Surrogate Modeling of Agent-based Airport Terminal Operations

Benyamin De Leeuw [1], S. Sahand Mohammadi Ziabari [2] [0000-0003-3803-6714], Alexei Sharpanskykh [3] [0000-0001-6879-1134]

[1, 2, 3] Delft University of Technology, The Netherlands
B.DeLeeuw@student.tudelft.nl, S.S.mohamamdiziabari@tudelft.nl, O.A.Sharpanskykh@tudelft.nl

Abstract. The airport terminals are complex sociotechnical systems, which are difficult to understand and their behavior is hard to predict. Hence, an agent-based model, the Agent-based Airport Terminal Operation Model (AATOM), has been designed to represent and analyze diverse airport terminal processes, actors, their behavior and interactions. The main issue with such models is the large computational requirements for simulating detailed processes, making it computationally inefficient. Furthermore, the dynamics of such models are difficult to understand. Therefore, the goal of this research is to approximate the dynamics of AATOM by a surrogate model, while preserving the important system properties. A methodology is suggested for training and validating a surrogate model, based on the Random Forest algorithm. The trained surrogate model is capable of approximating the AATOM simulation and identifying relative importance of the model variables with respect to the model outputs. Firstly, the results obtained contain an evaluation of the surrogate model accuracy performance, indicating that the surrogate model can achieve an average accuracy of 93% in comparison to the original agent-based simulation model. Nonetheless, one indicator, the number of missed flights, has shown to be more difficult to predict, with an average accuracy of 83%. Secondly, the results show that the airport resource allocation has an important impact on the efficiency of the airport terminal, with the two most important variables being the number of desks at the check-in and the number of lanes at the checkpoint. Last, the developed surrogate model was compared with a second Artificial Neural Network-based surrogate model built for the same agent-based model.
Keywords: Surrogate modeling, Agent-based model, Random forest.

1 Introduction

The airport terminal plays a crucial role in the modern air transportation system. Previous studies have focused on modelling and simulating the airport terminal operations, concentrating mainly on security analysis [1, 2]. For this purpose, An Agent-based Terminal Operation Model (AATOM) has been developed for modelling and analysis of complex sociotechnical airport systems with diverse interacting actors. The emergent in such complex systems is hard to understand [3]. Furthermore, AATOM has a high computational complexity. One of the approaches to improve understanding of agent-

based models is surrogate modeling. In essence, it consists of generating a 'model of the model', obtaining an approximation of the original model. Surrogate modeling is used through various domains with the objective to emulate/surrogate an existing agent-based model [4]. Subsequently, the surrogate models are used for calibration [5], validation [6] or behavior space exploration [7].

Hence, two elements form the basis of the research, being the original AATOM model and surrogate modelling to abstract this model, to decrease its high computational complexity. Thus, the research objective is to obtain a computationally efficient AATOM while preserving the important dynamic (emergent) properties of the model and getting an insight into the underlying mechanisms of the model. The former underlines that by applying the surrogate modelling method (Random Forest) on the AATOM model, an approximation of the model can be obtained, preserving the system properties by accurately predicting the model output under given conditions. The latter is the ability of the approximation obtained from the original AATOM model to reveal the relations between the model inputs and outputs, leading to a better understanding of the system behavior. The main contributions of the study are the generation of a surrogate model from AATOM, the evaluation and validation of the surrogate model, and the comparison of the developed surrogate model (the Random Forest model) with another surrogate model (the ANN model) to gain better credibility in the obtained results.

In the following in this paper, background elements regarding AATOM and the surrogate model used for approximation are given in Section 2. Furthermore, the methodology developed for generating the surrogate model based on the AATOM is described in Section 3. Section 4 presents the results obtained by applying the methodology, focusing on the performance of the surrogate model and the input-output relationships. Last, the conclusions are drawn in Section 5.

2 Related Work

The AATOM is an agent-based simulation model used to represent the dynamics of passengers and airport terminal staff in the context of airport terminal operations [2]. It comprises the agents with their properties, the environment and the interaction between the agents and the environment. The agent has a three-layered architecture, with each level adding a layer of abstraction. The three layers are: operational, tactical and strategical. In essence, the three-layered architecture dictates how the agent observes the environment and the other agents, and the way the agents interacts with both, based on the observations. Ultimately the agent is able to make decisions based on the beliefs about the environment and the other agents. For a more detailed description of the AATOM architecture we refer to [2]. The environment is composed of three objects: the areas, the flights and the physical objects. The first being two-dimensional polygons that delimit the different terminal areas (check-in, checkpoint, entrance, gate and facility).

