MACROSCOPIC MODELLING OF PASSENGER STREAMS ON THE AIRPORT AND ITS ADAPTATION IN MATLAB SIMULINK

Yousef Farschtschi[1], Dominik Formella[1], Kai Himstedt[1], Jochen Wittmann[2], Dietmar P.F. Möller[1]

[1]University of Hamburg, Faculty of Informatics,
Vogt-Kölln-Str. 30, 22527 Hamburg, Germany
[2]University of Applied Sciences, Faculty of Ecological Informatics,
Wilhelminenhofstr. 75A, 12459 Berlin, Germany

farschtschi@informatik.uni-hamburg.de (Yousef Farschtschi)

Abstract

We present an approach on macroscopic modelling of passenger streams on airports. In today's logistic areas like harbours or airports the efficiency has to be increased constantly by for example reducing the costs. The efficiency of all stations from the arrival to the boarding of passengers in an airport is very important. The airports and accordingly the airlines have to hold their labour costs low and handle the stream of passengers. Therefore the operator of an airport like the one in Hamburg has to search for useful methods. This paper will show the concept of a macroscopic passenger model and the resulting possibilities will support the efficiency of an airport. The concept will be introduced in UML in order to offer a clear model and will be adopted in a simulation programme to show the bottlenecks in the workflow. The adaptation will be made in the widespread simulation programme MATLAB Simulink. The goals of this contribution are the macroscopic model and its adaptation in UML and MATLAB Simulink. The main goals will be to increase the efficiency by offering the airport operator or airlines the possibility to receive a forecast of their workflow by using different input data for the simulation. The result will be to identify the bottlenecks in the workflows.

Keywords: Model, Airport, Bottleneck, Passenger, Simulation.

Presenting Author's biography

Yousef Farschtschi. He studied computer science at the University of Hamburg and achieved his diploma in 2009 [1] in the field of airport optimization and workflow optimization. On the basis of his experience in the airport area, he has made more research activities in this context (see e.g. [2, 3]). Now he is working in the University of Hamburg in the Faculty of Informatics as an employee and Ph.D. student.

1 Introduction

The constant increment of the passenger volume on airports has to be handled. The airport Hamburg for example has a continuous increment of passengers over the last four years. In the year 2006 there were about 11,954,560 passengers and 2008 about 12,840,000 passengers. The only exception is the year 2009 as a result of the worldwide economic crisis with 12,229,131 passengers [4]. These streams of passengers have to be handled. The hiring of more personnel is associated with more costs. The airport operators and the airlines want to know where they require more personnel and where they require fewer personnel in their workflow.

The concept of the passenger model will be described and presented by using a UML (Unified Modelling Language) [5, 6, 7] activity chart. This will be adapted to MATLAB Simulink [8, 9] and allows the user like an airport operator or an airline to find their bottleneck in their workflow. Past experiences have shown that simulations deliver good result at low costs.

2 Passenger model

In order to minimize expenses to the first prototypical implementation the passenger model is split up in two parts. One relates to the departure and the other one to the arrival of the passengers. In this case the passenger model disregards passengers with a connection flight for the arrival and departure. A frequently used term is bottleneck and therefore it will be explained in the next subsection

2.1 Bottleneck

The term bottleneck is responsible for the limitation and capacity of a system or a subsystem. The following frequently used example will give a good description. If water is pouring out of a bottle, the rate of outflow is limited by the width of the conduit of exit. A possible solution for increasing the rate of which the water flows out is to increase the width of the bottleneck.

In the context of the airport the number and capacity of counters, luggage car and so on are responsible for bottlenecks. They arise from waiting queues.

2.2 Departure workflow of the passenger model

The passenger model for the departure is based on the following description of the workflow.

After a passenger reaches the airport by car, taxi, bus or subway the check-in has to be done. There are different possibilities for the check-in. The opportunities are check-in counter, check-in machine and online check-in. We will start here with the check-in counter. If many passengers arrive at the same time or problems with the check-in appear for example some passengers have excess luggage, the check-in system fails or something else happens a bottleneck may arise. This applies to all check-in possibilities besides the online check-in without luggage, because here the passenger can directly go to the security check. Another check-in possibility is to check-in with a machine. The passenger has the possibility to check-in with his/her passport, surname, entry code and credit card or the airline account card. The last possibility is the online check-in. This method is not supported by all airlines. The passengers are able to print their boarding card by using the internet. Only for the check-in machine and online check-in, special luggage counter can be used.

By using the first-in-first-out principle the passengers get served. Among some formalities the passengers check-in their luggage. Every airline has its own regulations in terms of weight and size of the luggage. This also applies for the cabin luggage. After checking in the luggage the passenger receives the boarding pass and the receipt for the luggage.

The next station in the workflow is the same for all check-in possibilities. The luggage gets in the cargo compartment of the aeroplane by the ground personnel.

After receiving the board card the passengers have to go through the security check. Because of the same reasons as before a bottleneck might arise here. For non Schengen-countries [10] the passengers have to go through the passport control. The Schengen area currently consists of 25 European countries and is a borderless zone. A bottleneck might arise here, too.

The last station of this workflow is the boarding. From the security area the passengers can reach their gate. The flight crew has to check the boarding card and the identity card or accordingly the passport of the passengers again. A bottleneck is here possible, too.

