

An Airport Pavement Traffic Simulation Based on CPN

Regular Paper

Heng Hong-jun^{1,*} and Yang Jue²

¹ Operations Management Team s.r.l, c/o Department of Enterprise Engineering, Roma, Italy

² University of Rome "Tor Vergata" - Department of Enterprise Engineering, Roma, Italy

* Corresponding author E-mail: henghjcauc@163.com

Received 16 April 2013; Accepted 1 September 2013

DOI: 10.5772/56959

© 2014 The Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract According to the characteristics of airport pavement traffic, we discuss a method of building an airport pavement traffic model which is based on CPN theory and simulate a practical situation as well. The method overcomes the shortage of modelling with normal Petri Net theory, solves the difficult problems of airport pavement traffic such as complex traffic nets, frequent road changing, etc., refines the process of the model, and will be good for the model's analysis and simulation.

Keywords Coloured Petri Net (CPN), Airport Pavement Traffic, Simulation

1. Introduction

As part of the air transport and urban infrastructure, an airport is an important part of the comprehensive transportation system. After decades of construction and development, the airport system in China has begun to take shape. With the rapid development of the aviation industry, the airport continues to expand the scale and the throughput capacity of airport is becoming saturated, that have brought more and more problems to the normal

operation of the airport and traffic control. There are many factors in airports that restrict the air traffic flow - for example, runways, taxiways and parking stands, etc. In recent years, the uncertainty in the taxi process has clearly influenced the entry and exit of flights. The research focus has gradually shifted to optimize the airport taxiing path and to improve the efficiency of airport operations[1,2]. In order to understand the airport pavement traffic actions and operating rules correctly and in detail, it is understood as a kind of effective measure that builds the airport pavement traffic simulation system to predict the time when a flight delay occurs as early as possible so as to reduce the influence of flight delays. At present, the study of airport pavement traffic simulation is almost non-existent in China, so building a reasonable airport pavement simulation system is imperative.

A Petri Net (PN, for short) is a three-tuple $\{P, T, \varphi\}$, where P is a finite set of places, T is a finite set of transitions, and φ is a flow function $\varphi: (P \times T) \cup (T \times P) \rightarrow N$. A marking is a mapping $\mu: P \rightarrow N$ (μ assigns tokens to each place of the net). Pictorially, a PN is a directed, bipartite graph consisting of two kinds of nodes: places (represented by circles within which each small black dot denotes a token)

and transitions (represented by bars or boxes), where each arc is either from a place to a transition or vice versa. In addition, each arc is annotated by either $\varphi(p, t)$ or $\varphi(t, p)$, where p and t are the two endpoints of the arc. See Figure 1 for an example in which all the arc labels are one and are, therefore, omitted.

A transition $t \in T$ is enabled at a marking μ iff $\forall p \in P, \varphi(p, t) \leq \mu(p)$. If a transition t is enabled, it may fire by removing $\varphi(p, t)$ tokens from each input place p and putting $\varphi(t, p')$ tokens in each output place p' . We then write $\mu \xrightarrow{t} \mu'$, where $\mu'(p) = \mu(p) - \varphi(p, t) + \varphi(t, p), \forall p \in P$.

Example 1. Figure 1 depicts a PN (P, T, φ) , with $P = \{p_1, p_2, p_3, p_4, p_5\}$ and $T = \{t_1, t_2, t_3, t_4\}$ modelling a simple producer-consumer system. We can view a marking μ as a five-dimensional column vector in which the i th component is $\mu(p_i)$. In Figure 1, transition t_2 is enabled at marking $\mu = (1, 0, 0, 1, 0)$,

since the only input place of t_2 (i.e., p_1) satisfies $\varphi(p_1, t_2) \leq \mu(p_1)$. After firing the transition t_2 , the PN reaches a new marking $\mu' = (0, 1, 1, 1, 0)$, i.e., $(1, 0, 0, 1, 0) \xrightarrow{t_2} (0, 1, 1, 1, 0)$.