Surrogate modeling is the approach for generating an approximation of the model in order to reduce its complexity while maintaining the dynamic properties of the original model. A surrogate model can be constructed by using a learning algorithm to obtain an abstraction of the model. This study considers two types of surrogate models. The first model is developed using the methodology described in Section 3. The second is

used for comparison purposes and a brief explanation of its elaboration is given. The algorithm chosen for generating the first surrogate model, is the Random Forest. This learning algorithm has been proven to be a reliable and efficient method for working with large data-sets, achieving low computational costs [8]. In addition, the implementation of the algorithm for surrogate modelling is relatively simple [9-11]. Furthermore, there are classification and regression trees, differing on the nature of the target variable, being qualitative or quantitative respectively. In this research only regression trees are considered as the AATOM output target variables are only continuous. The target variables, represent different airport terminal indicators measuring processing times or counting the passenger flux. In addition, the Random Forest method has a specific measure for ranking the different parameters on their relative importance, called the variable importance measure (VIM) [12]. Measuring the relative importance of the variables can identify specific relations between the model variables and indicators. Giving additional insights into the underlying relationships of the original (AATOM) model [13]. The second model is based on Artificial Neural Network (ANN). The key motivation to use ANN is the low computational cost and the capability of approximation. ANNs have good generalization properties, can deal with large datasets and are able to represent nonlinearity. However, vanishing gradient is the main problem of ANN in the back propagation. In order to measure the sensitivity of parameters, different algorithms, mainly focusing on the connection weights in artificial neural networks, have been proposed. In [14] feature selection is described as 'the problem of choosing as small subset of features that ideally is necessary and sufficient to describe the target concept'. The main issue with the previous algorithm [15] was that when the training finished, it naturally assumed that the higher the amount of weights, the more important a parameter while regularization techniques make weights to not increase and become smaller [16]. The method which we used to measure the relative importance of parameters is called variance-based feature importance [17]. This is based on the principle that the more important is a parameter, the morethe weights, which are connected to the corresponding input neurons, vary during the training of the model. To measure the relative importance of each parameter, variances of each weight connected to the input layer is calculated in the training time [18]. The algorithm that we used for computing these variances is an adaptation of Welford's online algorithm [17] for computing these variances.

3 Methodology

A step-wise systematic iterative procedure is defined in order to obtain an accurate approximation of AATOM by surrogate modelling. The schematic representation is given in Figure 1. Configuring the AATOM model is the first step of the procedure. The well-defined AATOM can subsequently be used for simulation in order to generate simulation data for training and validating the surrogate model. The simulation data is pre-processed with the result that the surrogate model can achieve better performance. It consists of defining the relevant parameter values and re-sampling the simulation data to better train the surrogate model. Pre-processing is performed after the first iteration of generating the surrogate model. The following step is training the Random Forest. The training data-set is provided from the simulation data of the second step. The

simulation data is randomly divided into two thirds for training data-set and one third for validation data-set [19]. The fifth step, the evaluation of the surrogate model, is subdivided into the validation of the surrogate model and a scenario-based evaluation. The last step is the comparison of the two surrogate models.