2.3 Arrival workflow of the passenger model

After the aeroplane reaches the gate position the passengers leave by using the aerobridge or busses on the apron. This depends on the airline, because a terminal park position is more expensive than an apron park position. If the gate has to be reached by bus, a bottleneck might arise.

If the passengers checked-in luggage they can receive it on the luggage carousel. Here a bottleneck might arise. The luggage of passengers of the first and business class has a higher priority and will be available faster. Passengers from non Schengen-countries have to go through the passport control and some passengers possibly have to go through the custom control. Here bottlenecks might arise, too.

After leaving the terminal the workflow is finished.

3 Grading of the intended model

The described model from section 2 now has to be defined by the grade of detail. There are different factors. The factor time encompasses 24 hours and will show the passenger handling for one day at the airport. In order to give the user the opportunity to demonstrate one day of their workflow this factor was set. The factor space is in this model the airport Hamburg. The passenger model encompasses the terminal and possibly the apron during the transit by bus. It is assumed, that this model could be adapted to other airports. Depending on the airport few changes have to be made for the adaptation. The resolution deals with the individual passenger and it is caused by the goal of the macroscopic model.

The target of this contribution is to create a macroscopic passenger model and to visualise it with UML. This can be implemented in simulation programmes to reach the main target to increase the efficiency by finding the bottlenecks. These factors define the level of the intended model and will be used in the following implementation.

4 Transfer in UML activity chart

In this section the passenger model will be presented by using a UML (Unified Modelling Language) [5, 6, 7] activity chart. This implementation is based on the work of Frahm [11]. UML is one of the most popular modelling languages. It combines techniques from the business modelling like workflows, data modelling like entity relationship models, component modelling and object modelling. In the meantime UML is a de facto industry standard.

First the specification will be presented and afterward the chart will follow.

4.1 Specifications

This modelling language has special notation elements which can be seen in Figure 1. There are several other elements which are not used here.

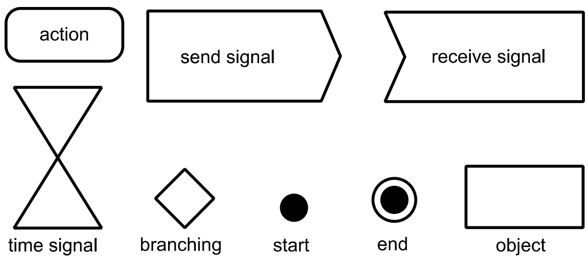

Fig. 1 Notation elements

The UML activity chart completes a series of activities. To show the interdependency with other processes the send signal and receive signal was brought in. If a control flow reaches a receive signal station the workflow will be continued as soon as a signal was received. The time signal stops the workflow and waits a specified time. Actions describe the current state or an action. An object contains at least one object and can be applied to an action. The branching element contains at least two branches. Every branch might have an edge width. Every model or sub model needs a start and end element.

4.2 UML activity chart of the passenger model

These models are determined by the description of section 2. The target here is to visualize the procedure of an airport in a clear UML model.

The different possibilities for the check-in are shown in figure 2, 3 and 6.

