. w(weight) is a mapping from E to real numbers (0 to 1), and the sum of the weights of the same MT is 1. (j) extends the marking range from positive natural number to positive real number, which is more convenient to describe some intermediate states converging according to probability when transiting. Parameters Q and W in the system both relate to perceptual parameters.

Definition 5: Under external conditions (E), the transition firing rules of PPN are as follows:

(a)For $t \in ST$, the condition of transition $M[t\rangle$ is:

$$\begin{cases} \forall s \in {}^{\bullet}t : M(s) \geq 0 \\ \forall s \in t^{\bullet} : M(s) + \min(M({}^{\bullet}t), Z(t)) < \max(A(E, S^{\bullet})) \end{cases}$$

(b)For $t \in MT$, the condition of transition $M[t\rangle$ is:

$$\forall s \in {}^{\bullet}t : M(s) \geq 0$$

(c)For the time spent τ in the transition process:

$\tau = \max(\tau x_i + \tau y_i)\ (\tau x_i \sim X(E, t_i), \tau y_i \sim Y(E, t_i), t_i \in s^{\bullet})$

(d)For $t \in ST$, $\forall s \in S$, if $M[t\rangle M'$:

$$M'(s) = \begin{cases} M(s) - \min(M(s), Z(t)), s \in {}^{\bullet}t \\ M(s) + \min(M({}^{\bullet}t), Z(t)), s \in t^{\bullet} \\ M(s), others \end{cases}$$

(e)For $t \in MT$, $\forall s \in S$, if $M[t\rangle M'$:

$$M'(s) = \begin{cases} 0, s \in {}^{\bullet}t \\ M(s) + \sum_{s_i \in {}^{\bullet}t} M(s_i) W(E, s, s_i), s \in t^{\bullet} \\ M(s), others \end{cases}$$

As shown in (a) and (b), two types of transition rules are defined. *Place MS* of PPN is described by queuing model parameter A. If transition (a) cannot happen, it indicates that the service node has become congested. (c) explains that transition is the maximum value of the sum of the arrival time and service time in each service queue of post-set, and it is not done in an instant. The arrival time and service time obey the distribution of $X(E, t)$ and $Y(E, t)$. (d) and (e) describe the state of the system after the occurrence of two types of transitions. It should be noted that the physical meaning of *weighted function* is the probability that a service object that has finished receiving service goes to another node. Also, the physical meaning of *marking* is the value of the number of service objects in each node that converges by probability.

We modeled every service node i in the system by using PPN, including *place* ss, ms and *transition* st_i, mt. Among them, st is the mapping of E to queuing model $M/0/Z/A/\infty/FCFS$; ms is a fixed queuing model $D/0/\infty/D/D/FCFS$. Modeling a single service node using PPN is shown in Fig.2.

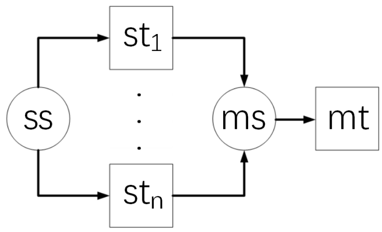


FIGURE 2. **Single service node model.**

Fig.3 describes a general service model with three service nodes; each node has a *transition* st. In Fig.3, the service object moves to node 2 and node 3 at the probability of $w_{1,2}$ and $w_{1,3}$ respectively after receiving service in node 1. $w_{1,2}$ and $w_{1,3}$ are functions related to the external condition E and the sum of $w_{1,2}$ and $w_{1,3}$ is 1.

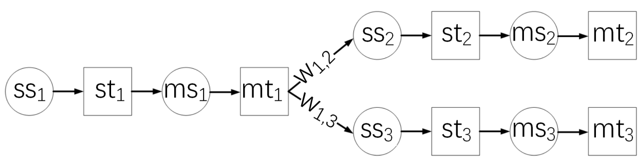


FIGURE 3. **Service process model with three service nodes 1, 2 and 3. Each node has a transition st.**

In order to describe the life-and-death process of service objects in the system, we abstracted the environment outside the system as a special service node 0 that has only one *place* and one *transition*. Service objects in the system enter from external node 0 and eventually return to it. The system shown in Fig.3 is expanded as Fig.4. Particularly, *transition* t_0 is a mapping of E to queuing model $D/N/Z/A/D/FCFS$. It should be noted that service node 0 can also be as a connector between two sub-models; then the *place* of the whole sub-model is s_0 and the *transition* is a combination of all internal transitions.

In PPN, each service node describes the concurrent relationship and the link between service nodes describes the serial relationship in the process. Therefore, this PPN can be used to quickly model various general service systems.

Some conclusions can be reached from the above discussions. For each service object, our model can describe the whole process, including entering, leaving and receiving

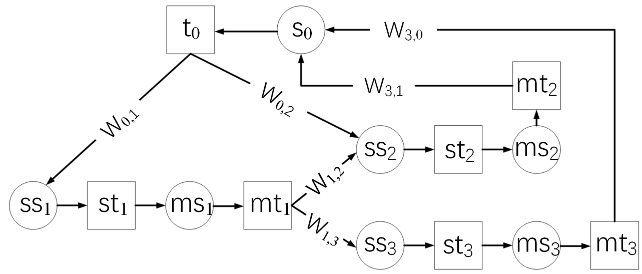


FIGURE 4. **General service system model.**

various services under the external environment E. Thus, we can obtain the estimations of time at which the object arrives at each node and accepts each service. The prediction and estimation of a single service object can be viewed as a perception of the microscopic situation of the service system. The point estimations for situation awareness will be discussed in details in the following section.

For the whole system, the model can be used to calculate the convergent value of the system state M according to the probability under certain external condition E. This can reflect the trend of systems. We can infer which nodes in the system are more congested and which nodes have abnormal traffic. This information can provide decision supports about the state of systems for decision makers. Therefore, it can be seen as a prediction of macroscopic situation of the service system.

V. PPN MODEL SOLVING

The PPN can be formalized as follows:

$$\begin{aligned} M &= [SM|MM]^T \\ SM &= [M(ss_1), M(ss_2), M(ss_3), \cdots M(ss_n)]^T \\ MM &= [M(ms_1), M(ms_2), M(ms_3), \cdots M(ms_n)]^T \end{aligned} \quad (1)$$

The vector set M contains perceptual petri elements. The sub-vector MM contains markings of ms (movement place). The sub-vector SM contains markings of ss (service place), which is the value needed to be estimated.

