Analysis of urban air mobility’s transport performance in São Paulo Metropolitan Region using MATSim

Sérgio Fonseca dos Santos
sergiofsantos@tecnico.ulisboa.pt

Instituto Superior Técnico, Universidade de Lisboa, Portugal
November 2021

Abstract

Due to rapid population growth, it is estimated that the level of congestion will increase further in cities. With this, there is a need to integrate new transport modes, as a complement to the existing ones, in order to reduce travel times. Recent advances in aeronautical technology have allowed the development of a new transport mode, Urban Air Mobility (UAM). This, when implemented, will allow people to move from one point to another in cities using electric or hybrid vehicles with Vertical Take-Off and Landing (VTOL) capabilities. While these vehicles have been widely studied, for example, in terms of engines, aerodynamic and aeroacoustic performance, there is little research regarding the impact that the introduction of these vehicles will cause and what changes will occur in cities. It is therefore essential to predict the behavior of cities when introducing this new transport mode, so that the best measures can be adopted and, thus, this new service can be enhanced. That said, this work has two objectives: to predict what impact the introduction of this new transport mode will have in the city of São Paulo, Brazil, through an agent-based Mobility Simulator (MATSim) in conjunction with its UAM and extension and to create a simulation metamodel based on Gaussian Processes and active learning, which allows to predict the result of the simulations, instead of them being carried out, thus saving time and computational power. Regarding the simulation results, it is verified that in the simulated base scenario there is great air congestion. However, the introduction of this new mode of transport can serve to alleviate congestion on the roads and is therefore a good complement to the transport modes already available. In relation to the metamodel, it is concluded that it is capable of predicting, with some accuracy, the results of simulations carried out. However, it is necessary to improve the strategy in order to bring the metamodel closer to the simulator used.

Keywords: UAM, MATSim, Machine Learning, Gaussian Processes, Metamodel, São Paulo.

1. Introduction

Due to rapid population growth it is estimated that, by 2050, the population will reach 9.7 billion inhabitants [1], an increase of about 2 billion, compared to the current number (equivalent to an increase of 25%). With this predicted increase, the level of congestion on the urban centers is expected to increase further. That said, the need arose to integrate new transport mode, in addition to the existing ones, in order to reduce travel times. In 2018, the average car journey time in São Paulo, Brazil, was 30 minutes [2].

Recent advances in aeronautical technologies, have enabled the development of a new transport mode, UAM. This will allow people to move from point to point in cities using electric or hybrid vehicles with VTOL capabilities. While these vehicles have been extensively studied, regarding their main areas, like engines, aeroacoustics and aerodynamics [3], there is little research on its introduction into cities and what influence it will have on them.

In view of this, there is a need to simulate the behavior that the city of São Paulo will have, in order to predict the number of people who can benefit from this new transport mode and to adopt the best policies for its introduction.

Therefore, this work has two main objectives:

- to instantiate an agent-based transport model of on-demand UAM services in São Paulo Metropolitan Region under the MATSim traffic simulator using the UAM extension. The analysis consists of studying the impact of this new urban transport paradigm on the overall transportation system performance, when compared with traditional transportation means. The study will try different selected UAM system properties, such as the number of stations and their locations, total process time, fleet size, vertical and cruising
speed, vehicle capacity and costs. To reach this goal, it is necessary to create a synthetic population, a network, a model that allows the agents to do their choices and it is necessary to define the UAM service.

- create a metamodel based on Gaussian Processes with active learning, to predict the simulation outputs instead of doing the simulation itself, which can be a time-consuming process.

