SYM-OP-IS 2009
36th Symposium on Operational Research
Mathematical Institute SANU - Belgrade

BUILDING THE TRUST IN THE MODEL: RAMS PLUS CASE

BOJANA MIRKOVIC
Faculty of Transport and Traffic Engineering, Belgrade, b.mirkovic@sf.bg.ac.rs

Abstract: *The aim of the study presented in the paper was to test one of commercially available simulation tools for airfield modeling, in order to build the trust in the mode before it is used for supporting the decision making. Airfield module of the RAMS Plus simulation tool is tested on Munich airport example. Desirable enhancements of the model are suggested.*

Keywords: *Gate-to-gate fast-time simulation, Airfield modeling, RAMS Plus*

1. INTRODUCTION

Continuous air traffic growth, scarcity of the capacity, tendency for more efficient usage of available resources, is some expressions which in the best manner describe air transportation nowadays. Last 10-15 years airports are identified as air transportation most serious bottleneck. Globally observed there is enough capacity, but the problem is in mismatching of offered capacity with demand. There are airfields with barely any traffic, runways serving several aircraft per week, while on the other hand major airports become more and more congested, operating at the ultimate capacity limit.

In such circumstances, simulations specialized for air traffic analyses have become a very powerful tool for supporting the planning process. The greatest value is recognized for the so called "gate-to-gate" fast time simulations which offer possibility of simulating the traffic from departure gate to arrival gate. Such models, in addition to en-route, also treat terminal airspace and ground movements at airports of origin and destination.

In order to base the decisions on output from simulation, one has to build the trust in the model, first. Only if we believe that the model mimics the real system good enough, we can rely on output we receive from it. Testing the model is the best way by building the trust in it. In this paper the testing of the groundside module of RAMS plus software on Munich Airport example is described. The reason we chose RAMS plus is that we already were one of the academia users of the airspace module, so the airfield (groundside) module was available to us, as soon as it was released.

2. RAMS PLUS

RAMS plus is a gate-to-gate ATC/ATM fast-time simulator, which helps answering a spectrum of questions about the ATM system, from airspace design, capacity, working procedures and safety concerns, to airport movements, capacity and delay. ISA software is officially in charge for developing, support, licensing and distributing of this software.

By releasing so called groundside module, at the end of 2003, RAMS plus is upgraded to "gate-to-gate" model (version 5.0 and later). Earlier, it was used only for airspace analysis and the airports were defined only by the coordinates of their reference points.

Groundside module of the RAMS Plus is tested on Munich airport (MUC) example. In the reminder of the paper is explained how to model an airfield in RAMS Plus with special emphasis on what should be improved. Also, some of the simulation results are presented and weak points of the tool which gathers and analyses the results from the simulation (so called ATM Analyzer) are commented. We used the version 5.19 which was the latest version at the time the simulations is done.

Case study – Munich Airport

Supply side in the MUC model is based on MUC layout depicted on Figure 1: two parallel independent runways, one apron, and one terminal building (as it was the case before 2003). Current state, with new Terminal 2 and Aprons 2/3 would not be suitable for the initial testing of the model, since they have special policy of usage; they are in use only by Lufthansa and Star Alliance members.

For the demand side we used available data at that moment, which were the 5-peak hour traffic from 1998, summer timetable, day Thursday.

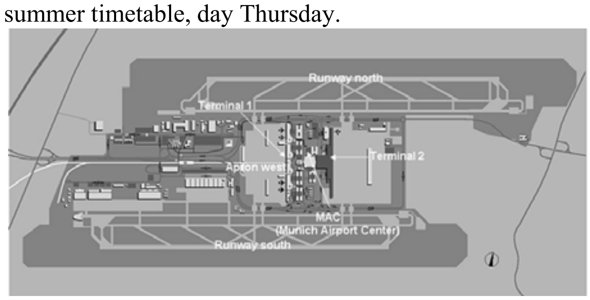

Figure 1: Munich Airport layout

A. Runways

An airport model in RAMS Plus is a network model. Each element of the airport structure consists of nodes defined by their latitude and longitude, and links which are a connection between certain pairs of nodes. The runway is defined by the nodes representing ends of runways but as well the nodes which are intersections between runways and runway exits. The directions connecting these nodes define the direction of takeoff and landing. To enable operations in both directions an option to creating a reverse direction runway has to be applied. In these models we fixed directions for use for each runway. MUC has two independent parallel runways (4000m long, 60m wide), with direction 80/260. For the Munich Airport model it was assumed that only direction 08 is in use (RWY 08L and 08R).

Except length, width and direction amongst the physical characteristics of runway are *touchdown marker* and *flight strip width*. There is a possibility to define touchdown variance, but in MUC model it was assumed that distance between threshold and touchdown is a constant value (by default - 948ft).

