

Research Article

Capacity Analysis for Parallel Runway through Agent-Based Simulation

Yang Peng,¹ Gao Wei,² and Sun Jun-Qing¹

¹ School of Computer Science and Technology, Tianjin University of Technology, Tianjin 300191, China

² Civil Aviation University of China, Tianjin 300300, China

Correspondence should be addressed to Yang Peng; pqxx@sina.com

Received 28 March 2013; Revised 18 June 2013; Accepted 18 June 2013

Academic Editor: Ming Cao

Copyright © 2013 Yang Peng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Parallel runway is the mainstream structure of China hub airport, runway is often the bottleneck of an airport, and the evaluation of its capacity is of great importance to airport management. This study outlines a model, multiagent architecture, implementation approach, and software prototype of a simulation system for evaluating runway capacity. Agent Unified Modeling Language (AUML) is applied to illustrate the inbound and departing procedure of planes and design the agent-based model. The model is evaluated experimentally, and the quality is studied in comparison with models, created by SIMMOD and Arena. The results seem to be highly efficient, so the method can be applied to parallel runway capacity evaluation and the model propose favorable flexibility and extensibility.

1. Introduction

Airports play a key role in the commercial aviation system by allowing airlines and their customers to converge. However, since the early 1970s, the peaking of traffic at airports has been a problem of increasing concern to airport operators around the world. This challenge is most pronounced in the runways, where multiple traffic flows converge within a confined region. A failure to manage these runways effectively can start a causal chain wherein an accumulating bottleneck effect leads to fighting delays, unused runway capacity, and increased controller workload which, in turn, leads to increased financial and environmental costs.

An airport's capacity may be broadly defined as its ability to handle a given volume. While a runway system capacity is defined as the hourly rate of aircraft operations which may be reasonably expected to be accommodated by a single or a combination of runways under given local conditions [1]. Congestion occurs when demand approaches or exceeds capacity. Nonintersecting runways, whose extended centre lines have an angle of convergence/divergence of 15 degrees

or less, are called near parallel runways. The use of parallel runways to increase aerodrome capacity is a common concept at busy aerodromes.

Results in this study are mainly focused on parallel runways, conducted using an agent-based simulation software called AnyLogic. Given its capabilities for modeling at a very high level of detail and closely representing reality in terms of applicable separation standards and air traffic management procedures. Agent based model is created to simulate proposed alternative. Using the methodology proposed here, the baseline and the different alternatives were evaluated in terms of design functionality and overall utilization of potential capacity.

2. Literature Review

2.1. Capacity Estimation Models. The primary analytical models used to estimate runway capacity include the LMI runway capacity model and the FAA airfield capacity model [2]. The LMI capacity model is an analytical model for computing the capacity of a runway system. Its fundamental

building block is a model that computes the capacity of a single runway. The FAA airfield capacity model is an analytic model which calculates the capacity of a runway system, but it assumes that all random variables in the model are normally distributed and a 5% probability of violation of separation requirements is used in determining spacing of runway operations, using these normal distributions. A hybrid of these two models, with the logic of the LMI model and the extension to multiple runways featured in the FAA model, is expected to be very useful in providing quick estimates of runway system capacity [3].

A distinction between analytical and simulation models is made based on the methodology used to compute capacity, delay, or other such metrics. Analytical models are primarily sought to provide estimates of capacity by manipulation of the representation formulated. These models tend to have a low level of detail [4].

Simulation of the airport environment is increasingly applied to obtain more realistic estimates of capacity by randomizing the various input parameters. Many commercial simulation tools for air traffic are available, for example, SIMMOD and TAAM. These tools dedicated to airport types of simulation seek to generate traffic flows through the airports, which are modeled and configured to represent actual constraints and uncertainties. Observations from these flows allow appropriate measures of capacity and/or delay to be computed. Simulations tend to have a much higher level of detail including conflict resolution, airport taxiway, and gate selection, to deal with more tactical issues.