Specifically, situation awareness of general service systems can be expressed as follows: find the point estimations of $SM(t_i)$ in the case that $SM(t_0)$ and E are known. Among them, t_0 is the current time; t_1 is a certain moment in the future; E is a set of values of external conditions. In particular, the microscopic situation awareness can be expressed as follows: find the point estimations of node t_1 in the case that $SM(t_0)$ and E are known when transiting to a different $SM'(t_1)$. Here $SM(t_0) = [1, 0, 0, ...0]^T$.

We will discuss how to solve the PPN, i.e., how to find the point estimations in details in this section. Before this, the method using ANN for solving perceptual parameters of PPN will be presented. Firstly, we used ANN technology to obtain the perceptual parameters W and Q. Fig.5 is a schematic diagram of a neural network for obtaining perceptual parameters. As shown in this figure, the input layer

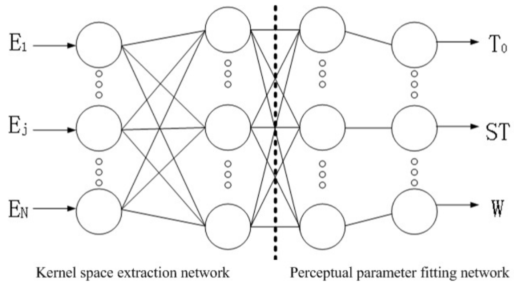


FIGURE 5. Use ANN to obtain perceptual parameters.

contains various external conditions, which are quantified and normalized. The output layer contains parameters of PPN and mainly includes parameter λ, which is shown as negative exponential distribution in queue T_0, parameters $\mu and \sigma$, which are shown as normal distribution in service queue ST, and W (weight) of the network. Because of the feature of many parameters and hidden layers, the ANN can be optimized in a specific calculation. For example, we used the data that is fitted with ARIMA (Autoregressive Integrated Moving Average Model) instead of the original data to eliminate some factors that may have a bad influence on ANN. Moreover, the parameters in the network are pre-trained layer by layer to obtain better initial parameter values [44], which helps to get the optimal values. The ReLU(Rectified Linear Unit) is also used to optimize the ANN and speed up the training [45].

We used history environment data about airports as input, history perceptual parameters obtained through statistics of history data as tags, to train the neural network shown in Fig.5. Then we can obtain a model that can be used to find new perceptual parameters. This model can be used as a black box in our algorithm.

A. MACROSCOPIC SITUATION SOLVING

Next, we solved the macroscopic situation parameters. According to our Definition 5, the transition in SM depends on the speed of service in service queuing and the number of objects waiting for service regardless of system congestion (the number of objects waiting for service is greater than the capacity of the service nodes, or some services cannot be in place as planned). When the external conditions were certain, we inferred the process of service objects entering the service node from outside:

$$\begin{aligned} SM(t_0+\Delta t) &= SM(t_0)+Q(\infty, s_0, \Delta t)\times\left[w_{0,1}, w_{0,2}\cdots w_{0,n}\right]^T \\ &= SM(t_0)+Q(\infty, s_0, \Delta t)\times T_{in} \end{aligned} \tag{2}$$

Δt is the time interval. T_{in} is the form of the matrix of the weight element W, which describes the probability that service objects enter each node of the system from outside. It is a perceptual parameter that changes with external environment. n is the number of nodes in PPN. Q is the function of obtaining the service number of queue nodes. Specifically, $Q(\infty, s_0, \Delta t, E)$ represents the number of transition objects in *place* s_0 with infinite number of service objects in Δt. How can we obtain the point estimation of Q?

Based on the discussion above, the queue from the external environment to the system is model $M/0/Z/A/\infty/FCFS$. The number of arrivals is consistent with the Poisson distribution, and the mathematical expectation is taken as the point estimation:

$$\overline{Q(\infty, s_0, \Delta t, E)} = \sum_{0}^{\infty} kp(k) = \sum_{0}^{\infty} k\frac{\lambda^k}{k!}e^{-\lambda} = \lambda \tag{3}$$

Among them, λ is also a perceptual parameter that changes with environment.

$$\begin{aligned} SM(t_0+\Delta t) &= SM(t_0)+MM(t_0)\times\begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,n} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n,1} & w_{n,2} & \cdots & w_{n,n} \end{bmatrix} \\ &= SM(t_0)+MM(t_0)\times T_m \end{aligned} \tag{4}$$

For MT, we only considered the transition process of objects between service nodes, shown as (4). T_m is the form of the matrix of the weight element W, which is a perceptual parameter that changes with the external environment.

According to the rules of transition, the service object must leave *place mm* to transit after receiving service, then $MM(t_0+\Delta t)$ only depends on the number of services completed in Δt:

$$MM(t_0+\Delta t) = Q(SM(t_0), ST, \Delta t) \tag{5}$$

On the basis of the discussion above, the service process in the service node is queuing model $D/N/Z/A/D/FCFS$. We took the mathematical expectation of $Q(SM(t_0), ST, \Delta t)$ as its point estimation. For node i:

$$\begin{aligned} &\overline{Q(sm_i(t_0), st_i, \Delta t, E)} \\ &\quad= \sum_{k=0}^{SM(t_0)} kp(k) \\ &\quad= \sum_{k=0}^{SM(t_0)} k\int_0^{\Delta t-\sum_{i=1}^{n-1} st_i} \prod\left(2\pi\sigma^2\right)^{-\frac{1}{2}} \\ &\qquad \exp\left(-\frac{(st_i-\mu)^2}{2\sigma^2}\right)dst_n dst_{n-1}\cdots dst_1 \end{aligned} \tag{6}$$

σ, μ are perceptual parameters. Let st_i denote the time of the i-th service and k denote the number of times the queue completes the service. In actual calculation, we can get the following formula after stipulating the integral step:

$$\begin{aligned} &\overline{Q(sm_i(t_0), st_i, \Delta t, E)} \\ &= \sum_{k=0}^{SM(t_0)} kp(k) \\ &= \sum_{k=0}^{SM(t_0)} k \sum_{st_1'=0}^{\frac{\Delta t}{\Delta t'}} \sum_{st_2'=0}^{\frac{\Delta t}{\Delta t'}-st_1'} \cdots \sum_{st_i'=0}^{\frac{\Delta t}{\Delta t'}-\sum_{i=1}^{n-1} st_i'} \prod\left(2\pi\sigma^2\right)^{-\frac{1}{2}} \exp\left(-\frac{(st_i-\mu)^2}{2\sigma^2}\right) \end{aligned} \tag{7}$$