Lock time is the period during which one operation blocks all other operations from using the runway. For each runway two lock times have to be assigned: lock time before landing and lock time after take-off. We adopted recommended values: 60s after take off and 120s before landing. In the case of runway intersection, which is not the case at MUC, RAMS is offering possibility to define *blocking time*. Each runway can optionally block other runways during landing and take off events. During the blocking time, flights can not use the blocked runway.

Runway occupancy during take off and landing can be calculated in three ways, each of them has different priority. The highest priority is calculating occupancy using acceleration and deceleration rates. All aircraft models are divided in 61 groups in RAMS. For each group of aircraft certain performance characteristics are given, among which are acceleration and deceleration rates. If these values are set as 0, for that group of aircraft we can define a mathematical distribution for runway occupancy during takeoff/landing. If we do not define the distributions for the group, occupancy is calculated using a lowest priority criterion - mathematical distribution for the occupancy during take off and landing defined for each runway, regardless of aircraft model using the runway. Only 5 of 61 groups did not have values for acceleration/deceleration rates but for these groups no aircraft models were assigned. It means that a first criterion was always used.

For each runway a time it has to be defined when a flight requests a *runway reservation*, and in this case it is by default 1800 s before landing. The departure runway is scheduled just before the departure runway's first node. During conflict resolution, a flight may be sent to a terminal holdstack, or a runway departure queue. *Departure Queue* is a node that represents counter of the aircraft in the queue which also measures the time each aircraft has spent in the queue. If this type of data is required such node should be located by each runway threshold. Terminal *holdstacks* together with arrival and departure routes has to be assigned for each runway, which will be explained in more details later.

B. Taxiway system

To create a taxiway system in RAMS Plus it was necessary to determine geographical location for each taxiway structure node and, then, to connect these nodes in an appropriate way. For each link the user must define these parameters: direction (uni-directional A-B, bi directional or reverse-directional B-A), passing (no passing or bi-directional passing), link type (arrival and departure, only arrival, only departure), types of aircraft and airlines allowed using the link, maximal taxi sped on the link, maximal number of aircraft on the link at the same moment. In MUC model all links are bi-directional (except bridges in the Munich Airport model, which are uni-directional), all links are no passing (one plane must wait if the other is already on the link), all links are used both by arrivals and departures, assumed values for maximal speed on the links are: apron-15kts, other taxiways- 40kts, high-speed exits- 60kts, maximal number of aircraft on the link depends on its length and it is defined for each link separately, and there were no restrictions by airlines and type of aircraft using the links.

The appropriate link can be defined as a *high-speed exit* in the case that one of its nodes is included in the runway structure, in the same time. If an aircraft manages to decrease its speed enough, at the moment it reaches a common node (for the RWY and high speed exit) then it is allowed to use that exit.

There is a possibility to define *taxipaths*, if they are required. Tahipath is simply the list of the links which are usually in use on the way from one parking position to the runway and reverse. One flight can use the taxipath only if it is assigned to that flight. An alternative aircraft is using the *shortest path* on its way from the runway to a gate and reverse. RAMS is calculating the shortest path using a shortest cost algorithm. The shortest path has priority to the taxipath, which makes the task more difficult if there is a need to simulate an airport where taxipaths are habitually in use. It would be necessary to assign an adequate taxipath to each flight.

There is a recommendation for the taxiway node structure to be connected to the start and to the end node of the runway. If it is not done like that, some of the flights can act quite unusual, performing something that looks like a missed approach, but it is certainly not (because a missed approach is not implemented in version 5.19 as part of simulation). This also means that aircraft can take off only from the start of the runway. In real conditions at Munich Airport smaller aircraft are not taking off from the start of the runway, but they use one of the exits to get out onto the runway and take off from there. So, in version 5.19 it was not possible to simulate that.

C. Aprons

A parking stand is symbolically described with a link that connects the apron taxiway with a particular gate. Each gate is defined by its geographic coordinates. Apron 1 at MUC has 60 parking stands. It is possible to allow gate usage only for some aircraft models, aircraft groups or airlines. These types of restrictions were not used in MUC model, because the available traffic data gate for each flight is already known. In the case that gates are not given in advance then RAMS is *allocating gates*, by searching alphabetically for the first available gate that is assigned to that particular airline and is able to accept the flight aircraft model. There are, also, two alternative criteria for gate allocation- searching first for the available gate closest to the departure runway, or closest to the arrival runway. It is obvious that in the case where RAMS Plus is doing gate allocation these kinds of restrictions are desirable because they can help in making the model closer to the real system.