2. Background and theoretical overview

2.1. UAM service – The future

UAM is one of the great current trends, with regard to urban mobility. With large cities reaching an extreme level of congestion, this new mode of transport has attracted the attention of the general public, as a way to substantially reduce the time spent in traffic jams. Thus, there are more than 100 vehicles in development worldwide, some still in the design phase, others already undergoing tests, according to [4], that will allow this new technology to advance. These vehicles can be divided according their morphology [5] in rotary-wing cruise or fixed-wing cruise. The latter is divided into tilt-wing/prop, lift+cruise and tailsitter. Although the new service can be used in many ways, there are 4 main ways that are usually foreseen, according to [6]: passenger transport, medical service, logistics and surveillance. In order for these vehicles to operate, infrastructures on land are needed, since these vehicles need specific infrastructures. As with other transports, this transport mode will have a preflight and a postflight time, which represent the time spent by people at the stations. There will also be a boarding and deboarding process. Due to advances in technology and since there is a great investment in electrification, a turn around time may be necessary, that is, a period in which the vehicles are inoperative to charge batteries. In addition to this, new regulations regarding airspace will be needed, which is not so trivial. This regulations will have as main objectives to define which airspace to use, maximum speeds, minimum distance between vehicles, maximum number of vehicles that can circulate, per hour, in a given area, as well as acoustic and environmental issues. In conclusion, although this service still needs some advances, mainly with regard to the level of regulations, it is expected that this service will start operating within the next 5 years.

2.2. Transport modelling and simulation

There are several simulation models, however, these can be divided essentially into: trip-based models, activity-based models and agent-based models. The last one, which is used in this work, have several applications, and can be used to model complex adaptive systems. These complex adaptive systems can be transit networks. In this case, the aim is to simulate the behavior of vehicles, the transport network and agents (persons), as well as their ability to adapt to the environment by maximizing the utility of each agent’s plans, making sure that they choose the best route. MATSim and AnyLogic are an example of softwares that uses this methodology.

2.2.1 Traffic simulations classification

The traffic simulations can be divided in four types: macroscopic, mesoscopic, microscopic and nanoscopic [7]. Macroscopic simulation models consider each vehicle in the same way and as a group and, therefore, are not good when the objective is to obtain simulations with great detail. On the contrary, microscopic simulators simulate individual entities, such as vehicles and drivers, with a high level of detail. These simulators, since they present a higher level of detail, are more realistic than the macroscopic simulators. The mesoscopic simulators aim to fill the gap between macro and micro simulators. In this case, the vehicles can be grouped and are treated as one entity. Finally, the nanoscopic model is a relatively new trend of traffic simulation and are mostly used in autonomous driving. As the work was developed with the mesoscopic MATSim framework, its operation will be described in detail.

2.3. MATSim framework

The Multi-agent Transport Simulation (MATSim) [8] is an open-source and java-based tool, developed by ETH Zürich and TU Berlin and which simulates the daily life of individuals. In order to understand how MATSim works, it is necessary to understand some concepts. First, the scenario that is simulated contains a synthetic population, consisting of agents (travel demand), a network and the location of the facilities. With the creation of the synthetic population, plans are assigned to agents, and plans are, as [9] describes, a chain of activities (e.g. home–education–shop–home), which include their locations and end times. According to [10], MATSim “utilizes an iterative, co-evolutionary learning approach” in which, the main goal of the agents is to optimise their daily plan of activities. The execution loop of MATSim, represented in Figure 1, is based on three major events:

- (1) Mobility Simulation or mobsim, which executes the agents’ plans. The available trans-

---

1https://www.easa.europa.eu/domains/urban-air-mobility-uam

2https://www.matsim.org

3https://www.anylogic.com
port modes during the simulation, will allow the agents to perform their activities;

- **(2) Scoring**, a score is assigned to the plans executed by agents during the mobsim, through utility functions. Working, for example, is an activity that gives a positive score (since it generates a monetary income), while commuting to work gives a negative score;

- **(3) Re-planning** allows agents to change their plans, via innovative strategies, in order to improve their score. In this event, not all the agents are able to re-planning their plans and the ones that are able to change it, can modify the mode, departure time, route or trip of the previous plan.

The execution of all plans (mobsim), its scoring and re-planning is called an iteration. This process is repeated until the system reaches a Nash equilibrium, that is, as [9] describes, the further development of agents’ plan scores is “sufficiently relaxed”.