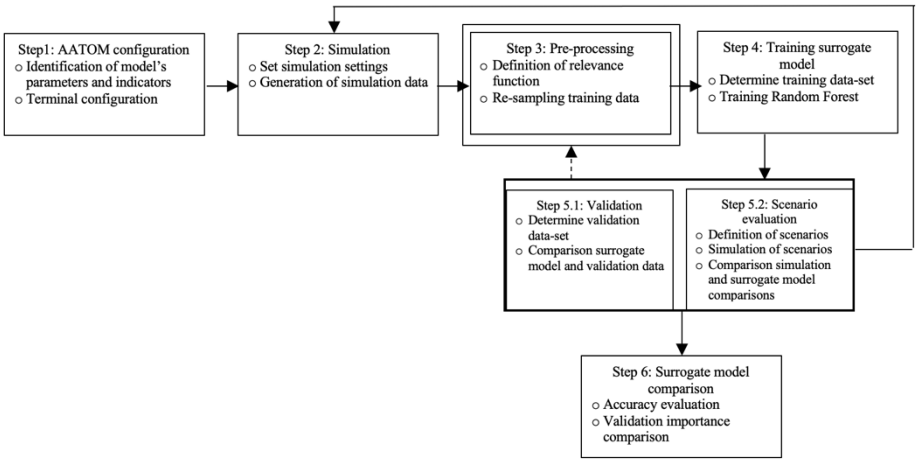


Fig. 1. The schematic representation of the methodology for obtaining a trained surrogate model based on AATOM.

The AATOM has a modular architecture, which requires a specific configuration for simulating airport terminal operations. In Figure 2, the chosen configuration for conducting the surrogate model generation is visually represented. The model consists of a check-in and security checkpoint with each having a queuing system in place. Moreover, the gate area is included in the model. The defined model parameters and indicators. Both are given in Table 1. The input parameters in Table 1 represents a one hour flight schedule, with the first time slot scheduled 400 seconds after the start of the hour and the last slot scheduled 3400 seconds after the start. Each slot is assigned either zero (if no flight is scheduled) or the number of passengers for the scheduled flight. Furthermore, regarding the output given in Table 1, both queuing time and throughput for security checkpoint and check-in are included. For both the training and validation data-set, relatively large pool of sample points is drawn from simulation using the Latin Hypercube sampling method [20].

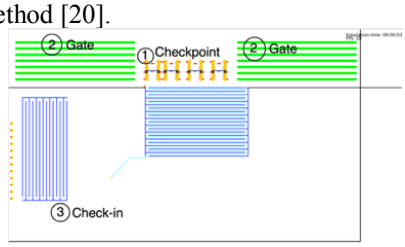


Fig. 2. The AATOM layout including the check-in, security, and gate ([Janssen, 2020]).

Table 1. The model input parameters and output, according to the AATOM architecture, chosen for simulation purpose.

Parameters	Description	Unit
n_{lanes}	Number of lanes at the security check-point	-

$n_{dropoff}$	Number of luggage drop-off points	-
$n_{collect}$	Number of luggage collection points	-
n_{desks}	Number of check-in desks	[s]
$slot_{400}$	Schedule slot at time 400 (sec)	[s]
$slot_{1000}$	Schedule slot at time 1000 (sec)	[s]
$slot_{1600}$	Schedule slot at time 1600 (sec)	[s]
$slot_{2200}$	Schedule slot at time 2200 (sec)	[s]
$slot_{2800}$	Schedule slot at time 2800 (sec)	[s]
$slot_{3400}$	Schedule slot at time 3400 (sec)	[s]
Output		
$scQueue_{avg}$	Average security checkpoint queuing time	[s]
$checkinQueue_{avg}$	Average check-in queuing time	[s]
$throughput_{checkin}$	Number of passengers that passed the check-in	[s]
$throughput_{sc}$	Number of passengers that passed the checkpoint	-
$TimeToGate_{avg}$	Average time to the gate	[s]
$n_{missedFlights}$	Number of missed flights	-

The surrogate model, given a combination of input values, is required to predict the values of the output variables. However, the accuracy of the surrogate model may differ for the different target variables. It is possible that the surrogate is unable to achieve the validity criteria for one or multiple target variables. There exist different strategies for coping with it [21, 22]. The strategy used in this research is the utility-based regression and re-sampling approach. The training data points consist of value combination of inputs and outputs as defined in Table 1. For the Random Forest model, several hyperparameters need to be determined prior to the training such as the number of trees and the number of sample points drawn for each tree. For the purpose of the research, the Bayesian optimization algorithm has been chosen for tuning hyperparameters, proven to be reliable and efficiently applied in previous studies [25, 26].