In many applications these commercial simulation tools will satisfy the simulation objectives, but when microsimulation becomes an issue to consider, very few models analyze air traffic metrics in combination with capacity and safety. With regard to microsimulation, we are concerned with events such as wake-vortex separation violations, simultaneous runway occupancies (SRO's), and collisions on a runway. Furthermore, the aviation system is a complicated stochastic system.

2.2. Agent-Based Systems for Air-Traffic Control and Management. Agent-based computing is one of the powerful technologies for the development of distributed complex systems [5, 6]. Many researchers believe that agents represent the most important new paradigm for software development since object-oriented design [7], and the concept of intelligent agents has already found a diverse range of applications in transportation systems.

The geographical and functional distribution and the highly dynamic nature of air traffic control (ATC) make it an ideal candidate with many potential applications that can be modeled with MAS [8], such as air traffic flow management [9]. The optimal aircraft sequencing using intelligent scheduling (OASIS) presented in [10] is a real-time agent-oriented system developed to support air traffic management. OASIS distributes air traffic control (ATC) tasks into two classes of autonomous and cooperating agents: aircraft agents and global agents. Each aircraft agent associates with an arriving aircraft and performs computation or reasoning relevant to

the aircraft. The system helps alleviate air traffic congestion by maximizing runway utilization through arranging landing aircrafts into an optimal order and monitoring the progress of each individual aircraft in real time.

An agent-based model enables modeling of every individual throughout the complete simulation lifecycle in complex scenarios, so, to follow these domestic properties, an agent-based simulation is created to provide the necessary information for parallel runways.

3. Operational Conceptions and Considerations

The impetus for considering operations on parallel or near-parallel instrument runways is provided by the need to increase capacity at busy aerodromes. The procedures for airports with multiple parallel runways are similar, with added safeguards to ensure that an aircraft is safely separated from those approaching the adjacent parallel runway. The relation among many parallel runways can be turned to the relation between two parallel runways. So we only focus our research on two parallel runways.

In order to enable evaluation of capacity of parallel runway, it is necessary to model the details of how arriving aircraft is guided through the final stage, including their behavior and interactions. Modeling of the agents was accomplished by comparison with a set of baseline operational data and consultation with subject matter experts.

3.1. Modes of Operation

3.1.1. Simultaneous Parallel Approaches. Independent parallel approaches: simultaneous approaches to parallel or near-parallel instrument runways where radar separation minima between aircraft on adjacent extended runway centre lines are not prescribed. Dependent parallel approaches: simultaneous approaches to parallel or near-parallel instrument runways where radar separation minima between aircraft on adjacent extended runway centre lines are prescribed.

3.1.2. Simultaneous Parallel Departures/Departures. Independent parallel departures: simultaneous departures from parallel or near-parallel instrument runways. Segregated parallel departures: simultaneous operations on parallel or near-parallel instrument runways. Under this mode, one runway is used exclusively for approaches and the other runway is used exclusively for departures. There may be semimixed operations; that is, one runway is used exclusively for departures, while the other runway is used for a mixture of approaches and departures; or, one runway is used exclusively for approaches while the other is used for a mixture of approaches and departures.

3.2. Safeguards for Parallel Runway. In order to obtain reliable and valid model to evaluate the capacity, it was necessary to collect these regulations in a systematic and controlled manner. Therefore, various aircraft controls are described in this section.

3.2.1. Separation Control. The separations imposed between aircraft by terminal controllers can be characterized by the interarrival separation distribution of aircraft as they fly along the final approach course. This distribution is a good measure of the applied separations since it is sensitive not only to the average separations applied, but also to the variability of the applied separations around the average. The variability of aircraft separations can have many sources, including the different required minimums between different aircraft types and the deviation of aircraft from their assigned routes and speeds. All of these sources are captured by the interarrival separation distribution along the final approach. The final approach is particularly important because it is where aircraft are moving most slowly, are most closely spaced, and have their routes most constrained.