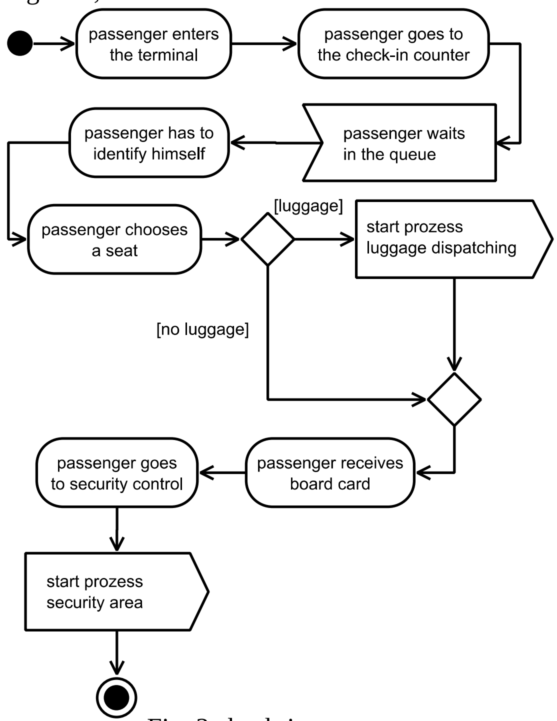

Fig. 2 check-in counter

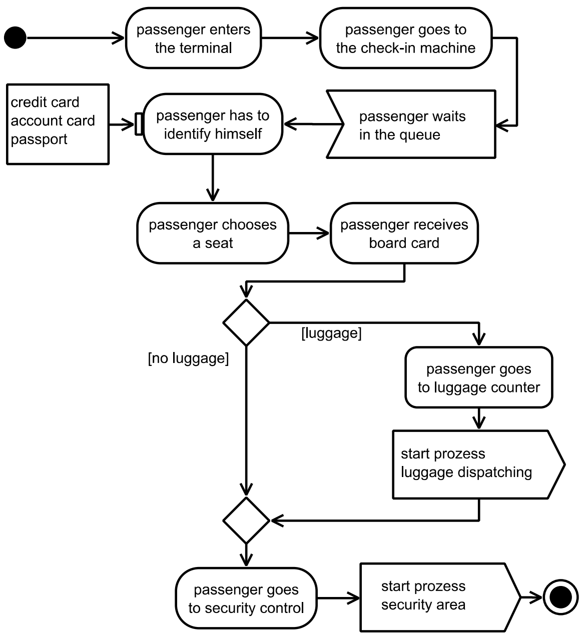

Fig. 3 check-in machine

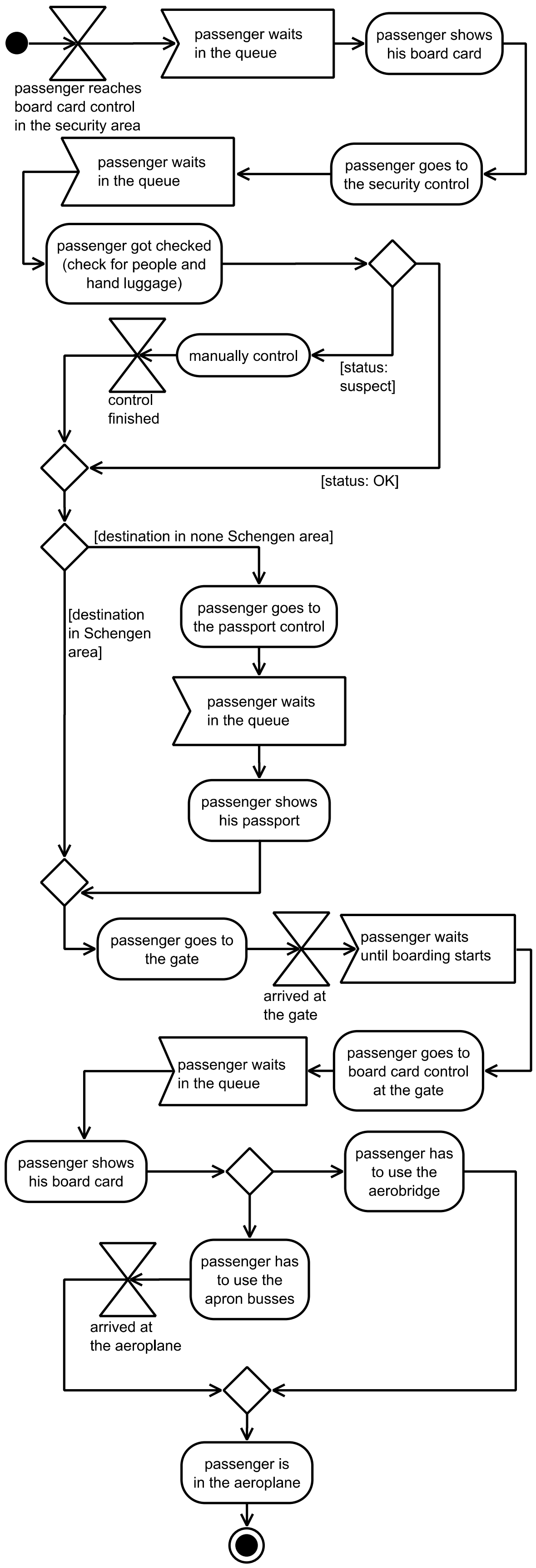

Fig. 4 Passenger dispatching in the security area

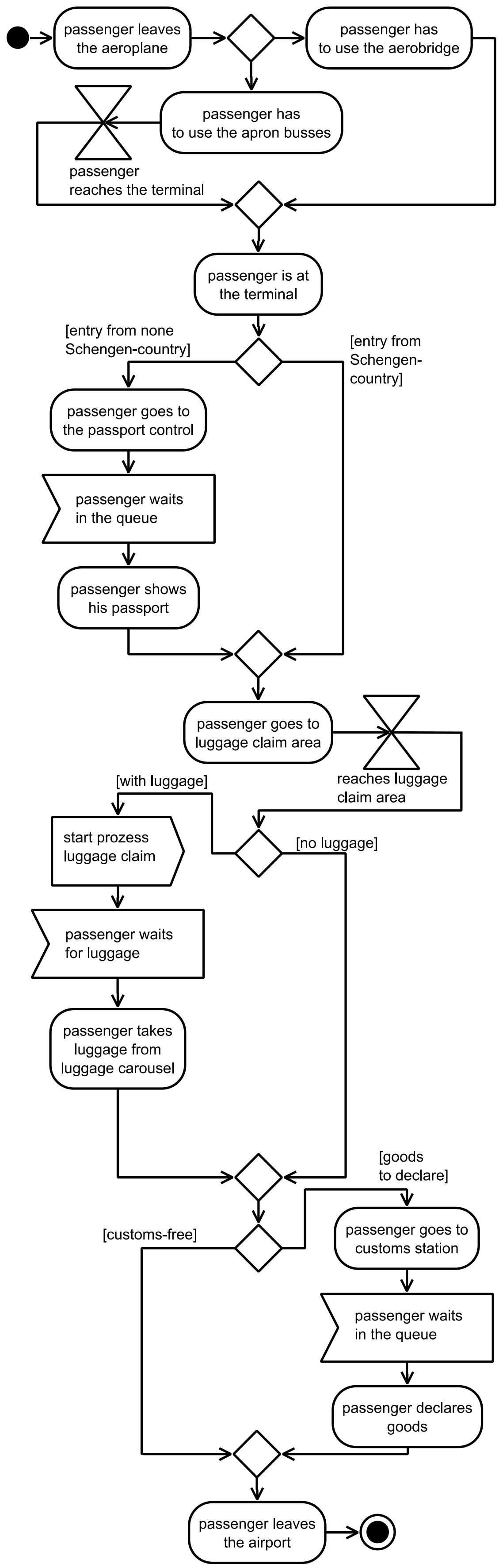

Fig. 5 Passenger arrival

Figure 4 shows the security area after the “start process security area” in figure 2, 3 and 6 has been triggered.