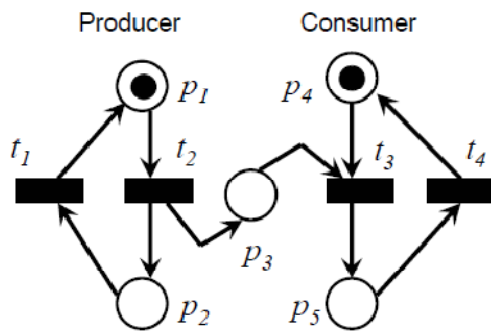


Figure 1. A Petri Net

Unless an execution policy is defined, the execution of a PN is nondeterministic: when multiple transitions are enabled at the same time, any one of them may fire. Since firing is nondeterministic and multiple tokens may be present anywhere in the net (even in the same place), PN are well suited for modelling the concurrent behaviour of distributed systems.

A Coloured Petri Net (CPN) is a backward compatible extension of PN. CPN preserves the useful properties of PN and, at the same time, extends initial formalism so as to allow for the distinction between tokens. CPN is a graphical language for constructing models of concurrent systems and analysing their properties; it is a discrete-event modelling language combining the capabilities of PN with the capabilities of a high-level programming language. PN provide the foundation of the graphical notation and the basic primitives for modelling concurrency, communication and synchronization.

CPN allow tokens to have a data value attached to them. This attached data value is called a 'token colour'. Although the colour can be arbitrarily complex, type places in CPN usually contain tokens of one type. This type is called the 'colour set' of the place.

A CPN is a nine-tuple $\{P, T, A, \Sigma, C, N, E, G, I\}$ where:

- P is a set of places.
- T is a set of transitions.
- A is a set of arcs

In CPN, sets of places, transitions and arcs are a pairwise disjoint $P \cap T = P \cap A = T \cap A = \emptyset$. In addition:

- Σ is a set of colour sets defined within the CPN model. This set contains all possible colours, operations and functions used within CPN.
- C is a colour function. It maps places in P into colours in Σ .
- N is a node function. It maps A into $P \times TU \times P$.
- E is an arc expression function. It maps each arc $a \in A$ into the expression e . The input and output types of the arc expressions must correspond to the type of nodes that the arc is connected to.

The use of a node function and an arc expression function allows multiple arcs to connect the same pair of nodes with different arc expressions. Finally:

- G is a guard function. It maps each transition $t \in T$ into a guard expression g . The output of the guard expression should evaluate to the Boolean values true or false.
- I is an initialization function. It maps each place p into an initialization expression i . The initialization expression must evaluate to a multi-set of tokens with a colour corresponding to the colour of the place $C(p)$.

A CPN is the high-level PN that is obtained by folding places and transitions based on the traditional PN. The value of different tokens in the same place belongs to the same data type. It could be a simple or a complex data type. The data type is the colour set of places, and each place has a colour set. CPN not only defines elements of net structures, such as transition, place and directed arc, etc., but also defines some concepts such as colour, multi-set, arc expression and guard, and so on. Like all net systems, CPN consists of net structures and net expressions. In addition, the net expression can express complex rules.

In recent years, domestic and international scholars have done a lot of research work on solving traffic flow problems with PN theory, and especially in the field of road and railway traffic. They have made some breakthroughs, but no-one has been involved in the field of airport pavement traffic. Now, we here have an airport

pavement traffic model which is based on CPN theory. The model has successfully transferred the airport pavement traffic net into a multi-layer CPN model, and has solved the problems of switching-road rules and conflicts when aircrafts land at the airport.

