Towards a Better Understanding of Agent-based Airport Terminal Operations Using Surrogate Modeling

Benjamin C.D. De Bosscher[1], S. Sahand Mohammadi Ziabari[1,2 [0000-0003-3803-6714]], Alexei Sharpanskykh[1 [0000-0001-6879-1134]]

[1] Delft University of Technology, Delft, The Netherlands
[2] University of Amsterdam, Amsterdam, The Netherland
B.C.D.DeBosscher-1@student.tudelft.nl, s.s.mohammadizibari@uva.nl, O.A.Sharpanskykh@tudelft.nl

Abstract. Airport terminals are complex sociotechnical systems, in which humans interact with diverse technical systems. A natural way to represent them is through agent-based modeling. However, this method has two drawbacks: it entails a heavy computational burden and the emergent properties are often difficult to analyze. The purpose of our research is therefore to accurately abstract and explain the dynamics of airport terminal operations by means of computationally efficient and interpretable surrogate models, based on an existing agent-based simulation model. We propose a methodology consisting of two stages. Stage I involves the development of faithful surrogates. Stage II then applies state-of-the-art techniques from the emerging field of explainable artificial intelligence to these models. Both model-agnostic and model-specific methods are considered, and their results are synthesized in order to explain the emergent properties. We prove the efficacy of this approach by conducting two case studies on AATOM, an existing Agent-based Airport Terminal Operations Model. Altogether, we clearly observed the preservation of emergent phenomena in surrogate models, and conclude that their combination with interpretable machine learning is an effective way to explain the dynamics of complex sociotechnical systems.

Keywords: Agent-based Modeling, Surrogate Modeling, Interpretable Machine Learning, Airport Terminal.

1 Introduction

In recent decades, the aviation sector has benefited from stable growth in air traffic demand. While long-term prospects have long been taken for granted, abrupt events such as a financial crisis or the outbreak of a disease have shown the vulnerability of this supposition [1]. Furthermore, a growing number of people are also becoming concerned about the environmental impact [2]. It proves that airlines should operate more agile and lean: they must react quickly to such events and adapt to the new status quo. Airports, in turn, are the infrastructural epicenter of the system, so their operations are directly affected by changes in passenger numbers. This demonstrates the need for reliable models of terminal operations. Such models would be useful to prevent chaotic events, like in the aftermath of the COVID-19 pandemic at European airports [e.g., 3, 4, 29]. The modeling of airport terminal operations has been previously explored by numerous scholars. Pao-Yen Wu and Mengersen [5] summarized these efforts in a meta-study, wherein was concluded that agent-based simulation models are most commonly used for operational planning and design purposes. Indeed, such models are preeminent for high levels of detail without compromising the complexity and emergent properties of sociotechnical systems like an airport terminal [6]. Notwithstanding, their computational requirements are often substantial, which might become a limiting factor as the scale of the simulation increases. To address this limitation, a worthy alternative is the consideration of surrogate modeling, also known as meta-modeling. A surrogate mimics model responses through so-called black-box approximation functions [7]. Fundamentally, the principle is subject to a dichotomy between savings in computational requirements and fidelity to the original model [8]. It is presumed to be viable as long as the reduction in computation time justifies the associated lower level of accuracy [9]. We intend to achieve the objective by proposing the following two-stage methodology. The starting point is AATOM, which is the abbreviation for agent-based airport terminal operations model. It was recently designed and calibrated by Janssen et al. [10], and has been further developed ever since. AATOM is known for its high-fidelity to the actual terminal system, although it suffers from large computational requirements. Hence, the first stage of our methodology relates to the creation of surrogate models. This includes generating a data set, training black-box functions, and validation. The process is not necessarily linear, as data collection can be combined with training surrogates — commonly known as active learning or adaptive sampling [11].

The paper is organized as follows. It starts with compiling the theoretical background in Section 2. After that, Section 3 further elaborates on the methodology based on its two stages. The results are described next in Section 4: the main principles behind the model are illustrated, along with the specific settings for the simulations. Finally, the conclusion is drawn up in Section 5.