Lastly, subsequent to training the surrogate model, validating the model is required. The output indicators of the AATOM on the validation set are compared with the outputs from the validation set using the Mean Absolute Percentage Error (MAPE). From literature [19], Random Forest surrogate models often achieve accuracies of more than 90%. The scenario evaluation is based on a set of scenarios that represent real-world airport terminal cases that are given in Table 2. The scenario is a combination of a set of resources, related to the check-in a checkpoint, and a given (one hour) flight schedule. Three different scenarios in Table 2 represents three different levels of passenger demand: low demand (LOW), medium demand (MED) and high demand (HIGH). The levels are based on a regular flight schedule at Rotterdam-The Hague Airport (RTHA). In each scenario, the fixed resource allocation is based on prior research analyzing check-in and checkpoint systems in the context of airport security [1, 27]. One security checkpoint lane has an approximate capability of 160 passenger per hour and one check-in desk is able to process on average 60 passengers per hour. The flight schedules are derived from the flight schedule of RTHA, having on a peak hour an average of six flights scheduled. The representation is an hour of time slots where the value indicates the number of passengers scheduled for that time slot (empty if no flight is scheduled). Moreover, the number of passengers correspond to the Boeing 737 and 738 that are common aircraft at RTHA.

Table 2. The set of scenarios for fixed resource allocation for the different flight schedules, with four resource variables (n_{lanes}, n_{drop}, $n_{collect}$, $n_{checkin}$) and six time slots (from 400 to 3400).

ID	Resource Allocation				Flight Schedule					
	n_{lanes}	n_{drop}	$n_{collect}$	$n_{checkin}$	**400**	**1000**	**1600**	**2200**	**2800**	**3400**
LOW	3	3	3	7	186			186		
MED	5	3	3	12	197		142	197		164
HIGH	7	3	3	19	186	197	186	142	197	197

The methodology proposed for analyzing the sensitivity of the agent-based model using the surrogate neural network model, consists of two phases. In the first phase, the security checkpoint parameters of the AATOM are used as an input for the neural network. The influence of each parameter's uncertainty on the output uncertainty is determined through a series of forced perturbations on the parameters. The variation of the six parameters of interest, and in the second phase, the result of changes with respect to the variation of each parameter is considered for the output layer in the neural network. The training and validation of the neural network for the prediction part are performed, and eventually, the value of the weight from each input is considered as the importance of that parameter. To measure the sensitivity of each parameter such as average queue time in the security check-point, we have used the method proposed in [17].

The range of parameter values are based on standard values for RTHA are given in Table 3. The $n_{flights}$ are divided into time of scheduling and number of passengers. It can take the following values: [0, 142, 153, 164, 175, 186, 197]. Both time slots are combined in an array, e.g. [[1600, 142], [3400, 142]]. The value ranges for the other parameters are determined by the limitation of the simulation model.

Table 3. Value ranges of the AATOM parameters for training.

AATOM parameter	Range
n_{lanes}	[1, 2, 3, 4, 5, 6, 7, 8]
$n_{dropoff}$	[1, 2, 3]
$n_{collect}$	[1, 2, 3]
n_{desks}	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
$n_{flights}$	[1, 2, 3, 4, 5, 6]

4 Results

In this Section results are presented, using the methodology described in Section 3. The evaluation results comprise validation using simulation data and scenarios, with the evaluation given in Table 4. It contains the time performance and accuracy measurements for both the Random Forest model and the ANN model. From Table 4, the Random Forest model has a low training and execution time. Regarding the measured using MAPE accuracy, the overall performance of the Random Forest model is attaining values of 90% and above, with an average accuracy of 92.90%. Moreover, the most accurate prediction, being the *TimeToGate*, is reaching 97.13%. It can be explained by considering that the *TimeToGate* is the indicator based on the most information contained in the simulation model. Nonetheless, the indicator for the number of missed flights is less accurately predicted, only achieving 83.44% accuracy.

Table 4. Comparison of performances between the Random Forest surrogate model and the Artificial Neural Network surrogate model.