2. Building the CPN Model

2.1 The characteristics of airport pavement traffic

Compared with ordinary road traffic, the study of airport pavement traffic simulation is a more challenging problem. By analysing the characteristics of airports, there are several differences between airport pavement traffic and normal road traffic, which are also the difficulties of simulation[3]:

- The complexity of airport pavement traffic. Airport pavement roads consist of an apron, a linking way, a taxiway and a runway, and each component has its own function requiring aircraft to obey different rules when running into actual traffic. If the evolution process of each part cannot be separated, it is hard to genuinely embody the airport traffic.
- The bidirectionality of airport pavement roads. When a vehicle travels on a normal road, the direction of travel on a single lane road is unchanging. However, when aircraft travel on airport pavement roads, the taxiing direction of one road is mostly decided by the ATC's instruction. So, it is necessary to avoid taxiing direction conflicts in the process of simulation.
- The dependence of choosing taxipaths. An aircraft's taxi path is mostly controlled by the ATC by means of radio. In order to increase the capacity of airports by shortening the taxi time, ATC always arranges constant and the shortest path to the aircraft. However, due to the speed and spacing of the aircraft is inconsistent, if there is no reasonable control during the aircraft taxiing, the constant taxiing route will cause a rear-end conflict and crossover conflict.
- The frequency of changing lanes when the aircraft taxiing on airport. There are many taxi exits from runway to apron for aircrafts changing lane when aircrafts taxi on the airport pavement (see Figure2). In order to avoid aircraft occupying runway and taxiway for a long time, it is necessary to make aircraft leave runway and enter apron through linking way as quickly as possible.

2.2 Maintaining the integrity of the specifications

CPN is a hierarchy of the high-level PN, it extends the characteristics of ordinary PN. The places and the tokens are labelled colour types to represent different types of resources in the system, at the same time, each place is bound with a specific set of colours, which means the

place can only store the corresponding colour token. Conditional expressions and functions exist in arcs and transitions, which show the values and colour attributes of arcs and the constraint conditions when transitions are triggered. For CPN, when it is executed, the token of every step is represented by an n-dimensional vector as well; nevertheless, each coordinate, as well as meaning the number of tokens in each corresponding place, also means a multi-set expression that not only describes tokens' numbers but also indicates tokens' colour types.

In order to reduce the complexity of building complex PN systems and to put the refining theory to use, we need to build a multi-layer PN. In the PN's description, there are two kinds of transitions: one is a basic transition denoting atomic tasks and the other is a subnet transition denoting complex tasks with an internal structure, an internal action and an internal status. Multi-layer PN modelling could bring such advantages as hiding the internal structure of the subnet, focusing on the appropriate abstract level. For a subnet that has the same structure, there is no need for repeated modelling and analysis. It should make the business process model have a good structure, and facilitate its analysis and processing[4].

Compared to common PN, CPN has two basic and important differences: a) tokens can represent arbitrarily complex data, and the description ability of a directed arc and transition is enhanced, greatly simplifying the complexity of the network; b) CPN has a hierarchical structure, which can page a system from the whole to the local, from the coarse to the delicate, refining gradually, so as to draw out the key points[5].

2.3 The description of the airport pavement traffic CPN model

2.3.1 A brief introduction to airport pavement traffic

Figure 2 gives an airport's pavement traffic map. According to the airport pavement traffic's movement characteristics, an airport pavement can be divided into four parts: an apron, a linking way, a taxiway and a runway. Furthermore, considering the characteristic whereby aircraft may change lanes frequently when travelling on the pavement, we divide the linking way and taxiway into several sections according to the export border.

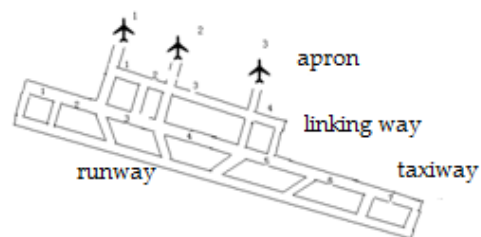


Figure 2. An airport's pavement traffic map, in brief

According to the prescribed route provided by ATC, aircraft taxi on a pavement at a certain speed, which is ordered according to the type of aircraft. The type can be queried by the aircraft's tail number from the database. The colour sets FLIGHT and PATH are defined to represent an aircraft's tail number and taxi path. In the departure process, after ground services on parking stand (such as refuelling, loading, supplying meals, boarding and so on) have been done, the aircraft applies to ATC for a taxi path, and taxis on the ground strictly by a specified path. The arrival process is just the reverse. The aircraft applies for a landing permit to ATC before preparing for landing, and then lands on the runway after this is allowed. When taxied to the end of the runway, the aircraft applies to ATC for a taxi path and taxis on the ground strictly by a specified path.