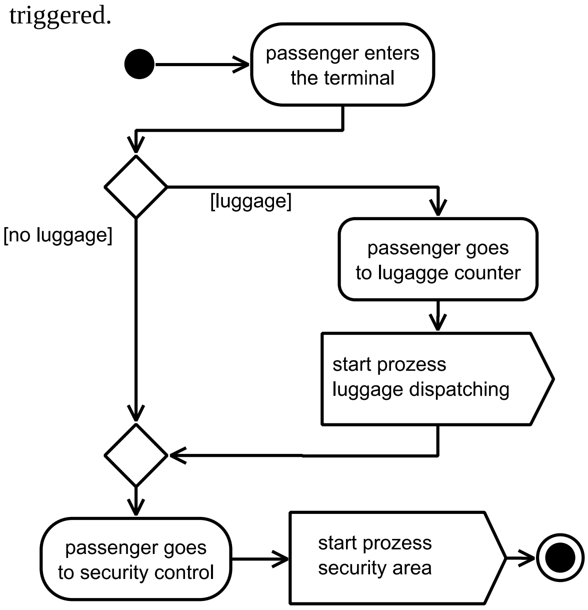

Fig. 6 online check-in

Figure 4 shows the dispatching of the passengers after the “start process security area” in Figure 2, 3 or 6 has been triggered. Figure 5 shows the arrival of passengers on the airport.

For more details on the starting process “luggage dispatching” in figure 2, 3 and 6 see [12].

5 Implementation in MATLAB Simulink

The described and visualised model from section 2 and 4 now will be adapted to MATLAB Simulink [9]. This model was formerly adapted in ModelMaker 4 [13] by Dreyer [14]. Because of its good framework and interface to programme languages like C++ and Java, the new adaptation in MATLAB Simulink was made. MATLAB Simulink is a popular simulation programme developed by MathWorks [8]. It is a commercial tool for modelling, simulating and analyzing dynamic systems. Simulink offers an integration with the MATLAB environment and either drive or be scripted from it. MATLAB for its part is numerical computing environment and fourth-generation programming language.

The main target of the adaptation is the discovery of the bottlenecks to help the user to increase their efficiency like costs and time. This main target will be shown in the following subsections.

5.1 Adaptation of a single station

Now we present exemplarily a single station from section 2 and 4 to proof the plausibility of this station until the whole model gets adapted. This adaptation can be seen in figure 7. It contains the security check from the section 2 and 4.

“Passenger flow” block in figure 7 of the type “from workspace” receives the number of arriving passengers from the MATLAB workspace. These numbers of passengers can be passed in a table. MATLAB Simulink uses scope blocks for the graphical output in x/y diagrams. They are represented by the interfered rectangles. “Security control” block, “flow” block and “security flow” block are scopes and give a graphical representation of the result. The last block “passengers enter the gate” is an “integrator”; all arriving passengers will be added up over time. The combination of “add” block and “integrator” block implements the flow between the blocks “passenger flow” and “security check”. In this example the “security check” block has 16 counters and each of them has a capacity of 60 passengers. Therefore from a queue length 960 passengers a bottleneck arises and the passengers will not reach their flights. The integrator block first saves the value of the passenger flow and due the limit of 960 passenger the rest will be subtracted by the add block. This rest now will be added to next passenger input in the add block. This cycle repeat itself until no rest remains.

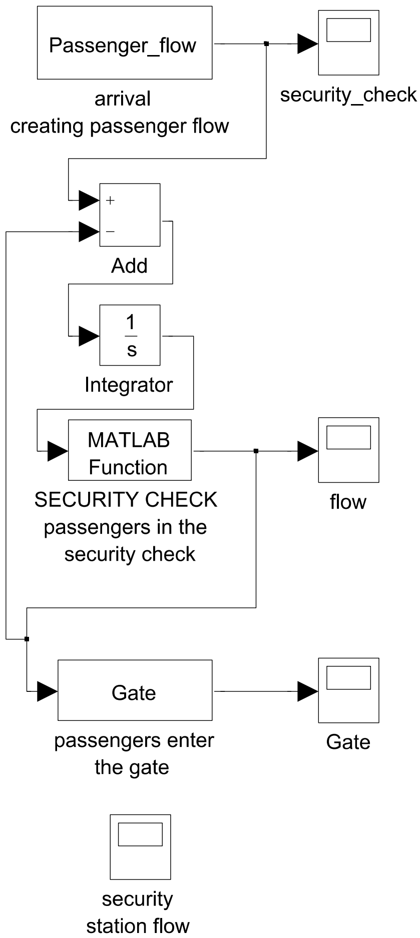

Fig. 7 Security check station

The arriving passenger will be passed to the simulation in a table which shows for each hour of a day the number of arriving passengers. The scopes deliver the result. The output of the scope “security station flow” in figure 7 and can be seen in figure 8. This diagram shows the arising bottleneck from a number 960. Because of the assumption that most of the passengers arrive in the morning and in the evening the output of figure 8 can be seen as a realistic result and the validation of the plausibility.