Δt controls the step length of integration accuracy. $\left(2\pi\sigma^2\right)^{-1/2}\exp\left(-(st_i-\mu)^2/2\sigma^2\right)$ is the density function of normal distribution, and its value can be obtained quickly by checking the normal distribution table written into the memory beforehand. It is worth noting that when $sm_i\left(t_0\right)$ is lager, the mathematical expectation of $Q\left(SM\left(t_0\right), ST, \Delta t\right)$ is too difficult to calculate. Usually, it can be simplified. We took μ as the point estimation of each service time, and the estimation of the number of objects receiving service is expressed as $1/\mu$ per unit time. The expression is as follows:

$$\overline{Q\left(sm_i\left(t_0\right), st_i, \Delta t, E\right)} = \max\left(sm\left(t_0\right), \frac{\Delta t}{\mu}\right) \tag{8}$$

When the post-set of *place ss* contains more than one transition, the service speed is limited by the slowest service process, i.e., the number of transitions of service objects in *ss* is equal to the number of transitions with the minimum number of services. The expression is as follows:

$$\overline{Q(sm_i(t_0), ss_i, \Delta t, E)} = \min \overline{Q(sm(t_0), st_j, \Delta t, E)}, st_j \in ss_i^{\bullet} \tag{9}$$

In fact, $SM\left(t_0+\Delta t\right)$ contains four parts, which are the value of $SM\left(t_0\right)$, the value of new entering outside the system, the value moved from other nodes and the value of transition.

When Δt is small enough, most of the service objects can be considered to have only completed a single transition, i.e., the above four parts can be considered as serial occurrences. We have the following expression:

$$\begin{aligned} &SM(t_0+\Delta t)\\ &\quad = SM(t_0)+Q(s_0, mt_0, \Delta t, E)\times T_{in}\\ &\qquad + Q(SM(t_0), ST, \Delta t, E)\times T_m - Q(SM(t_0), ST, \Delta t, E)\\ &\quad = SM(t_0)+Q(s_0, mt_0, \Delta t, E)\times T_{in}\\ &\qquad + Q(SM(t_0), ST, \Delta t, E)\times(T_m - I) \end{aligned} \tag{10}$$

In the above expression, I is the unit matrix. Taking the corresponding point estimation of function $Q\left(\infty, s_0, \Delta t, E\right)$ and $Q\left(SM\left(t_0\right), ST, \Delta t\right)$, we can obtain the point estimation of the macroscopic situation of the system. When Δt is far greater than the time of various transitions, we need to get $\Delta t'$, which is smaller, and use formula (10) to calculate $SM\left(t_0+\Delta t'\right)$. By iterating the above calculation $\Delta t/\Delta t'$ times, we can find the point estimation.

Particularly, when congestion occurs, the following formula will be calculated:

$$\exists t_i \in ST, sm\left(t_i\right) > a\left(t_i\right) \tag{11}$$

Formula (11) is used to express congestion, whose physical meaning is that when the time is t_i, the number of markings stored in *place sm* is larger than the capacity of the queue. When the macroscopic situation estimation of the system satisfies formula (11), managers need to deal with the congestion nodes timely and recalculate with our model before the moment of congestion.

An example is given to better illustrate our algorithm. Suppose we model the service system shown in Fig.4 by PPN. When $t_0 = 0$, $SM(t_0) = [0, 1, 2]$. We assume that the perceptual parameters have been calculated by the trained ANN: $\lambda_0 = 3$; $\mu_1 = \mu_2 = \mu_3 = 2$; $\delta_1 = \delta_2 = \delta_3 = 1$; $\omega_{0,1} = \omega_{0,2} = \omega_{1,2} = \omega_{1,3} = 0.5$; $\omega_{2,0} = \omega_{3,0} = 1$. Then, the macroscopic situation when $t_1 = 1$ is:

$$\begin{aligned} &SM(t_1)\\ &\quad = [0, 1, 2]^T + Q(s_0, mt_0, \Delta t, E)\times[0.5, 0.5, 0]^T\\ &\qquad + Q(SM(t_0), ST, \Delta t, E)\times(\begin{bmatrix}0 & 0.5 & 0.5\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix} - I) \end{aligned} \tag{12}$$

The point estimation of $Q(s_0, mt_0, \Delta t, E)$ calculated using Formula (3) is 3. The point estimation of $Q(SM(t_0), ST, \Delta t, E)$ calculated by Formula (8) is $[0, 1, 1]^T$. Add these two estimations into Formula (12):

$$\begin{aligned} &SM(t_1)\\ &\quad = [0, 1, 2]^T + 3\times[0.5, 0.5, 0]^T + [0, 1, 1]^T\\ &\qquad \times(\begin{bmatrix}0 & 0.5 & 0.5\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix} - \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}) = [1.5, 1.5, 1]^T \end{aligned} \tag{13}$$

Thus this result represents the value of the distribution converging in probability of the service object when $t = t_1$, i.e., the point estimations of the macroscopic situation are obtained.

B. MICROSCOPIC SITUATION SOLVING

Solving microscopic situation parameters is similar to the macroscopic situation parameters. In the case of no congestion, PPN can easily provide the mathematical expectation of the time at which a specific object receives each service. That is the point estimation of the service time of each node. Based on these estimations, it is easy to find the point estimations of specific time nodes in the process of transition.