The departure process of aircraft will be described in detail. As the arrival process of aircraft is roughly the reverse of departure, it will not be repeated here.

2.3.2 A top-layer model

Figure 3 shows how aircraft operate overall on an airport pavement. The places Plane I, Plane II and Plane III represent the flight resources on three parking stands respectively. According to the flight schedule, at the specified time, there will be specified aircraft entering the specified parking stand, while a corresponding place will have an initial token. The subnet transitions Apron I, Apron II and Apron III represent, respectively, operations on the corresponding aircraft at the aprons. The places Apron I Running Standby, Apron II Running Standby and Apron II Running Standby, respectively, represent the operations that aircraft standby before leaving the apron and choosing the taxi target of the next step. The subnet transitions Linking I, Linking II, Linking III and Linking IV represent the process for an aircraft to taxi to the corresponding four sections of the linking way, respectively. The places Linking I Exit, Linking II Exit, Linking III Exit and Linking IV Exit, respectively, represent the operation whereby aircraft arrive at the exit of the linking way and choose the taxi target of the next step after finishing to taxi to the corresponding linking way. The subnet transitions Taxiway I, Taxiway II, Taxiway III, Taxiway IV, Taxiway V, Taxiway VI and Taxiway VII, respectively, represent the process for an aircraft to taxi to the corresponding seven sections of the taxiway. The places Taxiway I Exit, Taxiway II Exit, Taxiway III Exit, Taxiway IV Exit, Taxiway V Exit, Taxiway VI and Taxiway VII Exit, respectively, represent the operation whereby aircraft arrive at the exit of taxiway and choose the taxi target of the next step after finishing to taxi to the corresponding taxiway. The place Runway represents the process whereby aircraft taxi on the runway. The place Fly Off represents not only the

status that an aircraft leaves the runway and begins to fly, but also the identity of any aircraft that has finished its taxi process on the airport pavement. The place Radio represents the ATC's radio resource, and is a shared resource.

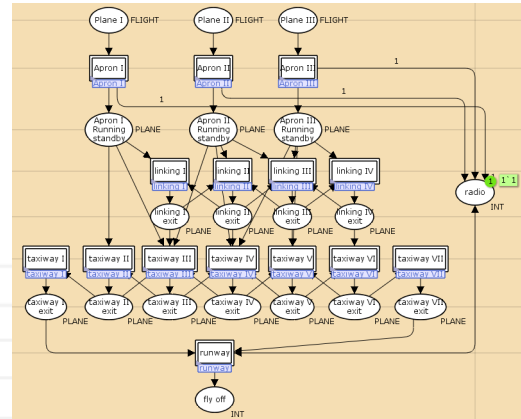


Figure 3. An airport pavement traffic model - top layer

2.3.3 An apron model

In Figure 4, the places Plane I, radio, Apron I running standby whose port type is in or out, correspond to the parts in the top layer (see figure 3). When a flight is ready to depart, the place Plane I will have a token, and the corresponding transition Services is fired up, which means that the flight accepts ground services. When these services are completed, the place Service Standby will have a token, which means that the ground services are finished; meanwhile, when the place radio has a token (i.e., the radio resource is free), the transition Radio Request is activated, which means that the flight is making a radio request to ATC and applying for a dropback operation. After the transition Radio Request is complete, the place Radio will have a token again and release the radio resource; meanwhile, the place Dropback Signal will have a token, and then the transition Dropback will be activated and the aircraft will begin to drop back. After dropping back, the aircraft chooses which stage to taxi to according to the path specified by ATC.