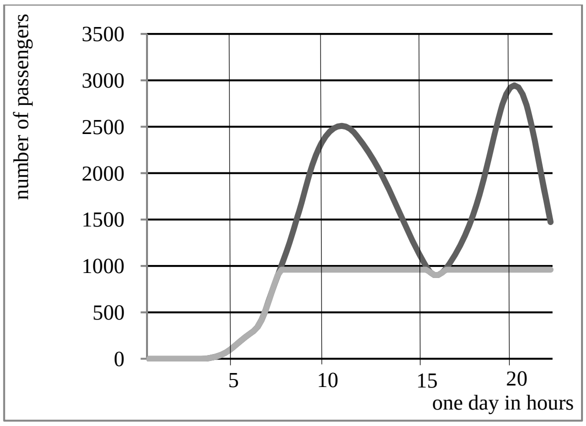

Fig. 8 Passengers in security check station

5.2 Adaptation of the whole model

In this subsection the adaptation of the whole model will be made in MATLAB Simulink. Figure 9 shows the passenger model for the departure. The procedure of this simulation builds on the description of section 2 and the UML model of section 4. After the validation of the plausibility of a single station was realised the adaption of the whole model can be made.

To increase the clarity the hide mechanism of MATLAB Simulink was used. Otherwise the simulation would be confusing from a degree of complexity. The example of figure 7 contains an add-block followed by an integrator-block. This can be prevented by using subsystems. All stations in capital letters contain a subsystem and the structure is like the one in figure 7. The arrival-block is responsible for generating the input of the arriving passengers. The airport Hamburg has about 12.2 Million passengers in the year 2009. Because of strikes, adverse weather conditions like natural disasters or something else the assumption was made that there are only 350 flight days per year. This results about 34,857 passengers a day and dividing this for departure and arrival about 17,428 passengers remain.

Table 1: Distribution of arriving passengers

time	passengers
6 - 12	7500
12 - 16	2428
16 - 20	7500

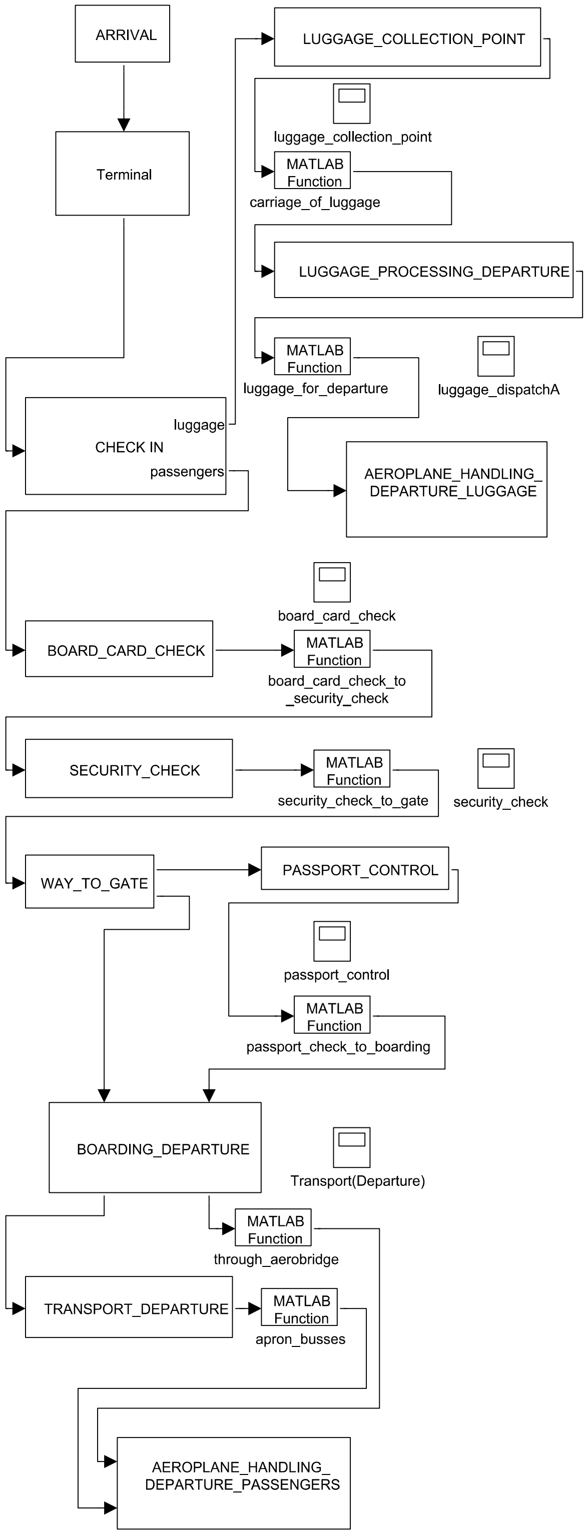

Fig. 9 Departure

Based on the number of parking slots of 11,558 [15] and the construction of the new subway line in the airport the following assumption were made: 1/3 of the passengers arrive by the subway and bus and 2/3 arrive by car and taxi. It is further assumed that most of the departures take place in the morning and evening. Table 1 shows the distribution on this assumption. The time space encompasses 24 hours. The input of the simulation is the result of table 1 and the different arrival possibilities. MATLAB works internal with matrices and vectors and therefore the input can be made in tabular form. This table is filled with arriving passengers divided in columns for each arrival possibility and rows for each hour. After the arrival these passenger will spread over the different check-in possibilities by table 2. Table 2 shows the distribution of the check-in possibilities. This is a plausible assumption on the basis of the experiences of one of the authors.