Take the PPN model of Fig.4 as an example. The point estimation of leaving the system of a service object that just entered can be inferred as follows:

$$\begin{aligned} &Time(SM_{start}\,[T\rangle SM_{end})\\ &\quad = w_{0,1}w_{1,2}(Time([st_1\rangle)+Time\,([st_2\rangle))\\ &\qquad + w_{0,1}w_{1,3}(Time([st_1\rangle\,)+Time([st_3\rangle))\\ &\qquad + w_{0,2}(Time([st_2\rangle))+t_{start} \end{aligned} \tag{14}$$

In the above formula, t_{start} is the time of entering the system, $Time\,([st_i\rangle$ is a function to solve the time at which the *transition* st_i occurs; $SM_{start} = [1, 0, 0, \cdots 0]^T$ is the state of a service node just entering the system; and $SM_{end} = [0, 0, 0, \cdots 0]^T$ is the state of a service node just leaving the system. Take the mathematical expectation of $Time\,([st_i\rangle$ as perceptual parameter μ_i to obtain the point estimation of $Time\ (SM_{start}\ [T\rangle SM_{end}\ )$. The expression is shown as follows:

$$Time\ (SM_{start}\ [T\rangle SM_{end}\ ) = w_{0,1}w_{1,2}\ (\mu_1 + \mu_2) + w_{0,1}w_{1,3}\ (\mu_1 + \mu_3) + w_{0,2}\ (\mu_2) + t_{start} \quad (15)$$

The above analysis is based on the case of no congestion. When the point estimation of the macroscopic situation satisfies formula (11), the system's ability to support a single service object has almost been lost, and it is meaningless to analyze the microscopic situation.

Similarly, to better illustrate our algorithm, one example is given below. Let's continue with the previous one. Assuming that there is a node that enters the system at $t = 0$, then the time point of the node leaving is as follows:

$$\begin{aligned} & Time(SM_{start}[T > SM_{end}) \\ & = w_{0,1}w_{1,2}(Time([st_1 >) \\ & \quad + Time([st_2 >)) + w_{0,1}w_{1,3}(Time([st_1 >) \\ & \quad + Time([st_3 >)) + w_{0,2}(Time([st_2 >)) \\ & = 0.5\times0.5\times(2+2)+0.5\times0.5\times(2+2)+0.5\times2+0=3 \end{aligned} \quad (16)$$

Thus, the point estimations of the microscopic situation are obtained.

VI. EXPERIMENTAL RESULTS

A. SITUATION AWARENESS IN CIVIL AVIATION AIRPORTS

To verify the validity of our method, we specifically discussed the microscopic situation awareness of ground supports in civil aviation airports in this section. We used the data set of Hefei Xinqiao International Airport in 2016 to evaluate our method and compare it with A-CDM, which is the most advanced airport operation management system used to solve a series of problems, such as route congestion and flight delays that affect the efficiency of flight operations.

In civil aviation airport operation control system, the ground support system of the aircraft is the most important. Several key time nodes in the ground support service are specified in the existing A-CDM system. For example, the Target Off Block Time (TOBT) is provided to the air traffic control as the basis for aircrafts to take off; the Calculated Off Block Time (COBT) predicted by the TOBT is the latest completion time of the ground support that needs to be guaranteed; the Target Take Off Time (TTOT) determines the time of flights' launch, push back and take off. Through these key time nodes, the flight guarantee process can be planned, targeted, and controlled, and this is able to improve the operation efficiency of the airport.

However, as mentioned earlier, the determination of special time nodes in the A-CDM is more of a plan than a prediction. Firstly, the determination of each target time (such as TOBT, TTOT) requires the three-way negotiation among airports, airlines, and air traffic control instead of an automatic prediction. Secondly, each calculated time (such as COBT) is only obtained by adding or subtracting the corresponding relaxation time on the basis of the target time, which is not accurate enough. Moreover, a large-scale air traffic flow management system has not been established in many places, and the A-CDM systems in each region can only rely on themselves, which further reduces the accuracy of airport predictions for various time nodes.

Therefore, the prediction of special time nodes in flight support has important practical significance, but it has not been well realized in the current system. Based on the theory proposed in this paper, the airport's ground support system can be easily abstracted as a general service system model, and the problems of special time nodes can be abstracted into the microscopic situation awareness problem of the system. Several key specific time nodes that are related to the ground support system are shown in Table1. In this table, the time between two associated nodes is the time when aircrafts receive support services. For example, it takes aircrafts the time between node 1269 and 508 to receive catering services. According to the discussion above, each support service in the ground support can be abstracted as an element in the *service transition* set ST. Support services that can be carried out at the same time have the same *service place ss* and the same *movement place ms*. Based on the modeling method discussed above, we can get the PPN model as shown in Fig.6. However, $st_{1.2}$ and $st_{3.2}$ are almost not recorded in the actual data set of Hefei Xinqiao International Airport, so they are not considered in this experiment. By the way, $st_{1.1}$ does not belong to support service, so it is not considered either.

TABLE 1. Definition of specific time nodes.

Node Number	Node Name
507	cleaning completed
508	catering completed
509	refueling completed
514	loading baggage completed
518	removing chocks
1200	check-in completed
1201	boarding
1202	boarding completed
1247	check-in
1267	calculated off block time (COBT)
1269	catering
1274	cleaning
1275	loading baggage
1279	refueling

For the ground support time t of the aircrafts, when the system is operating normally, the situation point estimation mentioned above is as follows:

$$t = max(\mu_{1.1}, \mu_{1.2}) + \mu_2 + max(\mu_{3.1}, \mu_{3.2}, \mu_{3.3}, \mu_{3.4}) + \mu_4 \quad (17)$$

The perceptual parameters above in order are the mathematical expectation of the time taken by the check-in, fueling, boarding bridge completed, passenger registration, cargo loading, catering, cleaning, and chocks removing. Considering the actual situation with no congestion,

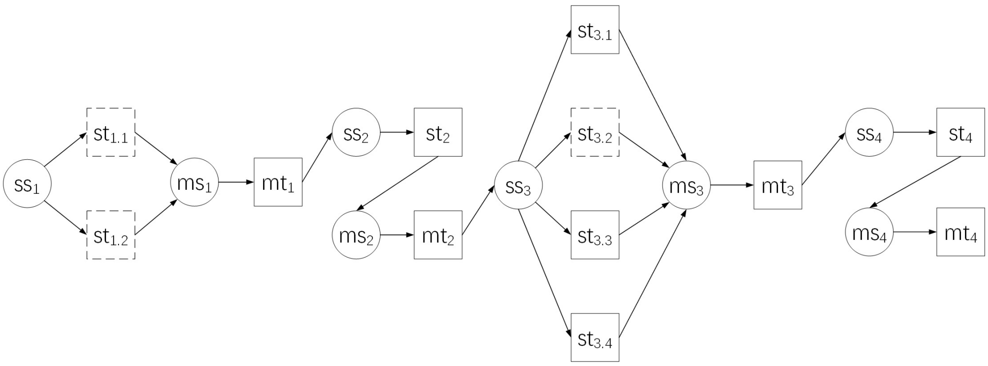


FIGURE 6. Airport ground support model($st_{1.1}$: check-in; $st_{1.2}$: refueling; st_2: before boarding;$st_{3.1}$: boarding; $st_{3.2}$: loading baggage; $st_{3.3}$: catering; $st_{3.4}$: cleaning; st_4: before removing chocks).