The path specified by ATC is represented by the follow-up node of the current node. Linking I-IV is represented by No. 01-04, taxiway I-VII is represented by No. 41-47, and runway is represented by No. 99. Thus, if the path is 'linking II->linking 1->runway', the path is represented by the number string '010299'.

Two functions are defined in the model: a) fun NEXT (pt : INT, nt : INT) = $((pt \text{ mod } 100) = nt)$, which means that the current node's follow-up node is nt ; b) fun NPATH (pt : INT) = $(pt \text{ div } 100)$, which means the following taxi path information after an aircraft has taxied by a node pt .

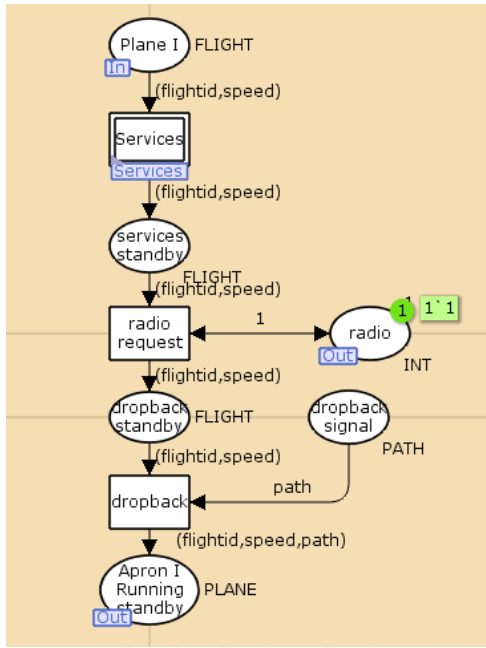


Figure 4. An apron's CPN model

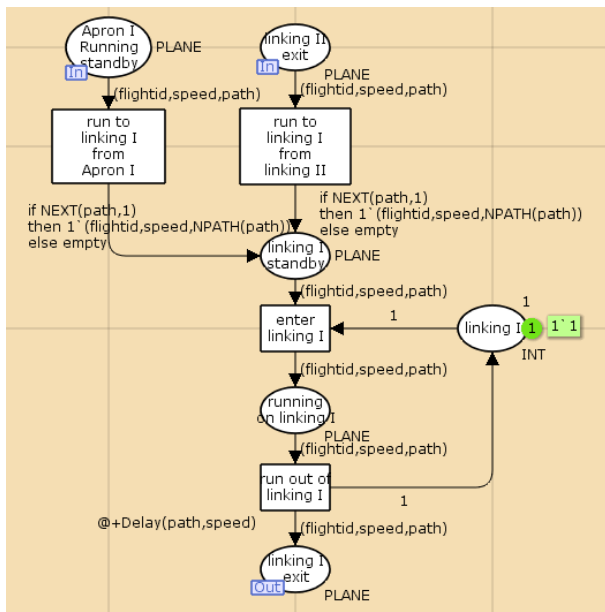


Figure 5. Linking way I's CPN model

2.3.4 A linking way and taxiway model

In Figure 5, the three places whose port type is in or out correspond with the part connecting to the other transitions in the top layer (see figure 3). There are two entrances that enter linking way I - one from apron I and the other from linking way II. After the aircraft is dropped back from linking way I, the place Apron I Running Standby will have a token, and the transition Run to Linking I from Apron I is activated. When the condition NEXT (path, 1) is satisfied (which means that the next path is linking way I), the place Linking I Standby have a token, which means that the aircraft waits at the entrance of linking way I and prepares for taxiing

to linking way I. The process whereby the aircraft taxis from linking way I to linking way II is similar. The place Linking I is a mutex lock, which means that no more than one aircraft is allowed to taxi on linking way I at any time. When the places Linking I Standby and Linking I both have a token, the transition Enter Linking I is activated, which means that the aircraft taxi into linking I, and then the place Running on Linking I have a token, which denotes the status of aircraft taxiing on linking way I. When the transition running on Linking I is activated, the aircraft taxis out of the linking way, and the place Linking I have a token again. The resource linking way I is released, which allows a new aircraft taxi in, and at the same time the place Linking I Exit have a token, which means that the aircraft has already taxied to the exit of linking way I.