Table 2: Distribution of check-in possibilities

kind of check-in	passengers as a percent
online check-in with luggage	10
online check-in without luggage	30
check-in machine with luggage	10
check-in machine without luggage	10
check-in counter with luggage	30
check-in counter without luggage	10

Figure 10 shows the subsystem check-in of figure 9. Because of the clarity of the simulation this subsystem was made. The incoming passengers spread over the different check-in possibilities by using table 2. This table was left in the distribution block. In the last subsection the functionalities of the add-block and integrator-block was described. Depending on whether the current branch is a check-in option with luggage or not the values will be spread over to the "join luggage" block and/or "join passengers" block and will be forwarded to the "luggage collection point"-block or accordingly to the "board card check"-block in figure 9. At the luggage collection point the luggage will be spread over the aeroplanes. Because of the assumption that a passenger just has one piece of luggage there are more passengers than luggage. The luggage branch in figure 11 contains simplified handling of the luggage. The dispatch of the luggage is although important for the simulation for example in figure 10 but it will be touched in this simulation. For further information concerning luggage model see [12].

A frequently used block is the MATLAB function. In these functions for example the limits of each check-in possibility can be set. In figure 10 the first functions of each twig are responsible to forward the receiving values and the second one describes the capacity of the check-in possibility. The only exception is the online check-in without luggage, the values will be forwarded here because of none existence of capacity. The function "carriage of luggage" in figure 9 stores the capacity of the luggage collection point. The capacity of the function "luggage for departure" is defined by the product of the number of luggage cars and their capacity. For example the functions "board card check to security check" and "apron busses" describes their capacity like mentioned before.

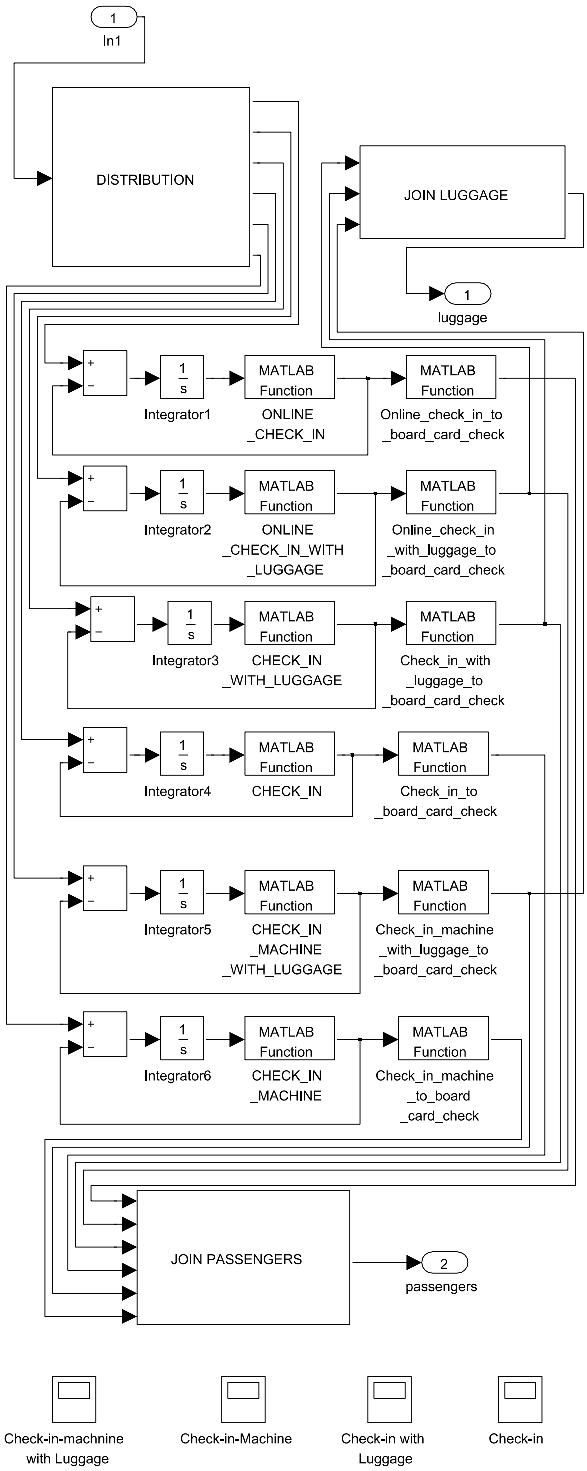

Fig. 10 Check-in

The other functions are responsible for passing the values.

The MATLAB Simulink model for the arrival is shown in figure 11. The arrival is also based on the description in section 2 and the UML model of section 4. The number of arriving passengers was observed under the same assumption like the departure model. The other half of daily arriving passengers in the airport of 17428 was distributed by using table 1. Table 1 is a result of the analysis of the arriving flights of one day at the airport Hamburg [16]. The station “aeroplane handling arrival” contains three factors for each branch: a factor for passengers with luggage to specify the number of luggage and two factors for the passengers using either the aerobridge or apron busses.