2 Related work

Airport terminals are central to passenger handling. It is the place where departure, arrival, and transfer flows congregate, each of which has its own characteristics and goals [12]. In particular, we focus on the departure flow of passengers. Typical activities include the check-in, security check, border control for international destinations, and possibly some non-aeronautical activities such as shopping or dining [13]. Scholars have shown great interest in modeling these processes as they are subject to stochasticity and non-trivial complexity inherent in natural human behavior. Tosic [14] is one of the earliest available review studies, yet it has not lost its relevance. The author identifies several ingredients of modeling airport terminal operations, the most pertinent of which are the following. First of all, the demand of air traffic is usually forecast with traditional statistics. The more recent literature has further subdivided it into problems with strategic, tactical and operational horizons [e.g., 15, 16]. Secondly, one can also consider specific physical locations; examples are single check-in counters or border control. They are often modeled using queuing theory, where performance is measured by quality of service. The results can then be directly benchmarked against the International Air Transport Association expectations [17]. Analytical approaches are quick, exact and not overly complicated. However, this affects their fidelity to the real world [18]. A simulation-based approach is therefore preferable if the system entails a certain degree of complexity. Lastly, the entire terminal building can be taken into account at once. In the end, individual processes influence one another and as such contribute to the overall emergent properties. Depending on the level of detail, one can still distinguish between microscopic, mesoscopic and macroscopic models, although the former is rather the standard when considering operational flows [18]. Alternatively, the more recent meta-study of Pao-Yen Wu and Mengersen [5] differentiate existing airport terminal models according to their use case. They identified four purposes: capacity estimation, operational planning, security risk evaluation, and performance measurement. This becomes particularly interesting in combination with Tosic [14], as it reveals appropriate methods to realize our research ambitions. Notably, most models to represent the operations of an entire departure flow seem to be agent-based. That is a microscopic bottom-up approach capable of simulating the behavior of individual passengers, along with the interactions between them and the environment [5]. It became particularly relevant as computing power increased over the years, giving researchers the opportunity to create simulation models that are meticulously close to reality [19]. Hence, agent-based modeling is indeed very suitable if one requires detailed information about terminal processes, which is crucial for understanding emergent properties. In line with the above observations and suggestions, Janssen et al. [10] have recently developed such an agent-based architecture and a simulator for airport terminal operations. Despite currently available technology, advances to understand complex systems in detail are often hampered by computational limitations. This has encouraged the development of surrogate modeling, which aims to fit black-box functions between the input and output of an expensive model in an attempt to accurately mimic its behavior [8, 7].

3 Methodology

As relevant dimensions of the research have been touched upon in the theoretical background, current section continues with the methodology. A high-level overview is depicted in Figure 1. The first step is to define and prepare the agent-based model of interest. This model is typically highly detailed and close to reality, but computationally intensive. Consequently, there are two reasons that make surrogate modeling an attractive alternative. On the one hand, it gives access to much faster models. On the other hand, they enable us to better understand the underlying system — recall that agent-based models reveal the emergent properties only a posteriori, thereby requiring numerous simulation runs [20]. These two reasons are reflected in stage I and stage II of the methodology. The former consists of sampling, fitting surrogate models, and validation. The latter is concerned with agnostic and specific analyses, after which their outcome is validated through triangulation. Finally, the results of the second stage are synthesized in order to interpret and understand the complex dynamics and emergence of the focal agent-based model.

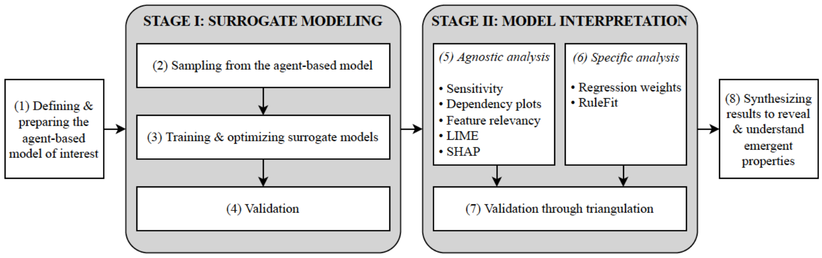


Figure 1: High-level overview of the two-stage methodology.

The surrogate modeling process commences with the creation of a training data set. This is often referred to as the design of experiments (DOE), which aims to extract as much statistical information as possible from the focal agent-based model [8]. Once an initial sample is available, the general procedure is to iteratively evaluate a meta-model and select a new point to sample until some stopping criterion is reached [21]. To demonstrate the applicability of our two-stage methodology, we aim to detect and explain emergent phenomena in a complex sociotechnical system. It follows from the theoretical background in Section 2 that a passenger terminal is the epitome of such a system: cognitive, social, technical, and organizational factors play a major role. With this in mind, Janssen et al. [10] recently developed an agent-based airport terminal operations model (AATOM) — existing alternatives were not accurate enough, too generic, too difficult to use, or the source code was not openly available. One of its main features is that it contains prebuilt components, such as check-in desks or a security checkpoint [10]. Some recent examples are Janssen et al. [22] on the relationship between checkpoint security and efficiency, Janssen et al. [23] on the management of airport security risks, and Mekic et al. [6] with an analysis on non-aeronautical activities and their impact on terminal operations. An agent-based model is known to consist of an environment, agents, and interactions between them [24]. These three components of AATOM are now briefly discussed in respective order. First of all, the environment contains all elements of an airport terminal. That includes various functional areas with physical objects, but also more abstract items such as flights [25]. The former is depicted in Figure 2, which resembles the terminal layout of Rotterdam The Hague Airport (RTHA). We specifically opted for RTHA due to the availability of data and associated insights from a previous study (see [26]). The layout was also available in the latest version by Mekic et al. [6]. Regarding the flights, we consider a typical morning rush hour at RTHA. Secondly, cognitive agents are the key players in an environment. Three types can be defined in AATOM: passengers, operators, and orchestrators [25]. The former are trivial, operators are generally the employees in the terminal (e.g., security officers, check-in staff, cashiers, etc.), and orchestrators help with coordination and monitoring (e.g., employees who open or close check-in counters based on an airport's strategy). Agents have certain goals on which they act and interact accordingly. The final component of AATOM is that agents can interact with the environment as well as with each other. The model reflects these two types of interaction in many different ways [25]. For example, check-in employees managing flights or security officers using sensors at the checkpoint are concrete cases of interaction between agents and the environment.