If it is not necessary to define each gate for the purpose of the simulation, then gates can be grouped using a *supergate option*. Supergate can accept more than one aircraft in the same moment. The user defines the number of aircraft for one supergate.

The turnaround process at gates is modeled through three activities in RAMS: departure boarding, arrival unloading and arrival turn-around time. These activities are not defined for each gate or for each apron, but as a characteristic of the airport. In the Munich airport model these activities are described using recommended Gamma distributions ((420,10) for departure boarding and (1200,10) for arrival unloading and arrival turn-around time). Once these parameters are set, it is possible to measure gate occupancy/delay. RAMS does not calculate total gate occupancy but differs departure and arrival gate occupancy. Departure occupancy is equal to departure boarding time and arrival occupancy is unloading time plus turn-around time. The same is applied when gate delay is in question.

D. Terminal Airspace

Terminal airspace is defined by navigation aids, approach and departure routs (STARS and SIDS) and terminal holdstacks. Each navigation aid is defined with latitude, longitude and type (VOR/DME, VOR, DME or FIX). There is a possibility to define navigation aid as fly-by, which is used to simulate turns as arcs. Approach and departure routes are defined on navigation aids, containing altitude and speed restrictions and separations. There are some additional restrictions for using SID and STAR like by airlines, by performance group, by navigation aids. Holdstack can only be defined on a navigation aid. There are two types of holdstacks in RAMS Plus software: terminal and en-route. Only terminal holdstacks are defined in MUC model.

E. Simulation results for given traffic sample

Two scenarios were developed as part of the experiment: the Basic scenario and Scenario 1. In both of them, the same supply side is used, but two different traffic samples were simulated. For the Basic Scenario we used available traffic data, that is one day traffic from 1998, from the summer timetable (day Thursday). The simulation was done for the 5 peak hours with total number of operations is 162, equally distributed by runways. In the Scenario 1 the traffic is increased by 10% (the total number of operations – 178). Distribution of operations by runways is shown in Table 1 for both scenarios. For increasing the traffic the cloning option is used. RAMS clones randomly picked flights by varying only the system entry time. All other parameters assigned to cloned flights are the same as for the original flight (flight number, gate, etc). In the case RAMS does the gate allocation, there is no problem with this kind of cloning. But, if the gates are already allocated to each flight, assigning the same gate to the cloned flight is likely to result with gate delay.

Table 1: Traffic distribution by runways

	RWY north		RWY south	
	Basic sc.	**Sc. 1**	**Basic sc.**	**Sc.1**
departures	40	40	40	46
arrivals	42	46	40	46

All the data from the simulation are collected by the tool named ATM Analyzer. From the collected data ATM Analyzer creates some reports by default, but also offers possibility to modify the reports and do additional analysis. During our study we experienced a lot of problems with ATM Analyzer for the ground side.

Simulation results in both scenarios are based on four iterations (base iteration and three additional, in which the system entry time was varied from uniform distribution in interval ±5 minutes). Counter to what is planned ATM analyzer does not record output data for each iteration, but only the first (base) iteration. So, each iteration had to be simulated individually which automatically means that all the reports are useless since they are based only on one iteration and user has to do the post processing on its own. This is very big disadvantage, because if we have to put so much effort in building the model, than we expect at least quick output analysis.

Performance indicators measured in MUC case are ground delay and departure queue. Only some basic charts will be illustrated without going in any deeper analysis with other available data.

In more than 90% of flights arrival delay was less than 15min, which is not considered as delay, so arrival delay was not further analyzed. Ground departure delay was much more significant, in average 33,7min (in average for 10 aircraft on runway north and 7 aircraft on runway south recorded delay was more than 45min). Main components of ground departure delay are: gate delay during passenger boarding, taxi delay and departure queue delay. The share of each component in the total ground departure delay is shown in the Figure 2. As expected ground departure queue delay has the greatest share in total ground departure delay. But, what is not expected, gate delay is always present, obviously with similar absolute values, since its share is decreasing with total delay increase. Even bigger confusion came from the finding that 90% of the departure flights left the gate exactly at the planned time, but they had a certain gate delay recorded. After deeper analysis it is discovered that this resulted from the fact that all activities at gates are described with Gamma mathematical distribution with activated option that forces the generated value to be equal or greater than the mean value. Since the gate delay is calculated as the difference between the generated and mean values is clear (but not justified) why it always exists, even when aircraft leave the gate at a planned time.