### 2.3.1 UAM extension

This extension arose from a collaboration between Bauhaus Luftfahrt e.V., the Eidgenössische Technische Hochschule Zürich, and the Technical University of Munich and has been published as an open-source project\(^4\). Under the assumption of on-demand operational models for UAM, the transport modeling of VTOL vehicles mirrors that of autonomous ground-based taxis, according to [11]. Thus, the UAM extension for MATSim is also based on the Dynamic Vehicle Routing Problem (DVRP) MATSim contribution by [12]. This extension allows to define the following aspects, inherent to this transport mode:

- **Stations** are the necessary infrastructure for the take-off and landing of vehicles;

- **Flight paths** link stations directly (euclidean distance) or via waypoints;

- **Vehicles** circulate through flight paths, from station to station. Each vehicle must be capable of vertical take-off and landing and a cruise phase, with each phase having different speeds;

- **On-demand operation** which combines UAM stations, flight paths and vehicles to enable this mode of transportation.

2.4. UAM studies

In relation to the different areas that need to be studied, [6] provides an overview of the studies that have been carried out regarding the vehicles, [13] presents some research gaps that must be addressed before this new service is introduced into society. In turn, [14] evaluates relevant prerequisites for Personal Air Transport System introduction into the urban transport modelling environment. Regarding the simulations that have been carried out to predict the impacts of the introduction of UAM and identify measures that need to be implemented, there are already several scenarios that have been simulated. [15] explores the addressable market for UAM as a multi-modal alternative in a community in the San Francisco Bay Area, using MATLAB. [16] and [17] present a methodology for simulation and demand estimation for personal aerial vehicles in Zurich, Switzerland using MATSim with discrete mode choice models. [18] uses the MATSim UAM extension to simulate a test scenario in Sioux Falls, USA. [19] define the required models and methods to analyze and quantify the potential demand for urban air mobility and possible impacts were defined and applied to the Munich Metropolitan region. [20] compares three pre-existent, and calibrated agent-based transport scenarios from three different cities, Munich, Paris and San Francisco, in terms of potential travel time savings, using the UAM-extension for MATSim. Globally, several studies have been carried out, with regard to the most diverse areas that need to be studied, in order to be able to introduce UAM in the near future. However, there is still little progress regarding the laws that will govern this new transport mode and one of the possible facts for this is the difficult forecast of what will happen when the UAM service is introduced.

2.5. Simulation metamodels via machine learning

Simulation metamodels are very often used to approximate the underlying simulation function. According to [21], the use of simulation metamodels has four goals: real-world problem understanding, simulation output prediction, optimization, and verification/validation. In more complex scenarios and where simulation data can be difficult to obtain, according to [22], the use of simulation metamodels in conjunction with active learning proves to be an important tool. Among many significant machine learning tools, Gaussian Processes (GP) framework, by [23], for its Bayesian and non-linear modeling properties, allows to develop methodologies that combine active learning and metamodeling.

----

\(^4\)https://github.com/BauhausLuftfahrt/MATSim-UAM
strategies.

2.5.1 Gaussian processes

As described by [23], a GP is a stochastic process that can be defined by a mean and a covariance function (also known as kernel function), \( m_f(x) \) and \( k_f(x, x') \), respectively, being \( x \) and \( x' \) two input data points. Therefore, it is denoted as \( GP(m_f(x), k_f(x, x')) \), where \( m_f(x) = \mathbb{E}[f(x)] \) and \( k_f(x, x') = \mathbb{E}[(f(x) - m_f(x))(f(x') - m_f(x'))] \). One of the applications of GP is to perform regression via supervised learning, which is called Gaussian Process Regression. In this context, the GP framework places a prior over functions with continuous domain, i.e., \( y = f(x) + \epsilon \), where \( \epsilon \sim N(0, \sigma^2) \) and \( f(x) \sim GP(m_f(x), k_f(x, x')) \). Regarding the kernel function, it measures the “similarities” between two different predictions, it means that a higher kernel function value implies that the two predictions are more similar. The kernel and the mean function have some free parameters – hyper-parameters of the GP – which can be optimized by marginal likelihood maximization subjected to the training data. After these parameters are obtained, the conditional distribution for any unlabeled test point \( x_* \) is given by \( f_*|X, y, x_* \sim N^T(k_{f*}[K_y]^{-1}y, (k_{f*} - K_f^T[K_y]^{-1}k_f)) \) where \( k_{f*} = k_f(X, x_*) \), \( k_{f*} = k_f(x_*, x_*) \), \( [K_y] \) is the covariance matrix, \( X \) the design matrix and \( y \) is the vector of the output values.