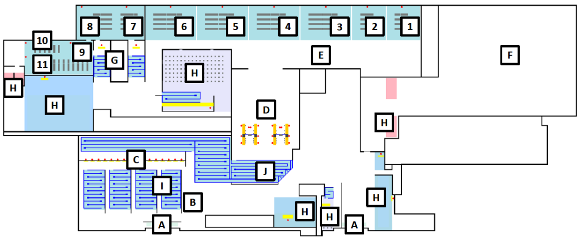


Figure 2: Terminal layout of RTHA represented in the model. Passengers arrive through entrances (A) in the public area (B). Those who have not checked-in online can do so at the counters (C) via designated queues (I). Thereafter, all passengers continue via queues (J) to the security checkpoint (D) to access the restricted area. This area is split up into a departure hall (E) and an arrival hall (F). The arrival hall is not further developed as our research focuses on solely the outbound passenger flow. The departure hall has gates 1 to 6 for flights with destinations in the Schengen area and gates 7 to 11 for flights outside the Schengen area (the numbers on the map correspond to the gate numbers). To access the latter gates, passengers should go through border control to have their passports checked (G). Along the journey, passengers are free to make use of the facility areas for non-aeronautical activities (H) [10].

For more detailed information, the reader is referred to Janssen et al. [25] and Janssen et al. [10] as the key principles behind the architecture of AATOM have now been touched upon. Mekic et al. [6, pp. 20–21] made a comprehensive overview, though two examples are the distribution of the time required for checking-in at airport counters and the distribution of passengers arriving at the terminal. We use the defaults for most of these settings. The remaining features that are considered variables for our case study are listed in Table 1.

Table 1: Considered input parameters of AATOM. A remark regarding the number of passengers is that Pax_t is defined for every available time slot t. In other words, if RTHA has 7 available time slots during the simulated time frame, the model requires 10 input parameters (i.e., 7 parameters to define the number of passengers and 3 strategy parameters).

Input parameter	**Unit**	**Explanation**
Pax_t	[#]	An integer indicating how many passengers are traveling on the flight on time slot t. It is strictly positive, bounded by the maximum capacity of an aircraft. If the occupancy rate is below 50%, it becomes 0 because the flight is canceled.
$CTG_{\mathbf{strategy}}$	[s]	A positive real number that determines the time when passengers are called to their gate prior to the departure time. It represents the airport's call-to-gate (CTG) strategy [48].
$CI_{\mathbf{strategy}}$	[-]	An integer that determines the number of open check-in counters over time. It represents the airport's check-in (CI) strategy. An orchestrator agent couples the number with a predefined strategy [48].
$SC_{\mathbf{strategy}}$	[-]	An integer that determines the number of open lanes at the security checkpoint over time. It represents the airport's security check (SC) strategy. An orchestrator agent couples the number with a predefined strategy [48].

Furthermore, similar to the input, the output parameters are presented in Table 2. AATOM allows a user to define and extract any indicator, so again a selection has to be made. We believe that the listed metrics yield a solid indication of the airport's passenger handling performance, hence no other indicators are defined and these will form the basis of the analysis. Now that all important aspects of AATOM have been described, the next section continues with applying our proposed methodology on the focal model. It includes the outcome of both stage I and stage II, with the ultimate purpose of explaining interesting dynamics and emergent properties of terminal activities related to the entire departure flow in RTHA.

Table 2: Relevant key performance indicators of AATOM.

Output parameter	Unit	Explanation
AvgQueueTime_{CI}	[s]	Indicates the average time that passengers wait in a queue until they can be served at an available check-in (CI) counter.
AvgQueueTime_{SC}	[s]	Indicates the average time that passengers wait in a queue until they can be served at a security checkpoint (SC) lane.
$\text{MaxPaxInQueue}_{CI}$	[#]	Indicates the maximum queue size at check-in during the simulated time frame.
$\text{MaxPaxInQueue}_{SC}$	[#]	Indicates the maximum queue size at security during the simulated time frame.
AvgTimeToGate	[s]	Indicates the average time it takes passengers to get to their gate. It is counted from the moment they arrive at the airport.
PaxCompleted_{CI}	[#]	Indicates the total number of passengers that have completed the check-in (CI) activity at the airport counters (i.e., the throughput at check-in).
PaxCompleted_{SC}	[#]	Indicates the total number of passengers that have completed the security check (SC) activity at the checkpoint (i.e., the throughput at security).
NumMissedFlights	[#]	Indicates the total number of passengers who could not reach their gate at the time of departure.
TotalExpenditure	[€]	Indicates the amount of money that all passengers together have spent during their non-aeronautical activities [48].