Performances indicator	**RF**	**ANN**
Training time [sec]	3.82	12.54
Execution time [sec]	0.12	0.91
Mean Absolute Percentage Error		
$scQueue_{avg}$	96.02%	94.39%
$checkinQueue_{avg}$	91.66%	93.71%
$throughput_{checkin}$	94.71%	91.29%
$throughput_{sc}$	94.44%	95.61%
$TimeToGate_{avg}$	97.13%	98.81%
$n_{missedFlights}$	83.41%	80.12%
Mean Accuracy	92.90%	93.87%

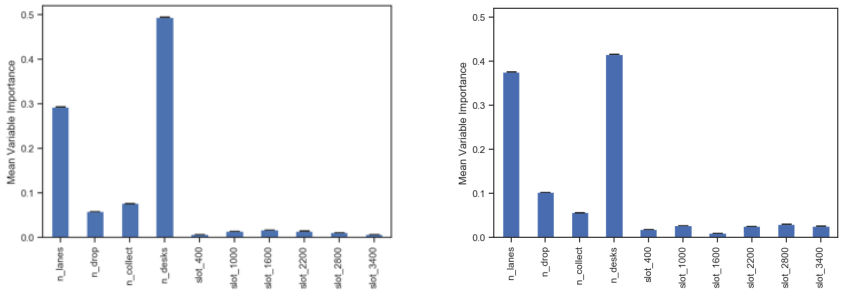
Furthermore, comparing the Random Forest and ANN model, it can be seen that the ANN model is slower on both the training and execution time. However, the ANN model is still largely faster than the AATOM. On the accuracy, both models have similar performances. The *TimeToGate* is also the most accurate prediction for the ANN model. Lastly, the same difference between the accuracy of the $n_{missedFlights}$ and the other indicators is present in the ANN case, only reaching 80.12%. Successively, the second evaluation is given in Table 5. The accuracy of the surrogate model is given for each indicator at every level of passenger demand. First, the model has on overall poor performance on the LOW scenario compared to the two other scenarios, with the largest differences on the accuracy for the two checkpoint indicators and the $n_{missedFlights}$. The most apparent reason is the lack of training data sampled around the region of the variable values defined in the LOW scenario. The *TimeToGate* performance is a direct consequence of the poor performance on the checkpoint indicators, measuring the behavior.

Table 5. The prediction for different scenarios from the trained RF surrogate model.

ID	$checkinQueue_{avg}$	$scQueue_{avg}$	$TimeToGate_{avg}$	$n_{missedFlights}$	$throughput_{checkin}$	$throughput_{sc}$
LOW	89.13%	60.56%	80.01%	0%	94.1%	75.88%
MED	93.58%	88.62%	92.64%	8.81%	92.82%	98.21%
HIGH	88.17%	94.95%	95.38%	61.53%	91.45%	94.94%

Furthermore, the surrogate model's low accuracy on the $n_{missedFlights}$ is also visible in the results of the two other scenarios. There is a visible increase in accuracy with increasing passenger demand, growing from 0% to 61.53%. Moreover, the surrogate model was trained on a wide range of cases with highly varying $n_{missedFlights}$ values. Additional insight is given into the underlying relationships between the parameters and indicators of the original model (the AATOM). The first analysis is the measure of importance of the different variables. The variable importance measure (VIM) is calculated for both surrogate models (Random Forest and ANN), given in Figure 3. As can be seen in Figure 3(a), there are two distinct variables, identified by VIM as the most important variables: the number of lanes of the security checkpoint and the number of desks at the check-in. Both variables are regulating the processing capacity of passengers at the two systems (checkpoint and check-in), therefore the indication from the VIM conforms with the representation of the two variables in the airport terminal operations.

Moreover, from Figure 3(a), the four resource allocation variables (n_{lanes}, $n_{dropoff}$, $n_{collect}$, n_{desks}) are relatively more important than the flight schedule variables (time-slots). All indicators, except for the $n_{missedFlights}$, are measuring time performance through queueing or throughput. Hence, the resource allocation is more directly related to the time performance of the terminal than the flight schedule, dictating the check-in and checkpoint behavior. The last observation in Figure 3(a) is the trend of the time-slot importance, with the first and last slot being the least important slots. Presumably, the importance of the middle slots stems from the effect of reducing the time between flights by adding more slots in the scheduling hour. The VIM of the ANN model is given in Figure 3(b) for comparison purposes with the RF model. Multiple similarities are perceivable between the two models, with the most recognizable that the same two most important variables (n_{desks} and n_{lanes}) are identified with the VIM. Additionally, the resource allocation variables are prominently more important than the flight time-slots. Besides, there are several differences between the RF model's VIM and the ANN model's VIM. First, the importance measure difference between the two most important variables is lower. The lower difference is possibly due to the use of a different surrogate model, hence both models are not using the variable information in exactly the same manner to make predictions on the different target variables. Second, the ranking of the collect and drop-off location variables is reversed. Nonetheless, the order of magnitude is similar for both variables in both models.