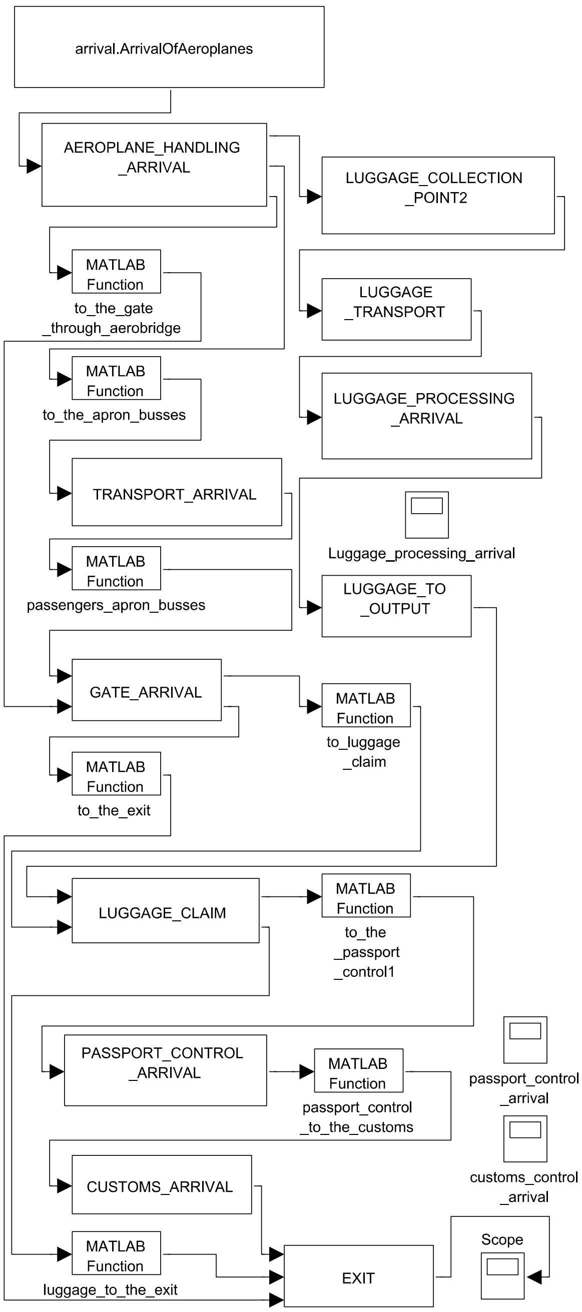

Fig. 11 Arrival

The rest of the procedure is self-explanatory.

5.3 Results of Simulink model

The result of a simulation can be shown by using scope-blocks of the simulation. The used number of counters and their capacity are plausible assumptions and can be changed by the user. Due to the limited scope of this paper only the most important scope-blocks will be represented.

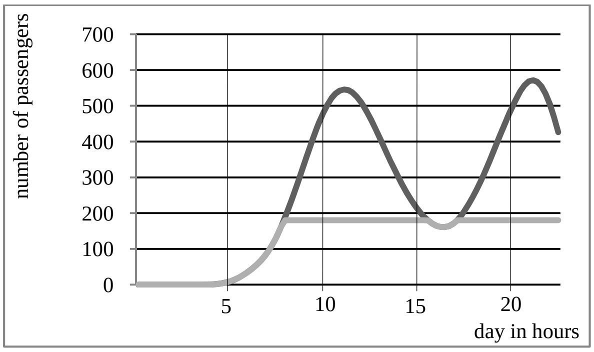

Fig. 12 Check-in with luggage

First some important results of the departure will be exemplary introduced. The check-in with luggage in figure 10 has a capacity of 30 passengers for each of the six counters.

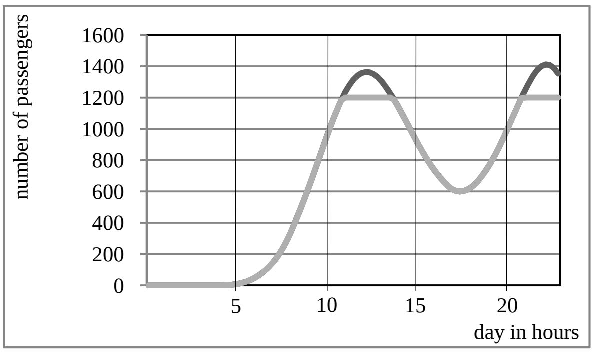

Fig. 13 Board card check

As seen from 8 am a bottleneck occurs and forces the airlines to open more counters. Around half past three pm the bottleneck disappears and occurs again from five pm.