the ground support time t, which is from the end of the check-in to the chocks removing, is as follows (If congestion occurs, use simulation based on specific business rules of the airport to obtain results):

$$t' = \mu_2 + max(\mu_{3.1}, \mu_{3.3}, \mu_{3.4}) + \mu_4. \tag{18}$$

After modeling, we need to train ANN to solve perceptual parameters. We have concluded that time (during holidays and weekends, more time may be spent on check-in and loading baggage than usual), weather and the type of airplane have a significant influence on the support time. Therefore, we used these three external conditions as input parameters of ANN, and these data are evenly normalized to 0 to 1. The output of ANN is the perceptual parameters of the PPN. Specifically, the difference between the end time and the start time of each service is the time spent, and we can further calculate the average time and variance of each service.

In the training of this ANN, we randomly selected 10% of the data as a test set to verify the training effect and repeat it many times. One training result is shown in Table2. Among them, the average accuracy is calculated as $(1 - MAE/MAST) * 100\%$(MAE: mean absolute error; MAST: mean actual support time).

TABLE 2. Training Results of Perception Parameter(μ_2: preparing time of boarding bridge; $\mu_{3.1}$: boarding time; $\mu_{3.3}$: catering time; $\mu_{3.4}$: cleaning time; μ_4: time before removing chocks).

Perception Parameter	MAE(min)	MAST(min)	Average Accuracy
μ_2	2.24998	18.8108	88.039%
$\mu_{3.1}$	3.22633	19.6966	83.620%
$\mu_{3.3}$	2.52471	15.1924	83.382%
$\mu_{3.4}$	2.96283	16.4037	81.938%
μ_4	3.22769	17.1842	81.217%

In our experiment of specific time nodes' prediction, we used our formulas for the airport ground support time for each flight in December 2016 and compared the results with A-CDM. The results are shown in Table3, Fig.7 and Fig.8. Fig.7 shows the support rate of A-CDM, and Fig.8 shows the support rate of PPN. In this regard, "fail to support" means that the actual support time is longer than the predicted time; "completely support" means that the actual time is within the range of the predicted time, and "error" means that the difference between the actual time and predicted time is within 1 min. Actually, if the flight fails to remove chocks within the specified time, it will greatly miss the resources, such as the route allocated in advance, resulting in the flight delay.

TABLE 3. Predicted Results of Support Time.

	PPN	A-CDM	Actual Value
Average Support Time(min)	55.692	48.742	58.866
Average Predicted Error(min)	13.407	17.890	0
Average Accuracy	77.23%	69.61%	100%

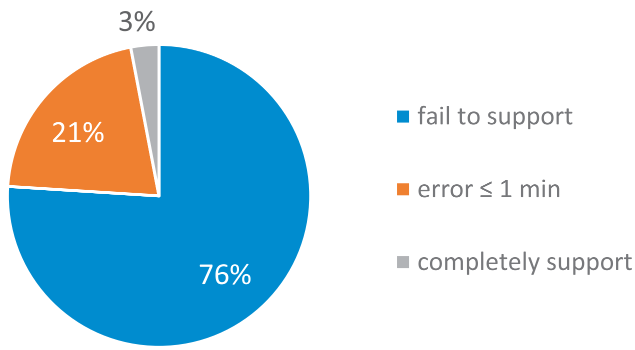


FIGURE 7. Support rate of A-CDM.

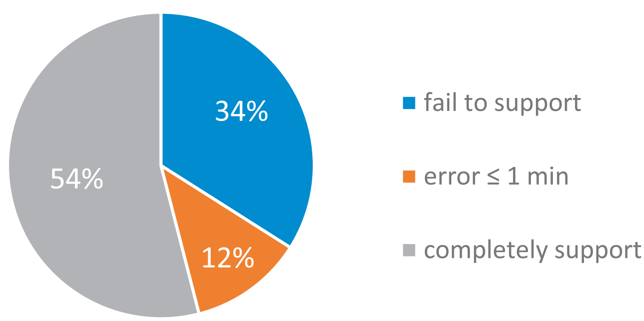


FIGURE 8. Support rate of PPN.

The experimental results show that the ground support time given by PPN point estimation algorithm is more accurate than that of A-CDM, and the average accuracy rate is improved by 7.62%. Since the A-CDM system does not consider the influence of external conditions, the predicted time it gives is much lower than the actual time. In contrast, the predicted time calculated by our method is closer to the actual value. In other words, the aircraft can remove chocks during the specified time, thereby increasing the on-time rate of the flight. We believe that if the planning and management of the aircraft support business is carried out according to the point estimated value given by PPN, the accuracy of the predicted time of various ground support will be greatly improved. Our method can provide effective decision support for civil aviation airport managers, and greatly increase the efficiency of existing system.

B. SITUATION AWARENESS IN ACCESS OF WEBSITE

The above experiment proves that PPN theory and its algorithm have good performance on the microscopic situation awareness problems. To better verify the effectiveness of our method for macroscopic problems and its applicability among general service systems, we further experimented on the access of website. We evaluated our method and compared it with SPN. SPN is a time PN model with negative exponential transition time and is the closest one to our PPN. We used a set of scientific research data to discuss situation awareness of website traffic. The web browsing data used in this experiment was shared by Laurence Berkeley National Laboratory and comes from the Kennedy Space Center website in July and August 1995 [46]. This data set contains a total of 3461612 requests, and the time accuracy is 1s.

Same as the first experimental step, we abstracted the related services in web browsing as a general service system and modeled it using PPN. To be specific, each visitor is a service object; each part of the website is the service node of the system, and the response of various requests to the web is the service. We took visitors, bounce rate and exit rate as items that the macroscopic situation needs to perceive. In addition, we regarded residence time on web page, residence time on site and conversion rate as items that the microscopic situation needs to perceive. According to the frequency of visits and reasonable estimates, we divided the site into three parts: home page (contains welcome page and various navigations to other pages), launching page (page about space launching information, reports and history) and all remaining pages (such as technical reports, software downloads and financial information). The traffic in these three regions accounts for approximately 20%, 65% and 15%. Based on the modeling method discussed above, we obtained the PPN model as shown in Fig.9.