To avoid the situation where a faster aircraft catches up with a slower aircraft, an extraseparation is required for the following faster aircraft at the beginning of its final approach. FAA has an official separation requirement for each aircraft mix, and the separations are chosen to avoid a wake vortex encounter. The target separation at the threshold is drawn from a Gaussian distribution $\sim N(\mu, \sigma^2)$, where the mean μ is given in Tables 1 and 2 based on the leader and trailer aircraft types, and the standard deviation σ is assumed to be 0.39 nm. The actual separation at the threshold depends on several factors including aircraft speed and time of arrival at the final approach fix.

3.2.2. Runway Entrance Control. The minimal time between an aircraft that wants to takeoff (takeoff lasts about 30 seconds) and an aircraft that wants to land is about 3 minutes. If the time difference is smaller, aircraft at the ground have to hold until the runway is vacated.

3.2.3. Runway Balance Control. The model flow manager chooses a landing runway for each inbound aircraft based on the need to balance throughput on the available runways. Balancing the throughput on all the runways of an airport is important because overwhelming one runway while another remains underutilized will lower the overall throughput of the airport. In order to choose which runway would be best, based on balancing considerations, the model flow manager first calculates an estimated demand on each runway and then decides whether reassigning the aircraft to other than the nearest runway would be desirable. The estimated demand is calculated as a weighted moving average constructed so that nearer flights contribute more to the estimated demand than distant flights.

Once estimated demands have been calculated for each available runway, the flow manager will reassign the incoming flight to the runway with the lowest estimated demand if the demand estimate on that runway is lower than the demand estimate on the nearest runway by at least a specified amount called the demand barrier. The demand barrier used in the simulations is 2 flights per hour. The purpose of the demand barrier is to ensure that flights are not reassigned to other than their closest runway unless there is a clear advantage in doing so.

TABLE 1: Separation standard matrix (nautical mile).

Leader	Trailer		
	Heavy	Large	Small
Heavy	4	5	6
Large	2.5	2.5	4
Small	2.5	2.5	2.5

TABLE 2: Hypothetical reduced separation (nautical mile).

Leader	Trailer		
	Heavy	Large	Small
Heavy	3.3	3.3	3.3
Large	2.5	2.5	4
Small	2.5	2.5	2.5

4. Architecture of MAS

4.1. Agent-Oriented Parallel Runway Architecture. Agent-based modeling develops computational representation of a complex system by modeling each of the components or subsystems as agents and it models the rules for possible actions and interaction between these agents.

To formulate the evaluation model, we created the following pseudoagent. The developed model considers each subsystem as an independent type of agent that acts based on its local knowledge and its interaction with other agents.

- (1) Runway agent is applied to simulate the runway and it was afforded statistical methods to recorder the utilization rate of the runway.
- (2) Runway entrance agent is applied to simulate the entrance of the runway and it was embedded in the runway agent.
- (3) Departure agent is applied to simulate the departure aircraft. Different aircraft types are modeled through database of aircraft performance parameters.
- (4) Departure environment agent is applied to simulate the surrounding condition of the departure aircraft. The departure aircraft can learn the surrounding information from this agent, and it is embedded in the departure agent.
- (5) Arrival agent is applied to simulate the departure aircraft.
- (6) Arrival environment agent is applied to simulate the surrounding condition of the arrival aircraft.
- (7) Departure track agent is applied to simulate the approach line of aircraft, it can record the quantity of arrival aircraft at any time.
- (8) Arrival track agent is applied to simulate the approach line of aircraft.
- (9) Rule agent receives and converts the regulation data, for example, the miss distance between aircraft.
- (10) Control agent balances the arrival and departure aircraft.

Considering that UML is a language widely accepted, we will design the architecture of the agent-oriented parallel runway system at this part by AUML. The AUML is extended from UML with agent-related modeling techniques, to enhance the modeling capabilities of UML [11]. Figure 1 shows the group membership hierarchy on which the design of the runway system is based. To describe the simulation model in more detail, some entities are further decomposed to virtual agents. For example, the arrival aircraft is decomposed to arrival agent and the corresponding arrival environment agent; in this way the model can be applied to more complicated condition without much change, as is shown in Figure 2. The main features of the important agents presented in this architecture are specified in the following sections.