The model of other linking ways, the runway and taxiways is broadly similar to that of linking way I, so it will not be repeated here.

3. Simulation

CPN-tools is an editing, simulation and analysis software package exclusively used in CPN, developed by the University of Aarhus, Denmark, and which continues to be improved. The hierarchical modelling tool, the time and colour sets' expression, and the automatic analysis tool provided by CPN-tools together make the users can extract effective data as the basis for performance evaluation from the simulation process of transition and the model state of experience.

We have simulated an airport's pavement traffic using the actual flight data of one airport by CPN-tools. Figure 6 shows the initial state of aircraft waiting on the airport's apron for one day. Figure 7 shows the final state after all planned flights have departed from the airport.

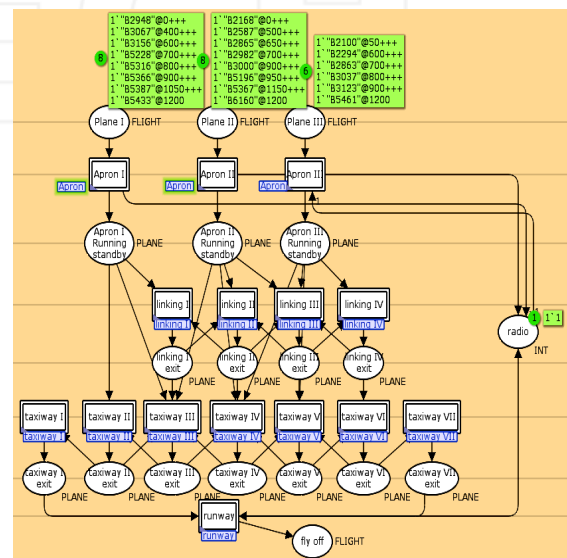


Figure 6. An airport pavement's initial state graph

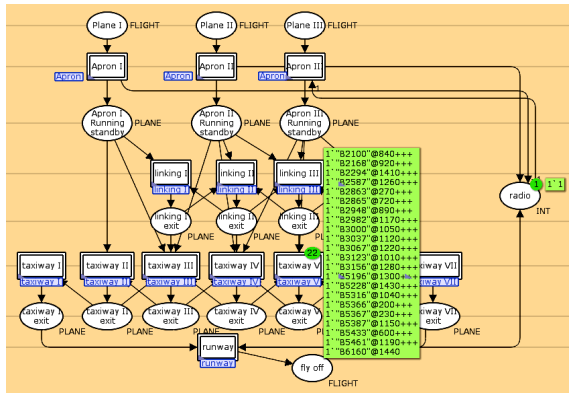


Figure 7. An airport pavement's final state graph

Flight no	Depart time		Flight no	Depart time	
	Sch	Act		Sch	Act
B5366	7:20	7:20	B3037	8:51	8:52
B5367	7:22	7:23	B5387	8:53	8:55
B2863	7:26	7:27	B2982	8:56	8:57
B5433	8:00	8:00	B5461	8:50	8:59
B2865	8:11	8:12	B3067	9:00	9:02
B2100	8:24	8:24	B2587	9:05	9:06
B2948	8:26	8:29	B3156	9:08	9:08
B2168	8:29	8:32	B5196	9:10	9:10
B3123	8:37	8:41	B2294	9:20	9:21
B5316	8:41	8:44	B5228	9:22	9:23
B3000	8:43	8:45	B6160	9:24	9:24