From the statistics of website traffic, we concluded that missions of space shuttle, hurricanes and dates have a significant influence on access behavior. Therefore, we used these three external conditions as input parameters of the neural network. In addition, the statistical results of the training set, namely historical perceptual parameters including the transfer matrix parameters and queuing nodes parameters, are evenly normalized to 0 to 1, which can be used as tags of the neural network. In the macroscopic situation awareness, we predicted the website traffic and the number of individual visitors to each part in the last week of the data set. In this experiment, we stipulate that the prediction time step size is one hour, and we will update the quantity of access objects in each part every hour. Besides, the perceptual parameters will be updated every hour based on the training model. The experimental results are as follows:

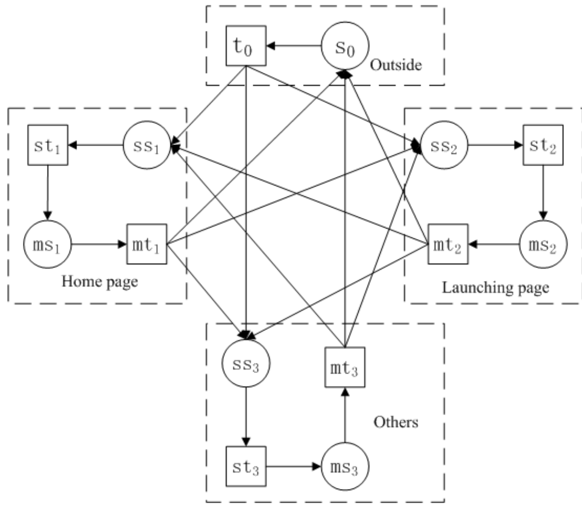


FIGURE 9. PPN model of website.

It can be seen in Fig.10, Fig.11, Table4 and Table5 that the prediction results of the SPN show obvious hysteresis without updating parameters of the model. In contrast, the PPN model estimates the perceptual parameters in the next hour in advance using neural network; thus, the final prediction results do not appear to be delayed in time as the SPN. Obviously, the prediction accuracy of PPN is significantly higher. It should be noted that the improvement in the accuracy of PPN in the home page is relatively not obvious, because the traffic of service objects in this page is small, which is not

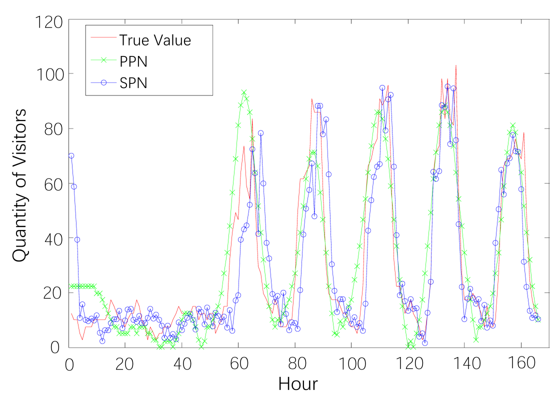


FIGURE 10. Prediction of home page.

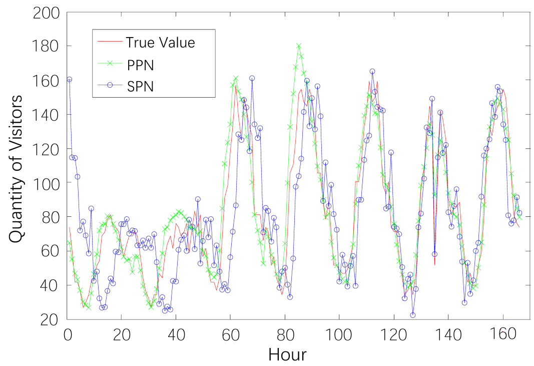


FIGURE 11. Prediction of launching page.

TABLE 4. Prediction errors of residence time (MSE: Mean Square Error; MAE: Mean Absolute Error; RMSE: Root Mean Square Error).

Model	MSE	MAE	RMSE
PPN	85.8844	7.1703	9.2674
SPN	257.7302	10.3382	16.0540

TABLE 5. Prediction errors of launching page (MSE: mean square error; MAE: mean absolute error; RMSE: root mean square error).

Model	MSE	MAE	RMSE
PPN	255.3232	11.6154	15.9788
SPN	777.4369	21.2364	27.8826

well suited to our discussion. Conversely, in areas with large traffic flows, such as the launching page, the PPN shows excellent performance advantages than the SPN.

PPN can also perceive the microscopic situation of the website. We calculated the average time of each visitor's residence time per hour in the last week of the data set, and predicted it using PPN and SPN. The experimental results are shown in Fig.12 and Table6. Even though the randomness of the access behavior of individual visitors is too strong to analyze the special time nodes, the effects are not as obvious as the first experiment, PPN still achieves better results than SPN. The point estimations of PPN can reflect the trend of residence time, which is expressed as a relatively smooth prediction curve. In fact, the point estimations of PPN are similar to the envelope of real value in which the window

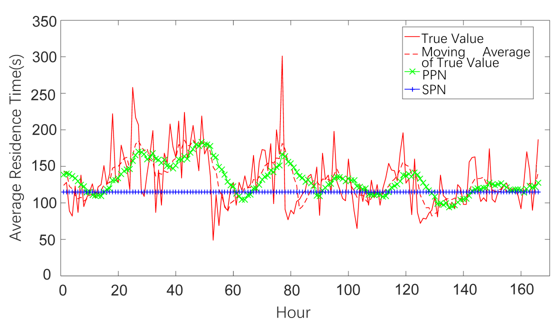


FIGURE 12. Prediction of residence time.

TABLE 6. Prediction errors of residence time (MSE: mean square error; MAE: mean absolute error; RMSE: root mean square error).

Model	MSE	MAE	RMSE
PPN	222.5069	14.9167	14.9167
SPN	1797.1	30.6016	42.3924

size is 6 hours. For the SPN model, since it does not update parameters such as transition time, its point estimations for residence time can only be constants, and they do not have a more practical meaning. The second supplementary experiment proves that our method has good generality. At the same time, experimental results verify that our PPN and algorithm have excellent performance both in macroscopic and microscopic situation awareness.