4.2. Agent Description

4.2.1. Arrival Agent. An agent will be created for each arrival aircraft. Therefore, in response to the arrival of an aircraft (arrival agent creation event) the system will create a new arrival agent instance for this aircraft. Each arrival agent faces a safe problem. In this problem, a strict set of rules must be followed, and arrival agent may get a copy of the rules from rule agent through communication and can get the position and characteristics information of nearby-arrival agents through its arrival environment agent. Also, the agent has to coordinate with the other service agents in order to resolve any conflicts. This arrival agent must have the following information: the ID of assigned runway to the aircraft, the aircraft characteristics, the operation time assigned to the different operations (approach/landing).

4.2.2. Runway Agent. Each aircraft will be assigned to a specific runway. Each one of the runway is controlled by a runway agent. To do this, the agent will have to manage the runway assigned and the corresponding runway entrance agents, informing the control agent and the corresponding aircraft agent when necessary.

When an arrival agent approaches to the target runway, it will have to communicate with runway agent to check the state of the runway. If the runway is not locked by departure agent and no aircraft will take the runway upon its arrival to the runway, the aircraft will continue its approach processes; otherwise it will have to hover at its current position.

4.2.3. Control Agent. The main goal of this agent is to determine the appropriate allocation for the arrival/departure aircrafts to specific time and runway. While an arrival agent is created, it has to wait in the approach queue for the instruction of the control agent before it moves to the target runway, so do departure aircraft.

The control agent will try to obtain the most appropriate scheduling to balance the arrival and departure stream. At this point, the agent will search the information of aircrafts at approach/departure queues. The agent will have to know the following information in order to perform its task: the

length of the approach/departure queues and the cumulative waiting time and max waiting time of aircrafts of the approach/departure queues.

4.3. Agent Communication. One of the most important tasks is the coordination and the negotiation between the agents. Each one is in charge of one independent part of the system. The message interchange between agents forming the system is not based on the most popular foundation for intelligent physical agents specifications (FIPA-ACL). Instead, we apply the regular interobject communication facilities of the simulation environment for agents: calling methods, sending messages via ports, linking continuously changing variables, and so forth.

The interaction between agents, while arrival aircraft approaches to the runway, is shown in Figure 3.

While arrival aircraft is created, it gets target runway and obstacle aircraft information from control agent. The aircraft registers its information to the arrival track agent and approaches to the runway, as soon as it gets permission from the control agent. During the approach process, the aircraft checks whether the distance to obstacle aircraft satisfies the safety rules. If the distance is too short, the aircraft will hold on current position, until it is safe for it to move forward again. Upon the final approach stage, the aircraft communicates with the runway agent and departure aircrafts to make sure that the runway will be available when it reaches the runway, and no departure aircraft has locked the runway.

5. Developing Models in AnyLogic

5.1. Simulation Control and Data Collection. The simulation approach used in this paper is based on the software tool AnyLogic. AnyLogic is a programming and simulation environment based on Java that is matching with unified modeling language (UML) inherently and the model based on AnyLogic possesses the open architecture as [12] characterizes. AnyLogic focuses on agent-based approach and business simulation and allows the user to combine different techniques and approaches.

The example problem is considered in this case study to conduct simulation although the model being developed is independent of the problem structure. The example is drawn partially with minor modifications from Bei Jing airport runway structure, the structure information of the parallel runway is stored in the database, and the distance between runways can be manually changed during simulation. Figure 4 shows the parallel runway structure.