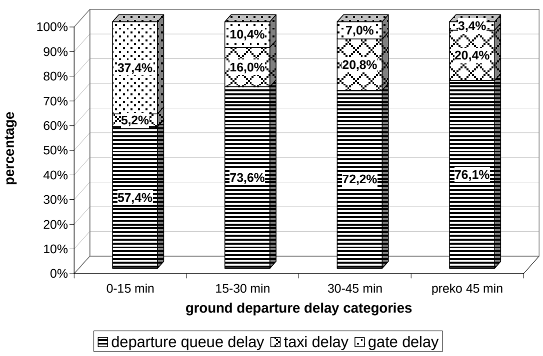

Figure 2: Share of components in ground departure delay

Problems appear with analyzing departure queue as well. As the matter of fact ATM Analyzer does not calculate departure queue at all. If interested in this performance indicator user has to extract it from the times each plane enters and leaves the departure queue. Figure 3 depicts maximal departure queue recorded in a 15- minute time intervals for both runways. On the runway south the maximal departure queue is 10 aircraft, on the runway north was 13 aircraft.

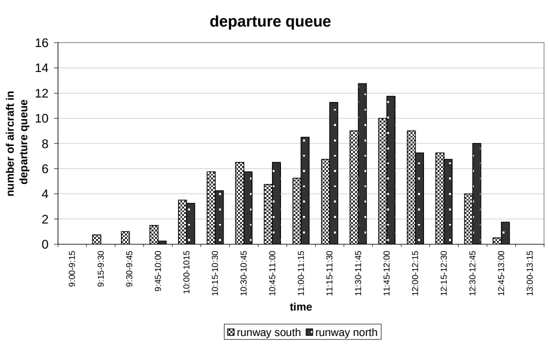

Figure 3: Maximal departure queue in 15-minute time intervals, both runways, average of four iterations

The way RAMS responds to traffic increase can be seen from the comparison of departure queue length in Basic and Scenario 1, Figures 4 and 5. In the case of runway south it can be seen that the departure queue is constantly longer in Scenario 1, as expected. The maximal number of aircraft in the queue is 15 which is 5 more than in the Basic Scenario. In the case of the runway north, something unexpected happened - the queue in Scenario 1 is smaller than in the basic scenario. Considering that the number of departures is the same in both scenarios, this is in fact some kind of basic scenario variation, with 4 additional landings (see Table 1), which caused "translation" of the queue.

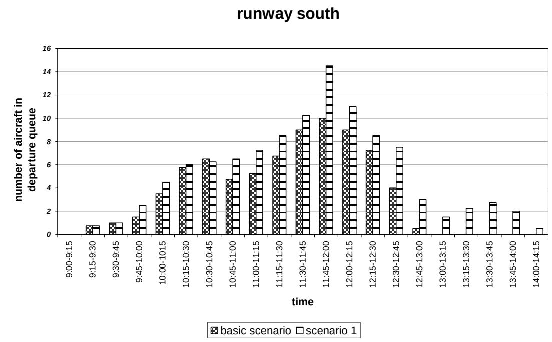

Figure 4: Maximal departure queue on southern runway in 15-minute time intervals, average of four iterations (comparison)

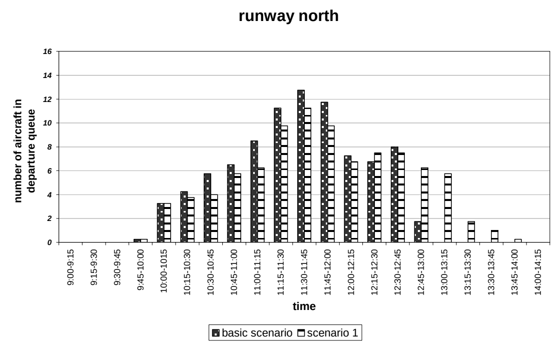

Figure 5: Maximal departure queue on northern runway in 15-minute time intervals, average of four iterations (comparison)

Based on all abovementioned it cannot be stated that RAMS mimics the real system satisfactorily. Another confirmation of this statement is that our attempt to simulate traffic increase by 20% failed, while in real life Munich Airport handled such traffic in 2001 successfully.

3. CONCLUSION

The final conclusion from the testing the groundside module in RAMS Plus is that (at least the first versions) are not mimicking the real system good enough that we can trust the model and output it is delivering. Our findings from the testing together with recommendations for possible enhancements are presented to ISA Software at one of the RAMS user's meetings. Some of the problems we pointed on are improved in the later versions of the software, most of them related to ATM Analyzer.

REFERENCES

[1] B. Mirković, *Traffic Simulation at Munich Airport*, Undergraduate thesis, Faculty of Transport and Traffic Engineering, Belgrade, 2005.

[2] RAMS Plus User Manual, Version 5.19, 2004

[3] RAMS Plus Data Manual, Version 5.19, 2004

[4] AIP (Aeronautical Information Publication), Germany, 1998