2.5.2 Active learning

Active learning allows any algorithm to actively choose the training data points during the learning process. This is extremely useful in scenarios where labeled data is expensive to obtain. Therefore, this paradigm choose the most informative points from which it learns. An arbitrary active learning strategy includes five elements, \( (L, U, M, O, Q) \). The first element, \( L \) is the labeled training data set and \( U \) is the set of unlabeled data points. Usually, the number of unlabeled data points is much higher than the labeled set. \( M \) is the machine learning model, which can be a classification or a regression model, what makes the values in \( L \) being discrete or continuous, respectively. \( O \) is the oracle, which function is to provide labeled values. The last, \( Q \), is the query function which contains the strategies and criteria for finding and selecting the most informative instances of \( U \) to be added to \( L \). Related with the query function, this learning paradigm can be divided into pool-based or stream-based. In the pool-based, each individual data point is presented serially or in successive blocks, whereas in the stream-based all the unlabeled instances are available for querying. Since this is an iterative process, there must be a stopping rule. In general, this criteria should take into account the performance of the model, is generalization capacity and the associated costs of acquiring new labeled data.

2.5.3 Related works

The application of simulation metamodels based on GP is still rare, however, there are some works that have been developed. These works can be divided into traffic prediction or optimization of networks, according to [24], [25] adopted a metamodel-based technique and conducted a sensitivity analysis using the mesoscopic traffic simulator AIMSUN. [26] also applied the metamodel for mesoscopic simulation to solve the bi-level Mixed Network Design Problem (MDPD) to enable dynamic traffic assignment (DTA) and the simulation-based optimization solution. [27] utilizing a simulation-based dynamic traffic assignment model, proposed a Bayesian stochastic Kriging metamodel to optimize integrated planning and operational ATM strategies for corridors using the real scenario of I-270 and MD355 in the state of Maryland, USA. [22] propose an active learning algorithm based on the GP framework. The methodology makes use of the most informative simulation data points in batches, according to both their predictive variances and to the relative distance between them. [28] in order to approximate the response surface for the transportation simulation input–output mapping and search for the optimal toll charges in a transportation network, adopts surrogate-based optimization approaches.

3. UAM simulation in MATSim - São Paulo case study

3.1. Methodology

To carry out the simulations, the eqasim framework which is based on the MATSim framework with added components for simulation of discrete choice models was used together with the UAM extension, available on GitHub\(^5\). Therefore, it was necessary to create a synthetic population together with the network, to define the discrete mode choice model, that is, the equations that govern the model and the UAM service, which includes the designation of vehicles and stations.

3.1.1 Population and network

The creation of the synthetic population of São Paulo Metropolitan Region was made based on open data and using the eqasim’s repository available on GitHub\(^6\). Only about 1% of the total population residing in São Paulo was generated, in order to reduce the time of the simulations. The data required for its creation are: census data, São Paulo

\(^5\)https://github.com/eqasim-org/uam  
\(^6\)https://github.com/eqasim-org/sao_paulo
household travel survey, OpenStreetMap data, the place of educational facilities, census zoning system, road network (OpenStreetMap) and public transit schedule (GTFS). The generated network does not yet have the UAM service. In Figure 2, it is possible to see an overview of the created network. This network represents well the real network, since it is obtained using the Open Street Map.