Validation of the computer simulation is a critical step since the reliability of predictions derived from it depends on the fidelity of the simulation to actual operations. It is a good idea to compare the simulation results of the model with real airport operation data, but the data is hard to collect. So, we verify the multiagent model through comparison with SIMMOD model and Arena model. The screenshot for multiagent-based parallel runway simulation model is presented in Figure 5. Several versions of the model have been

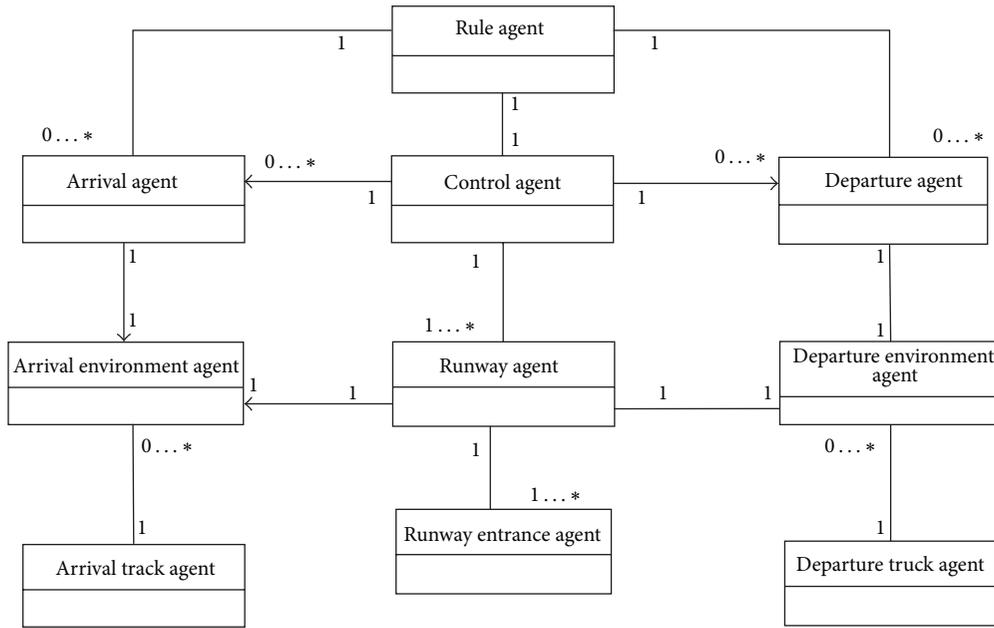


FIGURE 1: Agent Class Diagram.

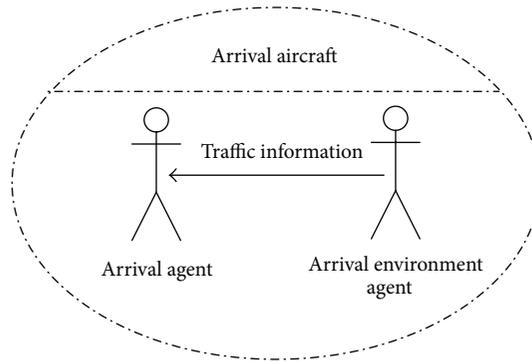


FIGURE 2: Collaboration of Virtual Agents.

simulated allowing a comparison of runway capacity under different operation modes.

The experimenter’s display is essentially an electronic map of an area. All aircraft are shown as symbols at their proper locations. In a separate window on the screen, a variety of information about the subject aircraft is displayed. This information is received from the model and is shown as a reference for the experimenter. Information about pseudoaircraft can be displayed, including the waypoints the pseudoaircraft is flying through.

5.2. Numerical Experiments. Small aircrafts are seldom used in recent years at Chinese aviation market, so we suppose that only large and heavy aircraft were available at the simulation experiments. Some of the important parameters of the aircraft are displayed at Tables 3 and 4.

Being one of the oldest and most commonly used air traffic simulation modeling tools, SIMMOD outputs were used as the baseline for expected results. The average delay

TABLE 3: Scenario data.

	Aircraft types	
	Medium	Heavy
Arrival velocity km/h	314	334
Take-off velocity km/h	309	315
Mixindex%	63.4	36.6

of aircrafts is used to measure the quality of a scenario after execution and to compare with other scenarios.

One of the main benefits of a simulation model is the wealth of information derived from each simulation run. Each scenario is repeated for 20 times, and the averaged simulation results are then presented to illustrate the use of the model developed.