4 Results

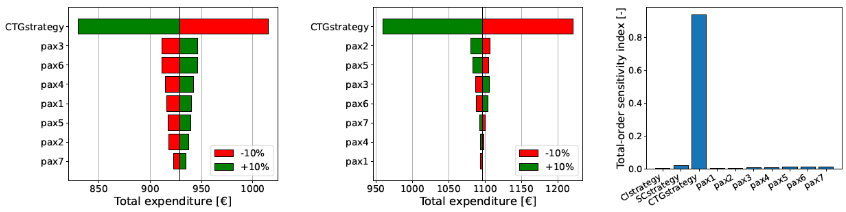
Before assessing fidelity, one must first collect data and tune the surrogate model architectures. We visualize the distribution of selected data points, analyze summary statistics of the responses, and show how the stopping criterion of the active learning scheme is reached. It turns out that the training sample is sufficiently informative with 300 data points in total. This automatically leads to validation and test sets with both 100 additional observations, so that the proportion of each equals 20% of the entire sample. With that, the next step is to evaluate the surrogates' out-of-sample performance. We perform validation in Table 3. Per output parameter of AATOM, the metrics are calculated for Gaussian process regression (GP), polynomial regression (LR), random forests (RF), and gradient boosting regression (GB) — the four selected metamodel architectures. The R2 and MAPE are dimensionless, but the RMSE and MAE are expressed in the same unit as the corresponding response, given in Table 2. Finally, the surrogate model that yields the best performance for each response is indicated by an asterisk. The first thing that immediately stands out is the disappointing generalization power of random forests. Their performance is clearly inferior to the other architectures, often with quite a large discrepancy. The initial hypothesis was that overfitting posed the issue, although their accuracy no longer improves near the stopping criterion of the active learning algorithm, nor did regularization help. This is in stark contrast to LR, GP and GB, whose performance is actually rather impressive. While the validation metrics of these three architectures are generally comparable, regularized polynomials seem to mimic AATOM most accurately. Namely, they have been selected 7 out of 9 times, with only the throughput at check-in and security being better estimated by gradient boosting. This may be somewhat surprising, but a plausible explanation could be that the associated responses behave similarly according to the format of higher-order polynomials.

Table 3: Validation of the surrogate model performance.

	PaxCompleted$_{SC}$				AvgTimeToGate				PaxCompleted$_{CI}$			
Metric	GP	LR	RF	GB*	GP	LR*	RF	GB	GP	LR	RF	GB*
R^2	0.90	0.90	0.79	0.93	0.91	0.92	0.55	0.89	0.93	0.94	0.86	0.98
RMSE	17.52	17.62	25.75	14.94	64.32	61.34	143.80	72.14	7.55	7.40	11.17	4.14
MAE	12.59	12.86	20.29	11.51	48.11	46.19	113.45	58.27	5.02	5.66	8.11	3.30
MAPE	0.02	0.02	0.03	0.02	0.04	0.04	0.09	0.05	0.01	0.02	0.02	0.01
	AvgQueueTime$_{SC}$				NumMissedFlights				TotalExpenditure			
	GP	LR*	RF	GB	GP	LR*	RF	GB	GP	LR*	RF	GB
R^2	0.90	0.92	0.57	0.86	0.70	0.80	0.53	0.43	0.97	0.98	0.94	0.97
RMSE	68.63	63.46	142.90	80.69	9.28	7.51	11.58	12.85	52.05	42.44	69.23	52.25
MAE	52.50	49.54	118.74	64.33	7.09	4.42	6.94	6.60	40.34	33.76	55.61	42.00
MAPE	0.08	0.08	0.18	0.10	N/A†	N/A†	N/A†	N/A†	0.03	0.03	0.04	0.03
	AvgQueueTime$_{CI}$				MaxPaxInQueue$_{SC}$				MaxPaxInQueue$_{CI}$			
	GP	LR*	RF	GB	GP	LR*	RF	GB	GP	LR*	RF	GB
R^2	0.91	0.95	0.87	0.95	0.91	0.92	0.65	0.91	0.90	0.95	0.78	0.92
RMSE	19.46	14.13	23.73	15.13	9.15	8.69	17.99	9.06	0.58	0.43	0.86	0.51
MAE	13.79	9.78	16.16	10.25	6.60	6.69	14.59	7.24	0.43	0.29	0.66	0.39
MAPE	0.05	0.04	0.06	0.04	0.07	0.06	0.14	0.07	0.04	0.02	0.05	0.03

*Best performing surrogate model architecture for the associated response
†Mathematically undefined because of division by zero