(a) RF model (b) ANN model

Fig. 3. The variable importance measure for the AATOM model parameters from the default RF and the ANN surrogate model

Furthermore, an analysis of the surrogate model accuracy is given by systematically adding variables to the model. At each addition the predictions made by the surrogate model are compared with the validation data-set. The different mean accuracy levels obtained for both models are given in Table 6. Figure 4 visualizes the accuracy change by adding model variables. From Table 6 and Figure 4, it can be seen that the four resource variables are supporting the RF model to make predictions with an averaged accuracy of 86.82%. Adding the remaining variables increases the mean accuracy by 6%. Hence, clearly observable in Figure 4, the resource allocation variables have the largest contribution to the increase in prediction accuracy of the surrogate model. For the RF model, there is the small reduction of accuracy after the addition of the first time-slot variable. However, the difference is sufficiently small (0.07%) to be neglected. Furthermore, by comparing the RF and ANN model in Table 6, several similarities can be observed. First, the resource variables are sufficient for the ANN model to reach an accuracy of 89.77 %. The remaining variables, similar to the RF model, are only slightly increasing the prediction accuracy by 4%. Second, the most important

variable is the n_{desks}, related to similar VIM. Nonetheless, there is an observable difference between the contribution of the time-slot variables in the RF model and the ANN model. The differences coincide with the VIM of the variables in Figure 3. As mentioned earlier, the $n_{missedFlights}$ is the one target variable for which the surrogate model experiences difficulties to make accurate predictions. The accuracy is lower for the specific scenarios than for an evaluation based on the validation data. In essence, the nature of the $n_{missedFlights}$ is different from all other indicators of the AATOM simulation. All other indicators are measuring the efficiency in time or passenger count of the check-in and/or checkpoint. The $n_{missedFlights}$ is not directly explained by the dynamics of either of the systems, i.e. check-in or security checkpoint. This target variable is more evenly dependent on all the operational elements in the airport terminal. This can be observed in the VIM for predicting the $n_{missedFlights}$ and the incremental accuracy by variable addition, given in Figure 5.

Table 6. The prediction for different scenarios from the trained RF surrogate model.

ID	Parameters	Included Parameters	RF	ANN
1	n_{lanes}	[1]	62.53%	68.03%
2	$n_{dropoff}$	[1,2]	65.50%	72.24%
3	$n_{collect}$	[1,2,3]	70.97%	76.91%
4	n_{desks}	[1,2,3,4]	86.82%	89.77%
5	$slot_{400}$	[1,2,3,4,5]	87.75%	90.01%
1	$slot_{1000}$	[1,2,3,4,5,6]	87.74%	90.80%
2	$slot_{1600}$	[1,2,3,4,5,6,7]	89.89%	92.26%
3	$slot_{2200}$	[1,2,3,4,5,6,7,8]	91.32%	92.89%
4	$slot_{2800}$	[1,2,3,4,5,6,7,8,9]	92.38%	93.15%
5	$slot_{3400}$	[1,2,3,4,5,6,7,8,9,10]	92.88%	93.76%

According to Figure 5(a), the n_{lanes} is the most important variable. The average time to check a passenger at a checkpoint lane is higher than for the check-in. Hence, the number of lanes present at the checkpoint largely determine the number of passengers that can arrive on time for the flight. Furthermore, from Figure 5(a), the importance of all remaining variables is approximately evenly distributed. Thus, it indicates the necessity of all variable information to achieve adequate accuracy (above 80%) on the validation data. The figure depicts the change in accuracy by step-wise addition of the variables. In contrast with the same plot from Figure 4, the accuracy is linearly increasing with every variable addition.