VII. CONCLUSION

Based on the big data operating environment of an airport, we proposed a situation awareness method for airports. The model constructed in this paper can comprehensively reflect the general service system to describe the service process of airports. This model introduces queuing theory and perceptual parameters on the basis of PN to describe service process and responds to external conditions better. Combined with ANN, we propose PPN to analyze and solve the situation awareness of general service systems. The experiment results verify that the model and its situation estimation algorithm have better performance than A-CDM. In addition, the experiment on the access of website also proves that our model has good generality and flexibility, so it can give more accurate predictions for service system and provide support for decision makers. PPN can only easily describe individual systems, but it cannot effectively reflect the relationship and cooperation between these individual systems. Based on our research, we will consider extending the scope of situation study and research on more kinds of queuing models and external conditions. In addition, we will further optimize and improve the methods proposed in this paper in the future.

REFERENCES

[1] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," *Hum. Factors*, vol. 37, pp. 32–64, Sep. 1995.
[2] M. Preuss and S. Thomas, "Wireless, mesh & ad hoc networks; military convoy location and situation awareness," in *Proc. IEEE Sarnoff Symp.*, Apr. 2008, pp. 1–5.
[3] J. Głowacka, J. Krygier, and M. Amanowicz, "A trust-based situation awareness system for military applications of the Internet of Things," in *Proc. IEEE 2nd World Forum Internet Things (WF-IoT)*, Dec. 2015, pp. 490–495.
[4] M. Panteli and D. S. Kirschen, "Situation awareness in power systems: Theory, challenges and applications," *Electr. Power Syst. Res.*, vol. 122, pp. 140–151, May 2015.
[5] X. He, R. C. Qiu, Q. Ai, L. Chu, X. Y. Xu, and Z. N. Ling, "Designing for situation awareness of future power grids: An indicator system based on linear eigenvalue statistics of large random matrices," *IEEE Access*, vol. 4, pp. 3557–3568, 2016.
[6] J. B. Lai, H. Q. Wang, and L. Zhu, "Study of network security situation awareness model based on simple additive weight and grey theory," in *Proc. IEEE Int. Conf. Comput. Intell. Secur.*, vol. 2, Nov. 2006, pp. 1545–1548.

[7] G. Settanni, Y. Shovgenya, F. Skopik, R. Graf, M. Wurzenberger, and R. Fiedler, "Correlating cyber incident information to establish situational awareness in critical infrastructures," in *Proc. IEEE 14th Annu. Conf. Privacy, Secur. Trust (PST)*, Dec. 2016, pp. 78–81.
[8] R. Davidrajuh and B. Lin, "Exploring airport traffic capability using Petri net based model," *Expert Syst. Appl.*, vol. 38, no. 9, pp. 10923–10931, 2011.
[9] R. Mori, "Aircraft ground-taxiing model for congested airport using cellular automata," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 180–188, Mar. 2013.
[10] T. Nagatani, "Nonlinear-map model for the control of an airplane schedule," *Phys. A, Stat. Mech. Appl.*, vol. 392, no. 24, pp. 6545–6553, 2013.
[11] J. Chen and K. Xie, "A prediction model based on unbiased grey Markov for airport energy consumption prediction," in *Proc. IEEE Chin. Autom. Congr. (CAC)*, Nov. 2013, pp. 291–294.
[12] Y. J. Kim, S. Choi, S. Briceno, and D. Mavris, "A deep learning approach to flight delay prediction," in *Proc. IEEE/AIAA 35th Digit. Avionics Syst. Conf. (DASC)*, Sep. 2016, pp. 1–6.
[13] S. M. Li, X. H. Xu, and L. H. Meng, "Forecasting of airport congestion level based on cluster and neural network algorithms," *Comput. Eng. Appl.*, vol. 49, no. 17, pp. 245–257, 2013.
[14] P. P. Wu, J. Pitchforth, and K. Mengersen, "A hybrid queue-based Bayesian network framework for passenger facilitation modelling," *Transp. Res. C, Emerg. Technol.*, vol. 46, pp. 247–260, Sep. 2014.
[15] S. Corrigan, L. Mårtensson, A. Kay, S. Okwir, P. Ulfvengren, and N. McDonald, "Preparing for airport collaborative decision making (A-CDM) implementation: An evaluation and recommendations," *Cognition, Technol. Work*, vol. 17, no. 2, pp. 207–218, 2015.
[16] C. A. Petri, "Kommunikation mit automaten," Ph.D. dissertation, Institut Instrumentelle Mathematik, Bonn, Germany, 1962, vol. 3.
[17] J. M. Proth, "Petri nets," in *Handbook on Architectures of Information Systems*. Berlin, Germany: Springer, 1998, pp. 133–151.
[18] T. Murata, "Petri nets: Properties, analysis and applications," *Proc. IEEE*, vol. 77, no. 4, pp. 541–580, Apr. 1989.
[19] D. You, S. G. Wang, W. Z. Dai, W. H. Wu, and Y. S. Jia, "An approach for enumerating minimal siphons in a subclass of Petri nets," *IEEE Access*, vol. 6, pp. 4255–4265, 2017.
[20] S. G. Wang, Y. Dan, and M. C. Zhou, "A necessary and sufficient condition for a resource subset to generate a strict minimal siphon in S^4PR," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 4173–4179, Aug. 2017.
[21] O. Karoui, Z. W. Li, N. Q. Wu, M. Khalgui, E. A. Nasr, and A. M. El-Tamimi, "One-step control-ahead approach for the design of an optimal Petri-net based deadlock prevention policy," *IEEE Access*, vol. 6, pp. 34307–34323, 2018.
[22] N. Ran, H. Y. Su, and S. G. Wang, "An improved approach to test diagnos ability of bounded Petri nets," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 2, pp. 297–303, Apr. 2017.
[23] Z. Y. Jiang, Z. W. Li, N. Q. Wu, and M. C. Zhou, "A Petri net approach to fault diagnosis and restoration for power transmission systems to avoid the output interruption of substations," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2566–2576, Sep. 2017.
[24] X. Yin and S. Lafortune, "On the decidability and complexity of diagnosability for labeled Petri nets," *IEEE Trans. Autom. Control*, vol. 62, no. 11, pp. 5931–5938, Nov. 2018.
[25] N. Ran, H. Y. Su, A. Giua, and C. Seatzu, "Codiagnosability analysis of bounded Petri nets," *IEEE Trans. Autom. Control*, vol. 63, no. 4, pp. 1192–1199, Apr. 2018.
[26] J. L. Luo, W. M. Wu, H. Y. Su, and J. Chu, "Supervisor synthesis for enforcing a class of generalized mutual exclusion constraints on Petri nets," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 6, pp. 1237–1246, Nov. 2009.
[27] N. Ran, S. G. Wang, and W. H. Wu, "Event feedback supervision for a class of Petri nets with unobservable transitions," *IEEE Access*, vol. 6, pp. 6920–6926, 2018.
[28] K. L. Espensen, M. K. Kjeldsen, and L. M. Kristensen, "Modelling and initial validation of the DYMO routing protocol for mobile ad-hoc networks," in *Proc. Int. Conf. Appl. Theory Petri Nets*. Berlin, Germany: Springer, 2008, pp. 152–170.
[29] A. Giua and C. Seatzu, "Modeling and supervisory control of railway networks using Petri nets," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 3, pp. 431–445, Jul. 2008.
[30] S. H. Zegordi and H. Davarzani, "Developing a supply chain disruption analysis model: Application of colored Petri-nets," *Expert Syst. Appl.*, vol. 39, no. 2, pp. 2102–2111, 2012.
[31] F. A. X. Da Mota, M. X. Rocha, J. J. P. C. Rodrigues, V. H. C. De Albuquerque, and A. R. De Alexandria, "Localization and navigation for autonomous mobile robots using Petri nets in indoor environments," *IEEE Access*, vol. 6, pp. 31665–31676, 2018.
[32] X. Z. Zhang, Y. Y. Du, L. Qi, and H. C. Sun, "An approach for repairing process models based on logic Petri nets," *IEEE Access*, vol. 6, pp. 29926–29939, 2018.
[33] A. K. Erlang, "The theory of probabilities and telephone conversations," *Nyt Tidsskrift Mat. B*, vol. 20, pp. 33–39, 1909.
[34] B. Gnedenko and I. Kovalenko, *Introduction to Queuing Theory* (Mathematical Modeling). Boston, MA, USA: Birkhaeuser, 1989.
[35] M. Fahimi and A. Ghasemi, "Joint spectrum load balancing and handoff management in cognitive radio networks: A non-cooperative game approach," *Wireless Netw.*, vol. 22, no. 4, pp. 1161–1180, May 2016.
[36] M. Woelki, "Queuing model of a traffic bottleneck with bimodal arrival rate," *Int. J. Mod. Phys. B*, vol. 30, no. 15, 2016, Art. no. 1541001.
[37] R. Cominetti and J. Correa, "Common-lines and passenger assignment in congested transit networks," *Transp. Sci.*, vol. 35, no. 3, pp. 250–267, 2001.
[38] S. Belciug and F. Gorunescu, "Improving hospital bed occupancy and resource utilization through queuing modeling and evolutionary computation," *J. Biomed. Inform.*, vol. 53, pp. 261–269, Feb. 2015.
[39] D. Silver *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.
[40] Z. Cui, H. Chang, S. G. Shan, B. N. Zhong, and X. L. Chen, "Deep network cascade for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 49–64.
[41] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra. (2015). "DRAW: A recurrent neural network for image generation." [Online]. Available: https://arxiv.org/abs/1502.04623
[42] D. Bahdanau, K. Cho, and Y. Bengio. (2014). "Neural machine translation by jointly learning to align and translate." [Online]. Available: https://arxiv.org/abs/1409.0473
[43] Z. Cui, H. Chang, S. G. Shan, B. N. Zhong, and X. L. Chen, "Deep network cascade for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 49–64.
[44] D. Erhan, Y. Bengio, A. Courville, P. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *J. Mach. Learn. Res.*, vol. 11, pp. 625–60, Feb. 2010.
[45] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.
[46] NASA-KennedySpaceCenter. *Nasa-HTTP*. Accessed: Aug. 5, 1995. [Online]. Available: http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html