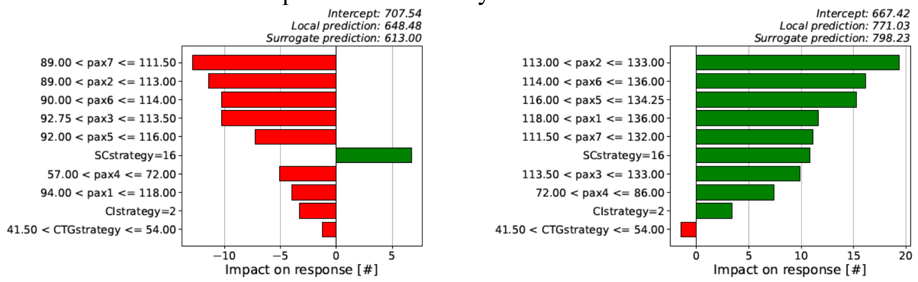
Their combination then naturally produces superior results. For instance, the average queuing time at security is resembled with an expected error of about 1 minute and the total expenditure with an error of about 40 euros. Only the number of missed flights appears to be more challenging: the coefficient of determination decreases to 0.80. Yet, even that is still acceptable, because it is the response most influenced by higher-order knock-on effects and less directly by the features themselves. Consequently, it becomes inherently more difficult to predict (see also the conclusions of De Leeuw et al. [27], which are consistent with our results). Furthermore, note the relative difference between the RMSE and MAE — the number of missed flights has the highest of all, indicating the presence of outliers. In spite of that, LR convincingly remains the best performing architecture for the response, while the tree-based ensembles are inadequate. Altogether, LR, GP and GB seem to mimic the output parameters of AATOM rather well, despite the fact that RTHA is a complex sociotechnical system. However, there is evidently "no free lunch", as multiple architectures must be considered and carefully optimized per individual parameter to achieve a high accuracy [28]. As there is now evidence that each response can be closely resembled by at least one surrogate, we continue with analyzing the total expenditure of passengers on discretionary activities. A precursory remark is that both case studies solely deploy a response's best performing meta-model for agnostic analysis. The first case study examines the spending behavior of passengers on non-aeronautical activities. In fact, this was the main topic of the analysis by Mekic et al. [6], though we go more in-depth to demonstrate the strengths of synthesizing interpretation techniques applied to surrogate models. The total expenditure represents the amount of money all passengers together spent on activities such as shopping, dining, etc., during the simulated time frame. These events are of course not mandatory to catch a flight and hence not a priority, so passengers will only consider them if they have enough time. Readily, it shows that the expenditure is an ideal starting point to analyze emergence; the indicator is affected by various interdependent phenomena in the airport terminal.



(a) Local sensitivity uncrowded terminal. (b) Local sensitivity crowded terminal. (c) Total-order global sensitivity.

Figure 3: Sensitivity analysis of the total expenditure.

We commence the analysis in Figure 3 by exploring the sensitivity of features. First, two one-at-a-time assessments are performed in Figure 3a and 3b, which depict tornado diagrams of local sensitivities. An uncrowded scenario is compared against a crowded one, both assuming poor airport staffing strategies (check-in and security check strategy 1) and an early call-to-gate (48 minutes before departure). The crowd is controlled by adopting a load factor of about 65% and 85% on all flights, respectively. Poor staffing strategies in combination with an early call-to-gate does not provide the ideal condition for discretionary activities — passengers have less spare time in the terminal. Nevertheless, the baseline values of the tornado diagrams suggest that busier scenarios lead to more spending. This makes sense, as larger crowds are naturally expected to have a higher expenditure. Both diagrams associate the greatest sensitivity to the call-to-gate strategy, though note that it has a negative direction. In other words, the sooner passengers are called to the gate, the less they spend along their journey and vice versa. While this is not surprising, a more striking difference is the influence of certain flights' load factor. They are all harmonious for the uncrowded terminal, but not when it gets busier. For flights 2 and 5 in particular, the total expenditure decreases as more passengers travel on those flights. This is rather counter-intuitive, so we resort to other methods to explain the negative effect and why it depends on the scenario. We conclude the sensitivity analysis by plotting total-order Sobol indices in Figure 3c. They attribute the variance of a response to the features in proportion to their contribution, so total expenditure appears to be most sensitive to the call-to-gate strategy. The global impression thus corresponds to the local impressions, although it is more pronounced. Figure 4 explains how the throughput at security is influenced. We compare an uncrowded terminal against a crowded one; both scenarios presume good staffing strategies and an early call-to- gate, the values of which are shown in the graphs. A trivial conclusion is that high load factors lead to positive contributions to the number of passengers passing through security, and vice versa. Furthermore, note that constantly operating the checkpoint at full capacity — security check strategy 16 — positively impacts the total flow. That is logical, as it delivers the best possible service, thereby maximizing throughput. Notwithstanding, one should remain vigilant about the discrepancy between predictions of LIME and the surrogate. The error is around 30 passengers for both scenarios, which is rather large compared to the size of the feature impacts. This calls for some caution with using LIME as the sole method of interpretation, even if it produces comprehensible insights. On the one hand, LIME appears to inflate the impact of security check strategy 16; it is not even visible on the bee swarm summary plot since its effect is negligible. On the other hand, SHAP never reports a negative impact for the second check-in strategy, despite it being almost negligible as well. Nevertheless, the other features are generally consistent and in line with expectations. We now focus on flights 2 and 6 for the remainder of the analysis. These two are among the most impactful, according to LIME and SHAP, and have an additional interconnection. Indeed, they are assigned to the same check-in counters, allowing us to examine whether there are again knock-on effects as in the previous case study.