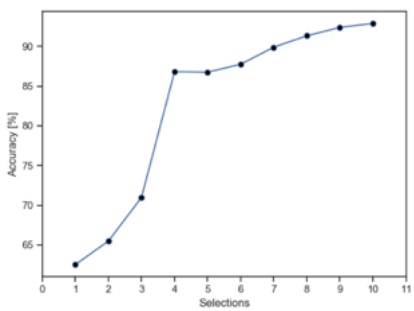
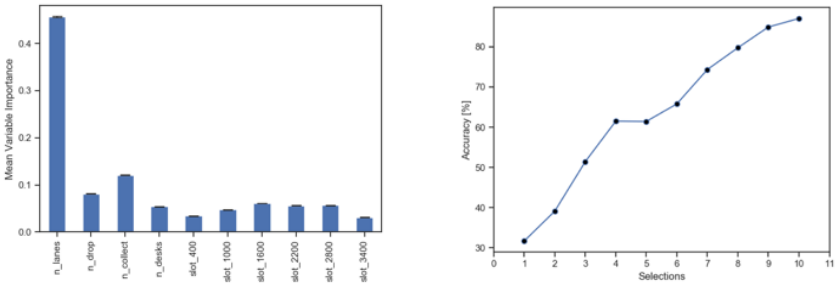


Fig. 4. The model accuracy with addition of variables from the RF surrogate model.



(a) VIM (b) Accuracy

Fig. 5. The variable importance and model accuracy with addition of variables from the RF surrogate model for predicting the number of missed flights.

Moreover, the surrogate model is predicting less accurately the smaller number of missed flights. The chosen measure for accuracy (MAPE) determines the absolute differences between the prediction of the surrogate model and the simulation result. Hence, for smaller numbers of $n_{missedFlights}$ the difference in percentage are larger, for example predicting 4 instead of 6 reveals an inaccuracy of 40%. Second, the number of missed flights occurring over the flight hour schedule is a rare event, especially rare for the lower values. The infrequent occurrence complicates the task of training and predicting this indicator. Last, the range of values for the $n_{missedFlights}$ is wide, visualized in Figure 6. Hence, it is more complex for the surrogate model to achieve high accuracy on the whole spectrum. However, utility-based regression and the method for re-sampling, did not improve the accuracy of the surrogate model. The method has been applied on re-sampling the data to under-sample the higher values of the number of missed flights.

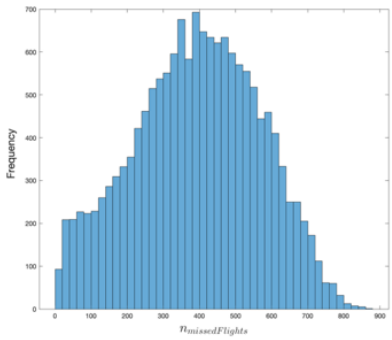


Fig. 6. Distribution of the $n_{missedFlihgts}$ from the dataset, with the number of missed flights over the number of occurrences in the dataset.

5 Discussion and Conclusion

The research purpose was to study an approach for using an approximation of the AATOM in order to make predictions on the behavior of airport terminal operations in a computationally efficient manner. The study included two additional aspects: preserving the properties of the original model and getting an insight in the underlying

dynamics of the AATOM. The method consists of an iterative process in which the AATOM generates simulation data for training and validation of the surrogate model. The surrogate model used in the method is a random forest model. The trained surrogate model was evaluated based on the absolute difference between the model predictions and the simulations results from the validation data-set. An additional evaluation was made on specific scenarios for different levels of passenger demands. Furthermore, an additional surrogate model was developed for comparison, based on ANN. The ANN model is also evaluated on the validation data-set and provided additional insight on the underlying relationships in the AATOM. Regarding the approximation, the obtained the RF-based surrogate model is able to make accurate predictions on all indicators except one, the number of missed flights. It achieves an average accuracy of 93%. For the number of missed flights, the accuracy is reaching 83%. Further analysis highlights the wide range of number of missed flights simulated and the difficulty for the surrogate to make accurate predictions on the lower numbers. Moreover, the evaluation on the scenarios has shown the generalization capability of the surrogate model, with the same difficulty to make accurate predictions of the number of missed flights. The ANN model attains similar accuracy on all indicators and encounters the same issue with the predictions on the number of missed flights. In general, the results have shown through accuracy evaluation that the emergent properties, represented by the model indicators, are preserved by achieving acceptable accuracy levels.