5.3. Results and Analysis. In the case of independent arrival, the distance between parallel runways is over 1035 m, and the

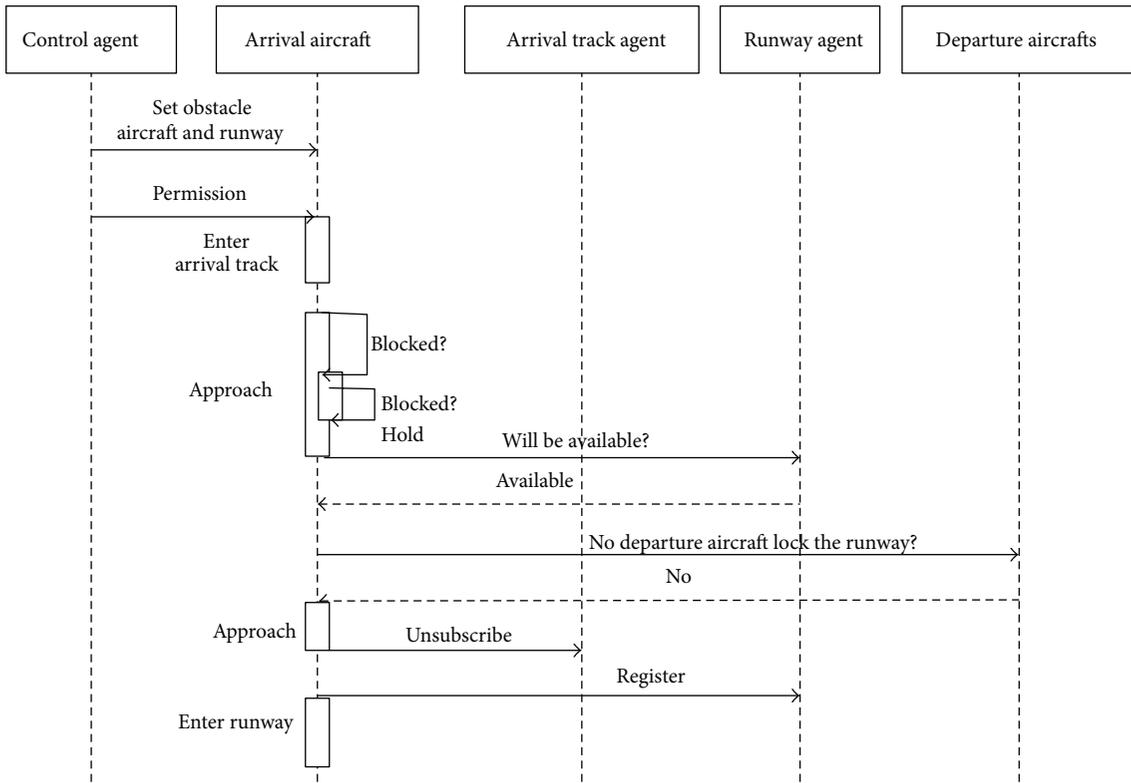


FIGURE 3: Message sequence diagram.

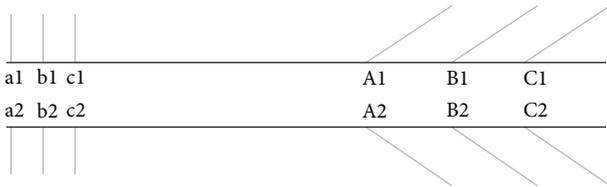


FIGURE 4: Runway Structure.

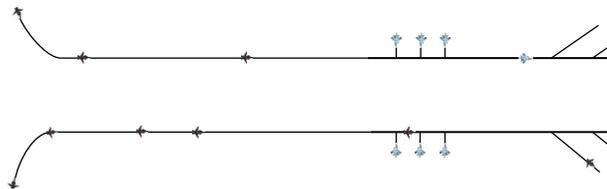


FIGURE 5: Screenshot of the model.

initial arrival rate is 10 aircrafts per hour. Then the arrival rate will gradually rise to an unrealistically high level. As expected, the level resulted in all flights being delay. The resulting delay distribution is depicted in Figure 6.