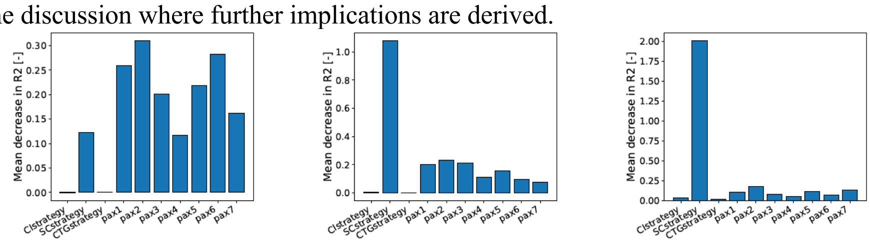


(a) Uncrowded terminal. (b) Crowded terminal.

Figure 4: Local interpretable model-agnostic explanations of the throughput at security. The interpretation is as follows. The vertical axis shows all input parameters and their assumed values, while the corresponding contributions to the response are plotted horizontally. These contributions can be considered as the impact on a prediction relative to the intercept of LIME's approximation. Bars appear red if the effect is negative and green otherwise. To connect the dots, LIME arrives at its local prediction by adding the individual contributions of all features to the intercept, which should then be close to the actual outcome of the investigated surrogate model.

Next, we also analyze feature importance to see exactly which key drivers control the checkpoint's through- put, average waiting time, and the ensuing number of missed flights. The results are shown in Figure 5, respectively. It is immediately noticeable that the graphs of the latter two are similar; both are driven primarily by the staffing strategy at security and to a lesser extent by occupancy rates. The opposite holds for the checkpoint's throughput, although the difference is not as pronounced. One should interpret these results as follows. Under normal circumstances, more passengers lead to more passage through security, which is therefore mainly determined by the load factor on flights. However, if the airport opts for a bad strategy, waiting times may increase considerably. The extent also depends on how busy it is, but personnel strategy is more decisive. That is logical, as it directly dictates the number of lanes to be opened. Ceteris paribus, fewer lanes will always lead to longer waiting times, but not necessarily to a lower throughput. This explains the difference between Figures 8a and 8b. However, there is a risk that the waiting time, which we know is predominately driven by strategy, will continue to rise so passengers are no longer able to reach their gate on time. At that point, the number of missed flights will increase rapidly, especially when it is busy. Queuing time and the number of missed flights thus have the same key drivers.

Finally, the interpretations are again juxtaposed with SHAP as a means of validation. Aside from the previously discussed inconsistencies between LIME and SHAP, the results are actually rather consistent and in line with expectations. However, there is one interesting finding to point out. Security check strategy 4 makes the checkpoint operate at full capacity from an hour onward, so it is presumed to be a solid approach. According to SHAP, throughput is indeed higher and fewer passengers end up missing their flight. Yet, it appears that the strategy also prolongs the expected waiting time at security. This may conflict with what one would initially believe, although in fact it makes sense. By operating with too small a capacity, more passengers arrive than can be handled, causing the queue to grow. If suddenly all lanes are opened, then there is already a considerable queue while passengers are still arriving. Eventually, the queue is eliminated and a smooth passage is possible, although it took some time and increased the average wait. This again confirms that longer waiting times can, but not always, lead to a higher number of missed flights. Ergo, one ought to be careful about implying such causalities. Nonetheless, our previous arguments are in accordance with SHAP, which concludes the second case study. In the next section, we continue with the discussion where further implications are derived.



(a) Throughput at security (b) Average queuing time at security. (c) Number of missed flights.

Figure 5: Visualization of permutation feature importance.

5 Conclusions

The motivation for our research originates from the observation that existing airport terminal operations models: 1) suffer from heavy computational requirements, and 2) reveal their emergent properties only a posteriori. These are typical challenges of agent-based modeling, the principle according to which they are usually built. Therefore, we introduced a two-stage methodology to analyze such systems in a more efficient way. The first stage involves the development of faithful surrogate models, whereafter the second stage applies techniques from the emerging field of explainable artificial intelligence to these abstractions. The novelty of our methodology lies thus in the amalgamation, rather than in the respective research fields themselves. Indeed, we have explored their common ground to take advantage of synergies. A successful application reveals the properties of the focal system, which in the case of a sociotechnical system mainly concerns emergent phenomena. Proof of the methodology's efficacy is provided by conducting two case studies on AATOM; a validated agent-based airport terminal operations model. On the one hand, we looked at the total expenditure on non-compulsory activities, like shopping and dining. It was found that the journey of some passengers may be disturbed in such a way there is an effect on their spending behavior. Knock-on phenomena were observed, with travelers from earlier flights holding up those from later flights at check-in and security. Consequently, less free time is left to engage in discretionary activities. It happens especially when the terminal is busy in combination with poor airport staffing strategies. This is a clear example of emergence, the root causes of which could even be associated to specific strategies and the occupancy on certain flights. On the other hand, we also examined the throughput at security. More passengers means more passage, but there is an evident point where the checkpoint reaches its maximum capacity. As a result, throughput remains constant, while the queue and therefore the waiting time quickly increase. This even goes so far as to put passengers at risk of missing their flight. Again, unequivocal evidence of emergent properties, which are thus clearly preserved in surrogates. The key drivers of the phenomenon could also be traced back, along with the critical settings; it only occurs under certain conditions. Altogether, the case studies demonstrated that the proposed methodology is indeed able to accurately abstract and explain the dynamics of airport terminal operations through surrogate modeling an existing simulation model.