References

1. Janssen, S., Sharpanskykh, A., and Curran, R. (2019). Agent-based modelling and analysis of security and efficiency in airport terminals. Transportation research part C: emerging technologies, 100:142–160.
2. Janssen, S. (2020). Capturing agents in security models: Agent-based security risk management using causal discovery.
3. Lee, J.-S., Filatova, T., Ligmann-Zielinska, A., Hassani-Mahmooei, B., Stonedahl, F., Lorscheid, I., Voinov, A., Polhill, J. G., Sun, Z., and Parker, D. C. (2015). The complexities of agent-based modeling output analysis. Journal of Artificial Societies and Social Simulation, 18(4):4.
4. Van der Hoog, S. (2018). Surrogate modelling in (and of) agent-based models: A prospectus. Computational Economics.
5. [Lamperti et al., 2018] Lamperti, F., Roventini, A., and Sani, A. (2018). Agent-based model calibration using machine learning surrogates. Journal of Economic Dynamics and Control, 90:366–389.
6. Zhang, Y., Li, Z., and Zhang, Y. (2020). Validation and calibration of an agent-based model: A surrogate approach. Discrete Dynamics in Nature and Society, 2020:6946370.
7. Edali, M. and Yücel, G. (2019). Exploring the behavior space of agent-based simulation models using random forest metamodels and sequential sampling. Simulation Modelling Practice and Theory, 92:62–81.
8. Villa-Vialaneix, N., Follador, M., Ratto, M., and Leip, A. (2012). A comparison of eight metamodeling techniques for the simulation of n2o fluxes and n leaching from corn crops. Environmental Modelling Software, 34:51–66.

9. Biau, G. and Scornet, E. (2016). A random forest guided tour. Test, 25(2).
10. Boulesteix, A., Janitza, S., Kruppa, J., and König, I. R. (2012). Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2(6):493–507.
11. James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An Introduction to Statistical Learning.
12. Behnamian, A., Millard, K., Banks, S. N., White, L., Richardson, M., and Pasher, J. (2017). A systematic approach for variable selection with random forests: achieving stable variable importance values. IEEE Geoscience and Remote Sensing Letters, 14(11):1988–1992.
13. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830.
14. Kira, K. and Rendell, L. A. (1992). The feature selection problem: Traditional methods and a new algorithm. In Aaai, volume 2, pages 129–134.
15. Garson, D. G. (1991). Interpreting neural network connection weights.
16. Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). Deep learning, volume 1. MIT press Cambridge.
17. Welford, B. (1962). Note on a method for calculating corrected sums of squares and products. Technometrics, 4(3):419–420.
18. Sadeghyan, S. (2018). A new robust feature selection method using variance-based sensitivity analysis. arXiv preprint arXiv:1804.05092.
19. Kleijnen, J. P. C. and Sargent, R. G. (2000). A methodology for fitting and validating metamodels in simulation. European Journal of Operational Research, 120(1):14–29.
20. Kleijnen, J. P., Sanchez, S. M., Lucas, T. W., and Cioppa, T. M. (2005). State-of-the-art review: a user's guide to the brave new world of designing simulation experiments. INFORMS Journal on Computing, 17(3):263–289.
21. Branco, P., Torgo, L., and Ribeiro, R. (2015). A survey of predictive modelling under imbalanced distributions. arXiv preprint arXiv:1505.01658.
22. Torgo, L., Branco, P., Ribeiro, R. P., and Pfahringer, B. (2015). Resampling strategies for regression. Expert Systems, 32(3):465–476.
23. Probst, P., Wright, M. N., and Boulesteix, A. (2019). Hyperparameters and tuning strategies for random forest. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(3): e1301.
24. Díaz-Uriarte, R. and Alvarez de Andrés, S. (2006). Gene selection and classification of microarray data using random forest. BMC Bioinformatics, 7(1):3.
25. Bergstra, J., Bardenet, R., Bengio, Y., and Kégl, B. (2011). Algorithms for hyperparameter optimization. Advances in neural information processing systems, 24:2546–2554.
26. Wu, J., Chen, X.-Y., Zhang, H., Xiong, L.-D., Lei, H., and Deng, S.-H. (2019). Hyperparameter optimization for machine learning models based on bayesian optimization. Journal of Electronic Science and Technology, 17(1):26–40.
27. [Janssen et al., 2020] Janssen, S., van der Sommen, R., Dilweg, A., and Sharpanskykh, A. (2020). Data-driven analysis of airport security checkpoint operations. Aerospace, 7(6):69.