As can be seen form Figure 6, the results of the models are quite similar when the density of arrival aircraft is small. While the arrival rate turns to unsustainable heavy, the distinction turns to be clear. This fact ensures that a simulation validated against airport arrival operations will have been stress tested in high-density traffic situations. The

TABLE 4: Taxiway choice parameters.

First runway	Ratio of medium aircraft that taxi to A1, B1, and C1	0.4, 0.3, 0.3
	Ratio of medium aircraft that taxi to A1, B1, and C1	0.2, 0.3, 0.5
Second runway	Ratio of heavy aircraft that taxi to A1, B1, and C1	0.3, 0.4, 0.3
	Ratio of heavy aircraft that taxi to A1, B1, and C1	0.2, 0.2, 0.6

results of multiagent model are in accordance with those of SIMMOD, but the results of the Arena and SIMMOD are significantly different.

Suppose that the tolerable level of average delay is 4 minutes, and the corresponding capacity is regarded as runway practical capacity. Figure 7 shows the simulation results of dependant approach scenarios.

Clearly there are differences between the results of Arena and SIMMOD models for this scenario, and, by inspection of the confidence intervals, it is clear that the results of agent and SIMMOD are similar. The most important reason for the difference lies in the technique used to implement origin-destination routing and separation control. While agent-based model uses a monitoring technology, checking the distance between nearby aircraft at any time. Arena uses a trigger technology, checking the distance at particular moment or

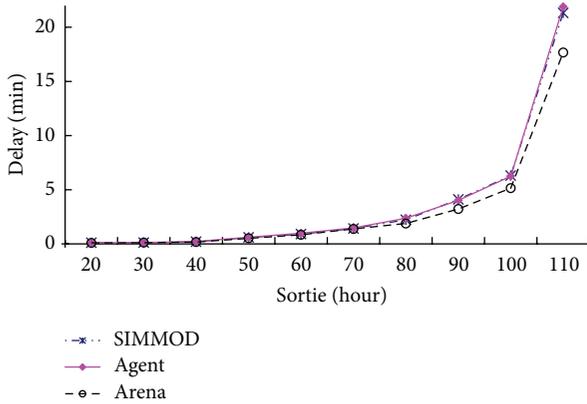


FIGURE 6: Comparison of delays under independent parallel approaches.

particular position. Consequently, Arena required each intersection to have hard-coded turning movement percentages. The reason for not implementing routing and separation mechanism in Arena to match that in Agent model is the complexity of the cooperative logic and component in a microscopic level system. From this, it is very difficult to determine specific vehicle turning directions at each intersection, which results in shorter separation for aircrafts. Due to the nature of the routing technique utilized in Arena, there was no way to assign specific destinations for aircrafts each origin. Because of this, there was no guarantee that destination volumes in Arena would match those established separation rules.

The purpose of introducing agents into parallel runway simulation is to increase the flexibility and the ability of the system to deal with uncertainty in a dynamic environment. Through the above comparison, we can see that the multiagent model consists of multiple functional stationary agents that are intelligent and cooperative. Interoperability is critically needed in making decisions based on information across systems, organizational and jurisdictional boundaries, or application scenarios in which the integration of multiple agent systems is needed.

6. Conclusions

This research applied agent-based modeling approach to the simulation of parallel runway system. The simulation model that seeks aircrafts and runway interactions using individual agents that attempt to fulfill a specific objective is developed. The coordination between agents is achieved through protocols. The hierarchical architecture makes it possible to examine and understand how approach mode and architecture decisions might affect the parallel runway capacity. Regarding the proposed model, the movement of aircrafts at taxiway should be also included at the multiagent model in future research.