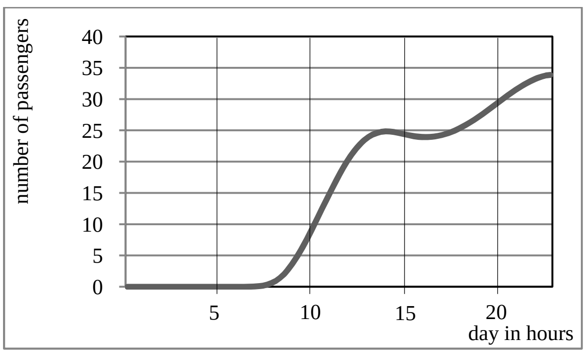

Fig. 14 Passport control arrival

A bottleneck occurs in this station from 180 passengers. This is represented in figure 12. The “board card check” block from figure 9 is shown in figure 13. The number of board card check stations is ten and the capacity for each station is 120. This causes the bottleneck from 1200 passengers. For the arrival process the passport control from figure 11 will be represented exemplary in figure 14. Because of the chosen number of control stations of two and capacity of 40 passengers this curve shows no bottleneck. A bottleneck would arise here from a waiting queue length of 80 passengers.

6 Result

The result of this contribution was to create a macroscopic model and its prototypical implementation in the MATLAB Simulink environment to detect the bottlenecks in the workflows. The implementation of the model was made by description of the workflow and using UML. UML is a powerful model language and delivers a clear model. The main problem of the airport context is the bottlenecks that arise whenever a station in the process is overstrained. The MATLAB Simulink model exposed the bottlenecks and their developments were shown.

The users like the airport operators or airline are able to use this prototypical implementation in MATLAB Simulink in order to get forecasts of any day of the year. They can use input data from their experiences to receive more detailed outputs. This would help them for example to make decisions concerning number of the deployed employees.

But one important fact that cannot be simulated is the customer satisfaction. The challenge is to increase the efficiency and not to rush the passengers through the airport. Economic attitudes of the airport are the shops, too. This is still typically handled manually by interviewing the passengers and to fulfil their desires.

7 Conclusion and Outlook

This contribution delivers a macroscopic model of workflows in an airport and its optimization can be done by using a prototypical implementation in MATLAB Simulink. This implementation offers the airport operators or airlines the possibilities to find the bottlenecks in their workflows.

To increase the plausibility of this contribution the employment of actually measured data would be useful. One of our project partners from the TUHH (Hamburg University of Technology) delivers, inter alia, an air transport demand forecasts (see [17]). It is planned to use these data with the MATLAB Simulink model to deliver more plausible results.

It is also planned to use this concept and the resulting simulation to test the effects of new technologies like: new luggage loading technology, new control systems like sending board card information from a mobile phone or the usage of RFID-chips (Radio frequency identification) for passenger tracking in airport terminals.

On the basis of our research the optimization of the passenger workflow of an airport can be done by using forecasts of simulations.

8 Acknowledgments

This work is financially supported in part by the BMBF via the Cluster of Excellence in Aviation Hamburg project L3 and the University of Hamburg. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding bodies.

9 References

[1] Y. Farschtschi and M. Widemann. diploma thesis. *Modulares Workflow Management und Optimierungssystem am Beispiel des ground handling am Flughafen.* Hamburg : s.n., 2009.

[2] Y. Farschtschi, M. Widemann, J. Wittmann and D. P. F. Möller. Global optimization of local workflow managers. *EUROSIM 2010.* 2010.

[3] M. Widemann, Y. Farschtschi, J. Wittmann and D. P. F. Möller. Workflow management of the ground handling at the airport through modular system optimizing. *EUROSIM 2010.* Mai 2010.

[4] Hamburg Airport: traffic figures. [Online] April 2010. http://www.ham.airport.de/de/u_daten_verkehrszahlen.html.

[5] *Unified Modeling Language.* [Online] Object Management Group. http://www.uml.org/.

[6] C. Kecher. *UML 2: das umfassende Handbuch - UML lernen und effektiv anwenden, alle Diagramme und Notationselemente.* Bonn : s.n., 2009. ISBN: 978-3-8362-1419-3.

[7] D. Pilone and N. Pitman. *UML 2.0 in a Nutshell.* 2005. ISBN: 0596007957.

[8] *MathWorks.* [Online] Mathworks, 2010. http://www.mathworks.com/.

[9] J. Hoffman. *MATLAB und Simulink.* Karlsruhe : Addison-Wesley, 1998. ISBN: 3-8273-1077-6.

[10] Department for Foreign Affairs: The Schengen Agreement. [Online] April 2010. http://www.auswaertiges-amt.de/diplo/en/WillkommeninD/EinreiseUndAufenthalt/Schengen.html.

[11] F. Frahm. bachelor thesis. *Spezifikation und Optimierung des Workflows zur Passagier- und Gepäckabfertigung.* Hamburg : s.n., 2009.

[12] M. Widemann, J. Wittmann, K. Himstedt, D. P. F. Möller. Macroscopic Modelling of luggage

streams in an airport terminal. *EUROSIM 2010.* 2010.

[13] ModelKinetix. [Online] 2003. http://www.modelkinetix.com/.

[14] B. Dreyer. bachelor thesis. *Makroskopische Modellierung von Passagier- und Gepäckströmen im Terminal eines Flughafens mit Modelmaker.* Hamburg : s.n., 2009.

[15] Hamburg, Airport. *Airport car parks.* [Online] Airport Hamburg, May 2010. http://www.airport.de/de/airport_parking.phtml.

[16] Arriving flights at the airport. [Online] 2010. http://www.airport.de/de/ankunft_live.phtml.

[17] C. Blank, C. Gertz and S. Löwa. Domestic German Air - Rail competition depends on. *EUROSIM 2010.* 2010.