References

1. B. De Leeuw, S. S. Mohammadi Ziabari, and A. Sharpanskykh. Surrogate Modeling of Agent-based Airport Terminal Operations. In *Multi-Agent-Based Simulation XXIII*, Auckland, New Zealand, Mar. 2022. URL https://mabsworkshop.github.io/articles/MABS_2022_paper_9.pdf.
2. J. N. Fuhg, A. Fau, and U. Nackenhorst. State-of-the-Art and Comparative Review of Adaptive Sam- pling Methods for Kriging. *Archives of Computational Methods in Engineering*, 28(4):2689–2747, June 2021. ISSN 1134-3060. doi: 10.1007/s11831-020-09474-6. URL https://link.springer.com/10.1007/ s11831-020-09474-6.
3. P. Pao-Yen Wu and K. Mengersen. A review of models and model usage scenarios for an airport complex sys- tem. *Transportation Research Part A: Policy and Practice*, 47:124–140, Jan. 2013. ISSN 0965-8564. doi: 10. 1016/j.tra.2012.10.015. URL https://linkinghub.elsevier.com/retrieve/pii/S0965856412001541.
4. V. A. Thurmond. The Point of Triangulation. *Journal of Nursing Scholarship*, 33(3):253–258, Sept. 2001. ISSN 1527-6546, 1547-5069. doi: 10.1111/j.1547-5069.2001.00253.x. URL https://onlinelibrary. wiley.com/doi/10.1111/j.1547-5069.2001.00253.x.
5. H. Noble and R. Heale. Triangulation in research, with examples. *Evidence Based Nursing*, 22(3):67–68, July 2019. ISSN 1367-6539, 1468-9618. doi: 10.1136/ebnurs-2019-103145. URL https://ebn.bmj.com/ lookup/doi/10.1136/ebnurs-2019-103145.
6. I. E. Manataki and K. G. Zografos. Development and Demonstration of a Modeling Framework for Air- port Terminal Planning and Performance Evaluation. *Transportation Research Record: Journal of the Transportation Research Board*, 2106(1):66–75, Jan. 2009. ISSN 0361-1981. doi: 10.3141/2106-08. URL http://journals.sagepub.com/doi/10.3141/2106-08.
7. V. Belle and I. Papantonis. Principles and Practice of Explainable Machine Learning. *Frontiers in Big Data*, 4:39, 2021. ISSN 2624-909X. doi: 10.3389/fdata.2021.688969. URL https://www.frontiersin. org/article/10.3389/fdata.2021.688969.
8. A. Fisher, C. Rudin, and F. Dominici. All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *Journal of Machine Learning Research*, 20(177):1–81, 2019. ISSN 1533-7928. URL http://jmlr.org/papers/v20/18-760.html.
9. C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9.
10. S. Janssen, A. Sharpanskykh, and R. Curran. Agent-based modelling and analysis of security and efficiency in airport terminals. *Transportation Research Part C: Emerging Technologies*, 100:142–160, Mar. 2019. ISSN 0968-090X. doi: 10.1016/j.trc.2019.01.012. URL https://linkinghub.elsevier.com/retrieve/ pii/S0968090X1830809X.
11. E. Borgonovo and E. Plischke. Sensitivity analysis: A review of recent advances. *European Journal of Operational Research*, 248(3):869–887, Feb. 2016. ISSN 0377-2217. doi: 10.1016/j.ejor.2015.06.032. URL https://linkinghub.elsevier.com/retrieve/pii/S0377221715005469.
12. D. Chicco, M. J. Warrens, and G. Jurman. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*, 7, July 2021. ISSN 2376-5992. doi: 10.7717/peerj-cs.623. URL https://peerj.com/articles/cs-623.