Figure 2: Network of the city of São Paulo without UAM.

### 3.1.2 Mode choice model

In order to use the discrete mode choice model, it was necessary to define the equations that will allow, during the simulation, to obtain a probability for each option, in order to allow the agent to make the best decision. The equations are already defined in the calibrated São Paulo scenario, in EqaSim’s GitHub repository, so they have not been changed. That said, it was only necessary to define the equation of the UAM service. This equation is represented below.

\[ u_{UAM} = \alpha_{UAM} + \beta_{travelTime,UAM} \times travelTime,UAM + \beta_{cost} \times c_{UAM} \]  

The options available are UAM, car, PT, taxi and walk. Bike is not considered because its modal share is zero in the real case, having in account the data of 2020. There are several parameters that must be defined, with \( \alpha \) being a constant specific to the transport modes, \( \beta \) a linear coefficient that translates all choice dimensions into a generalized utility, \( c \) represents the cost of the trip, \( x \) the distance of each trip and \( t \) the time. Having the equations, it is necessary to assign values to the various variables. The parameters of car, public transport, taxi and walk are based on the calibrated scenario. Regarding the definition of the UAM service, the parameter values were based on the values of public transport. With regard to the walk mode, there is also a penalty of -100 in the utility, if agents travel more than 20km on foot.

### 3.1.3 UAM service

Regarding the definition of the UAM Service, this was made based on the repository available on GitHub MATSim-UAM, which allows the creation of a file containing information about vehicles (name, vehicles per station, start and end time, range, cruise and vertical speed, boarding and deboarding time and turn around time) and stations (name, VTOL distance, preflight time, postflight time, default waiting time, ground and flight access capacity and free-speed), as well as updating the network, so that it includes the UAM service. It was then necessary to define the locations for placing the stations \((x, y, z)\) coordinates, as well as the parameters listed below, which concern the stations and vehicles. In order to assess the influence of some parameters, a base scenario is defined and these parameters are changed one by one, in relation to the base scenario. The parameters defined for the base scenario and their variations are: Cruising speed \((\text{km/h})\) – 50, (150), 250, 350; Vertical speed \((\text{m/s})\) – 5, (10), 20; Ground Process Time \((\text{min})\) – 0.5, (3), 5, 15; Vehicle Capacity – (1), 2, 6; Fleet Size (uniform distributed) – 100, (1000), 2000; Number of stations – 10, (25); Fare per km \((\text{BRL})\) – 0, (2), 5; Fare per min \((\text{BRL})\) – 0, (0.5), 2; Base fare \((\text{BRL})\) – 0, (3), 10, with the value of the base scenario being between parenthesis. It was also necessary to introduce the stations in the network and, for this, an analysis was carried out, based on the origin-destination pairs of all trips, of all agents.

### 3.2. Synthetic population validation

In order to prove that the population used in the simulations is effectively representative of the city of São Paulo and that it will reflect their choices, a comparative analysis was made to the data obtained with the real values, taken from the results of the 2010 census. Therefore, dividing the population by gender and age, we can conclude that the synthetic population represents well the true population. Comparing the percentage of the population residing in São Paulo that is employed, unemployed or is a student with the household travel survey, it is possible to verify in Figure 3 that there is a good matching between these categories.

### 3.3. UAM MATSim results

The results are divided into two parts. The first analyzes the base scenario and makes a comparison with the scenario without UAM. The second part analyzes the results obtained, taking into account the variation of the different input parameters.

---

7https://github.com/eqasim-org/eqasim-java
8https://github.com/BauhausLuftfahrt/MATSim-UAM
9https://www.ibge.gov.br/censo2010/apps/sinopse/
3.3.1 Overall evaluation

First, in order to verify the impact that the introduction of UAM has on the city, a simulation without UAM was carried out, in order to verify if it represents well the reality. Analyzing the mode shares of the simulation without UAM, it can be concluded that they represent the reality relatively well, comparing to the real data\textsuperscript{10}. However, there is a difference with regard to the use of the car, since in 2020 the real value was 36\% and in the simulation is, approximately, 21\%.