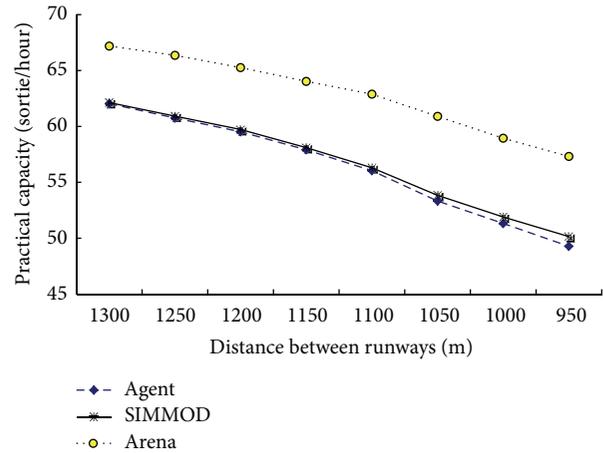


FIGURE 7: Comparison of capacities under dependent parallel approaches.

Conflict of Interests

Yang Peng, Gao Wei, and Sun Jun-Qing declare that they have no proprietary, financial, professional, or other personal interests of any nature or kind in any product, service, and/or company that could be construed as influencing the position presented in, or the review of, the paper.

Acknowledgment

This work is financially supported by National Nature Science Foundation of China under the Grant 61201234, Tianjin City, Board of Education Scientific Research Plan Project under the Grant 20110416, and the Basic Research Universities Special Fund Operations ZXH2010D011, ZXH2012G002.

References

- [1] B. Jiang, M. Hu, and Y. Tian, "Further research of airport runway capacity evaluation," *Journal of Traffic and Transportation Engineering*, vol. 2, no. 3, pp. 80–883, 2003.
- [2] H.-Q. Guo and J.-F. Zhu, "Calculation models of capacity and delay for γ closely spaced parallel runway," *Journal of Traffic and Transportation Engineering*, vol. 18, no. 14, pp. 68–72, 2008.
- [3] W. Wang, M. Wang, and X. Qian, "Study on method of determination of centrally distance and threshold staggering manner for closely spaced parallel runways of airport," *Journal of Civil Aviation University of China*, vol. 29, no. 2, pp. 23–26, 2011.
- [4] S. Alam, H. A. Abbass, and M. Barlow, "ATOMS: air traffic operations and management simulator," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 2, pp. 209–225, 2008.
- [5] F. Zambonelli and H. van Dyke Parunak, "Signs of a revolution in computer science and software engineering," in *Proceedings of the 3rd International Conference on Engineering Societies in the Agents World III (ESAW '02)*, pp. 13–28, Springer, Berlin, Germany, 2003.

- [6] S. Wollkind, J. Valasek, and T. R. Ioerger, "Automated conflict resolution for air traffic management using cooperative multi-agent negotiation," in *Proceedings of the American Institute of Aeronautics and Astronautics Conference on Guidance, Navigation and Control (AIAA '04)*, pp. 1078–1088, August 2004.
- [7] M. Luck, "Guest editorial: challenges for agent-based computing," *Autonomous Agents and Multi-Agent Systems*, vol. 9, no. 3, pp. 199–201, 2004.
- [8] B. Burmeister, A. Haddadi, and G. Matylis, "Application of multi-agent systems in traffic and transportation," *IEE Proceedings*, vol. 144, no. 1, pp. 51–60, 1997.
- [9] K. Tumer and A. Agogino, "Distributed agent-based air traffic flow management," in *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '07)*, pp. 342–349, May 2008.
- [10] M. Ljungberg and A. Lucas, "The OASIS air traffic management system," in *Proceedings of the 2nd Pacific Rim International Conference on Artificial Intelligence*, Seoul, South Korea, 1992.
- [11] N. Mani, V. Garousi, and B. H. Far, "A UML-based conversion tool for monitoring and testing multi-agent systems," in *Proceedings of the 20th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '08)*, pp. 212–219, Dayton, Ohio, USA, November 2008.
- [12] L. Bin, L. Wen-feng, and Z. Yu, "Study on modeling of container terminal logistics system using agent-based computing and knowledge Discovery," in *Proceedings of International Symposium on Advances in Computer and Sensor Networks and Systems*, pp. 164–171, Aardvark Global Publishing, Sandy, Utah, USA, 2008.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