13. L. Jia, R. Alizadeh, J. Hao, G. Wang, J. K. Allen, and F. Mistree. A rule-based method for automated sur- rogate model selection. *Advanced Engineering Informatics*, 45:101123, Aug. 2020. ISSN 1474-0346. doi: 10. 1016/j.aei.2020.101123. URL https://linkinghub.elsevier.com/retrieve/pii/S1474034620300926.
14. B. Timmins and K. Austin. Heathrow flight cancellations cause queues and 'chaos'. *BBC News*, June 2022. URL https://www.bbc.com/news/business-61857008.
15. IATA. *Airport Development Reference Manual*. Montreal, 9th edition, 2004. ISBN 978-92-9195-086-7.
16. C. Macal and M. North. Tutorial on agent-based modeling and simulation. In *Proceedings of the Winter Simulation Conference, 2005.*, pages 2–15, Dec. 2005. doi: 10.1109/WSC.2005.1574234. ISSN: 1558-4305.
17. F. Hutter, J. Lücke, and L. Schmidt-Thieme. Beyond Manual Tuning of Hyperparameters. *Künstliche Intelligenz*, 29(4):329–337, Nov. 2015. ISSN 0933-1875. doi: 10.1007/s13218-015-0381-0. URL http: //link.springer.com/10.1007/s13218-015-0381-0.
18. L. Magalhães, V. Reis, and R. Macário. A new methodological framework for evaluating flexible op- tions at airport passenger terminals. *Case Studies on Transport Policy*, 8(1):76–84, Mar. 2020. ISSN 2213-624X. doi: 10.1016/j.cstp.2018.03.003. URL https://linkinghub.elsevier.com/retrieve/pii/ S2213624X18300749.
19. S. M. Lundberg and S.-I. Lee. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, volume 30, Long Beach, CA, USA, 2017. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper/2017/hash/ 8a20a8621978632d76c43dfd28b67767-Abstract.html.
20. C. Q. Lam. *Sequential Adaptive Designs In Computer Experiments For Response Surface Model Fit*. PhD thesis, Ohio State University, 2008. URL http://rave.ohiolink.edu/etdc/view?acc_num= osu1211911211.
21. F. Lamperti, A. Roventini, and A. Sani. Agent-based model calibration using machine learning surrogates. *Journal of Economic Dynamics and Control*, 90:366–389, May 2018. ISSN 0165-1889. doi: 10.1016/j.jedc. 2018.03.011. URL https://linkinghub.elsevier.com/retrieve/pii/S0165188918301088.
22. S. Janssen, A. Sharpanskykh, and R. Curran. AbSRiM: An Agent-Based Security Risk Management Approach for Airport Operations. *Risk Analysis*, 39(7):1582–1596, 2019. ISSN 1539-6924. doi: 10.1111/ risa.13278. URL http://onlinelibrary.wiley.com/doi/abs/10.1111/risa.13278.
23. S. Janssen, A.-N. Blok, and A. Knol. *AATOM - An Agent-based Airport Terminal Operations Model*. Delft University of Technology, Apr. 2018. URL https://research.tudelft.nl/en/publications/ aatom-an-agent-based-airport-terminal-operations-model.
24. P. Westermann and R. Evins. Surrogate modelling for sustainable building design A review. *Energy and Buildings*, 198:170–186, Sept. 2019. ISSN 0378-7788. doi: 10.1016/j.enbuild.2019.05.057. URL https: //www.sciencedirect.com/science/article/pii/S0378778819302877.
25. K. C. James and M. Bhasi. Development of model categories for performance improvement studies related to airport terminal operations. *Journal of Simulation*, 4(2):98–108, June 2010. ISSN 1747-7778. doi: 10.1057/jos.2009.27. URL https://www.tandfonline.com/doi/full/10.1057/jos.2009.27.
26. S. Janssen, A. Sharpanskykh, R. Curran, and K. Langendoen. Using causal discovery to analyze emer- gence in agent-based models. *Simulation Modelling Practice and Theory*, 96:101940, Nov. 2019. ISSN 1569190X. doi: 10.1016/j.simpat.2019.101940. URL https://linkinghub.elsevier.com/retrieve/pii/ S1569190X19300735.
27. D. Curcio, F. Longo, G. Mirabelli, and E. Pappoff. Passengers Flow Analysis And Security Issues In Airport Terminals Using Modeling & Simulation. In *ECMS 2007*, pages 374–379. ECMS, June 2007. ISBN 978-0-9553018-2-7. doi: 10.7148/2007-0374. URL http://www.scs-europe.net/dlib/2007/2007-0374.htm.
28. B. Williams and S. Cremaschi. Selection of surrogate modeling techniques for surface approximation and surrogate-based optimization. *Chemical Engineering Research and Design*, 170:76–89, June 2021. ISSN 0263-8762. doi: 10.1016/j.cherd.2021.03.028. URL https://www.sciencedirect.com/science/article/ pii/S0263876221001465.
29. Ziabari, S. S. M., Sanders, G., Mekic, A., & Sharpanskykh, A. (2021). Demo paper: a tool for analyzing COVID-19-related measurements using agent-based support simulator for airport terminal operations. In Advances in Practical Applications of Agents, Multi-Agent Systems, and Social Good. The PAAMS Collection: 19th International Conference, PAAMS 2021, Salamanca, Spain, October 6–8, 2021, Proceedings 19 (pp. 359-362). Springer International Publishing.