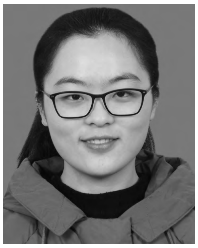
QIAN LUO received the Ph.D. degree from Sichuan University, Chengdu, China. He is currently with the Second Research Institute, Civil Aviation Administration of China. He serves as an expert in the field of information technology and has accomplished several key national projects on aviation area. He is a Researcher and has published many academic papers.

YANRU CHEN received the master's degree in management (finance) from the University of Melbourne, Australia. She is currently pursuing the Ph.D. degree with the Embedded Systems and Big Data Management Laboratory and the Internet of Things Laboratory, School of Computer Science, Sichuan University, under the guidance of Prof. B. Guo. Her research interests include wireless sensor networks, cognitive psychology, personal financial big data, computational finance, and financial data space.

LUNYUE CHEN received the master's degree in science from the College of Computer Science, Sichuan University, in 2017, where he is currently an Engineer with the Department of Artificial Intelligence of CaiNiao Network. His research interest includes flexible automation.

XIAO LUO is currently with the Second Research Institute, Civil Aviation Administration of China and serves as the Director. His research interests include electronic information of civil aviation airport, logistics, and aviation security. He is a Committee Member of the Eleventh All-China Youth Federation and the Standing Director of the China Association for Science and Technology. As a Technical Leader and Manager of aviation, he has won many national honors and published many academic papers.

HUAN XIA received the M.S. degree in software engineering from Yunnan University, Yunnan, China, in 2016. He is currently an Engineer with the Second Research Institute, Civil Aviation Administration of China. His major research interests include machine learning and airport operation.



YUANYUAN ZHANG is currently pursuing the master's degree with the School of Computer Science, Sichuan University, under the guidance of Prof. L. Chen. Her research interests include machine learning and big data.

LIANGYIN CHEN received the Ph.D. degree from the School of Computer Science, Sichuan University, in 2008, where he is currently a Professor. He was a Visiting Researcher with the University of Minnesota under the guidance of Prof. T. He, from 2009 to 2010. He has authored or co-authored more than 40 papers, many of which were published in premier network journals and conferences. His research interests include wireless sensor networks, embedded systems, computer networks, distributed systems, big data analytics, natural language processing, the Internet of Things, and the Industrial Internet.