Table 1. Results of simulation

From the process and results of the simulation, the model gives reasonable performance of the transport characteristics of aircraft in the airport. The behaviour of the aircraft (e.g., in taxiing, changing lanes, waiting, etc.) is basically consistent with the actual situation. According to the test data of Table 1, we found that the aircraft taxi time is longer when flights are taking off and that landing is more intensive, while the aircraft average taxi speed is slow. In contrast, when the schedule of flights taking off and landing is sparse, the aircraft taxi time is short and remained stable. From the actual simulation process, the blocking of the runway is more serious, indicating that the runway is the main bottleneck of the airport, while the parking stands also suffer some obstruction, though the obstruction of the taxiway is less. At the same time, we can see that the blocked number of aircraft is proportional to the density of the flight schedule, and when the interval of flights is large, no obstruction will be produced.

4. Conclusion

CPN has a wide range of applications in urban rail transit and highway traffic, but it has never been involved in the field of airport pavement traffic. Through the study of airport pavement traffic's characteristics and the analysis of the difference between normal road traffic and airport

pavement traffic, we developed a CPN model that is suitable for airport pavement traffic and simulated an actual situation. The CPN model reflects well the actual operational situation of the airport pavement, which achieved the model's anticipated effect.

For future work, we need to: a) gradually refine the influence of aircraft types in the model; b) add the factors that may influence aircraft's normal taxiing into the model, considering more adequately the influences of different types of interference factors that affect the accuracy of the flight time and thus enable the model to reflect the actual situation on the airport pavement more realistically; c) compare the results of the simulation with the state of actual airport pavement traffic in more depth, and facilitate managers in analysing the capacity of the airport so as to adjust the forecasts of airport traffic flow and, thus, improve the reasonableness of any scheduled flight plan.

5. Acknowledgments

The authors acknowledge the support by the United Funds of the Civil Aviation Administration of China and the National Natural Science Foundation (NSFC) under Grant No. U1333109.

6. References

- [1] China Investment Consulting Net, "Airport industry faces unprecedented opportunities to develop," *New Chinese Businessman*, 2009-01-06, <http://finance.sina.com.cn/chanjing/b/20090116/18225771129.shtml>.
- [2] WANG Zhi-qing, ZHONG Shan, ZHAO Chuang and LIU Xue-ni, "Study on economic and social benefit evaluation of Chinese airports," *China Civil Aviation*, vol. 107, Nov. 2009, pp. 43-45, doi: cnki: sun: mhjj.0.2009-11-016.
- [3] WANG Bin, ZHANG Yun and WANG Xiao-hong, "Research on the CPN-Based Hierarchical Workflow Process Modelling," *Computer Engineering & Science*, vol. 30, Oct. 2008, pp. 139-142, doi: cnki: sun: jsjk.0.2008-10-049.
- [4] Editorial Board, "Civil Aviation Airport Ground Services Practical Guide," China Knowledge Press, 2006, pp. 313-571.
- [5] ZHAO Qiang, ZHOU Lin, CHEN Wei and QI Li-hui, "Modeling of Repair of Surface to air missile Equipment Based on Hierarchy Coloured Petri Net," *Aeronautical Computing Technique*, vol. 38, Nov. 2008, pp. 28-31, doi: cnki: sun: hkjj.0.2008-06-009.
- [6] ZHU Lian-zhang, LIU Fan, "Modeling publish/subscribe systems based on colored Petri nets," *Computer Engineering and Design*, vol. 30, Feb. 2009, pp. 845-848, doi: cnki: sun: sjsj.0.2009-04-020.

- [7] The CPN Group, University of Aarhus, Denmark, "Getting started with CPN Tools," 2010-04-10, [http://www.daimi.au.dk/~wells/cpntools/HelpTranslation/getting_star ted_with_cpn_.html](http://www.daimi.au.dk/~wells/cpntools/HelpTranslation/getting_started_with_cpn.html).
- [8] Claude GIRAULT, Rüdiger VALK, "Petri Nets for Systems Engineering," Germany: Springer, 2003, pp. 71-226.

INTECH

INTECH