Regarding the result of the base scenario, 41128 UAM legs were performed and they represent 8.28\% of the total number of legs. The mode shares of each transport mode (including walk) were analyzed, and, organized in a decreasing manner, the result is as follows: walk (30.34\%), public transport (28.71\%), UAM (18.94\%), car (12.49\%), car passenger (8.7\%) and taxi (0.82\%). According to this results, it is possible to see that the UAM introduction will have a negative impact on all transport modes and this impact will have a greater effect on the reduction of the car’s mode share. One of the reasons that may justify the drop in the use of public transport is the fact that UAM stations are not placed in strategic places, where a change from train to UAM can be made, for example. These strategic places would be next to bus and train terminals. In this way, a better complement between all public transport could be achieved. Posteriorly, the average time of all journeys was calculated, which is approximately 28 minutes. In order to check whether this was a reasonable value for the trips that were being made, the average time of all trips was calculated assuming that all UAM vehicles would travel at the maximum speeds allowed, that is, a cruising speed of 150km/h and a vertical speed of 10m/s. The final result shows that the average time of all trips at maximum speed would be about 5 minutes, which shows that 23 minutes are being spent in traffic jams. Therefore, we can conclude that with the parameters defined for this scenario, there is strong UAM congestion, which can be due to several factors, including the unrealistic value of the ground and flight access capacity of 9999 persons per hour, which hardly puts a limit on the number of people at stations and the number of vehicles per link and the low preflight/postflight time. The average distance covered was 7.4km and, taking into account the average time of all trips, the average speed was 20km/h, a value far below which the vehicle can travel. In the base scenario, the minimum distance traveled in a trip is in the range of 1.5km-2.5km meters and the maximum distance in the range of 27.5-28.5km. However, most trips are in the range of 4.5km-5.5km meters.

3.3.2 Sensitivity to main UAM parameters

- Cruising speed variation: according to Figure 4 an increase in the cruising speed translates into an increase in the number of UAM legs and, consequently, an increase in congestion. On the other hand, a reduction of this parameter results in a reduction in the number of UAM legs and also in the congestion time. Due to a lower speed, it results in an increase in the average travel time.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure4.png}
\caption{Variation of UAM legs varying the cruising speed.}
\end{figure}

- Vertical speed variation: an increase in the vertical speed translates into an increase in the number of UAM legs and, consequently, an increase in congestion. By reducing this parameter, there is a slight reduction in the number of UAM legs.

- Ground process time variation: in this case, the only variation that occurred was the increase in congestion, either with the decrease or increase in the value of this parameter.

- Vehicle capacity variation: there were no significant changes with the change in vehicle capacity, due to the fact that the vehicles, in
the simulation, departure as soon as they have an agent, rather than waiting for more.

- **Fleet size variation**: according to Figure 5 an increase in fleet size translates into an increase in the number of UAM legs, as well as in the average travel time and the level of congestion. On the other hand, a reduction also causes a reduction in the number of UAM legs and in the level of congestion.

![Figure 5: Variation of UAM legs varying the fleet size.](image)

- **Number of stations variation**: the reduction in the number of stations translated into a drastic reduction in the number of UAM legs. However, with the same number of vehicles circulating, the average travel time and the level of congestion increased dramatically.

- **Fare per km, per minute and base fare variation**: a change in fares did not cause a major change in the results, due to the fact that those who use the UAM service in this condition are willing to pay more to use it, as it is still worth it. However, with a reduction in fare per minute, an increase in the average travel time can be verified.

- **Vehicles analysis**: analyzing the results obtained, it can be concluded that fully electric vehicles seem to be a good choice for the city of São Paulo, assuming these conditions. This choice is related to the fact that the trips are relatively short, on average, with the longest trip being around 28km.

4. Simulation metamodel methodology applied on UAM simulation

4.1. Methodology

This work presents an active learning metamodeling methodology based on the work developed by [29]. Therefore, a pool-based active learning strategy is adopted and it is presented in Figure 6. The algorithm was developed using the high-level programming language python and the scikit learn library.

![Figure 6: Active learning metamodeling methodology (adapted from [29]).](image)

This methodology can be divided into 3 main blocks: the simulation metamodeling, the active learning strategy and the policy analysis. The stopping criterion, in our case, is the main difference, comparing to [29]. Instead of using the total predictive variance, we use the mean variance across the unlabeled input simulation region. Therefore, the stopping criterion is the Current Mean Variance (CMV) divided by the Initial Mean Variance (IMV) and this should be bigger or equal to $1 - \alpha$. In this case, $L$ is the set of inputs-outputs made through simulations, $M$ is the GP, $U$ is the set of inputs that the GP will try to predict the output. Based on that, the $Q$ is the query function and is based on the analysis of the mean variance. The $O$, is the simulator, which will give the real output of the unlabeled input which have the greatest variance.

4.2. Corsica case study

Due to the long time that simulations of the São Paulo scenario took to be completed, and due to the high amount of results needed to carry out the metamodel, it became impracticable to apply this strategy to the case studied above, using this strategy. Therefore, the scenario of the island of Corsica, available in the MATSim-UAM repository, was chosen. 500 different simulations were performed, however, the available scenario only features 3 UAM stations, so it was necessary to add more stations. These were added randomly, as the objective here is not to assess the impact of the introduction of UAM on the island of Corsica, but rather to assess the usefulness of creating a metamodel for predicting the output of simulations. To carry out these simulations, a javascript was created that changes the input parameters, in a given range, between each simulation, in order to have 500 different simulations. Despite having used a scenario with 10 stations, these were assumed to be uniform, that is, they present the same values in each simulation, in order to reduce the number of inputs of the metamodel. However, this assumption is not realistic, as
each station will have its own characteristics. After the 500 simulations had been carried out, another 35 different simulations were carried out. Unlike the 500 simulations where all parameters could be chosen within the maximum range of values, these 35 simulations focused on reducing the range of each parameter in order to obtain results simulations in a smaller input space. After all the simulations performed, it was necessary to implement the metamodel based on active learning. In order to the model be trained and be able to make predictions of the unlabeled inputs, the 500 simulations were divided randomly, with 67% being used to train the metamodel \((x_{train} \text{ and } y_{train})\) and the remaining 33% being used to enable the metamodel to make predictions \((x_{test})\). \(x_{train}\) are the input values of the simulations, already presented, and \(y_{train}\) contains the output values corresponding to the \(x_{train}\), being, in this case, the number of UAM legs. The remaining 35 simulations were added separately, in order to see if, reducing the input space, it was possible to improve the metamodel. It is important to note that, in this case, when the metamodel asks for the results of the top 3 highest predictive variance points in \(U\), these values, although not being used, are already available and stored, due to the fact that the 500 simulations have already been carried out and they correspond to the outputs of the \(y_{test}\). Therefore, the algorithm just finds the true values and add them to \(y_{train}\).

### 4.3. Simulation metamodel Results

The results obtained can be divided into two parts. The first part is the one in which the metamodel is only based on the 500 simulations, while the second part takes into account the 500 simulations plus the 35 simulations carried out in a more restricted input space. Figure 7 presents the mean variance variation in function of the number of iterations.

![Figure 7: Comparison of the mean variation as a function of iterations between the metamodel with 500 simulations and the metamodel with 535 simulations.](image)

Regarding the results, the most important remark is that, until iteration number 47, the average variance decreased compared to the first iteration. The metamodel was stopped at iteration 47, as if it continued, it would have no more results to predict, since at the end of each iteration, 3 results are added to the training set. It should be noted that, due to this, the stopping criteria did not have time to be satisfied. Analyzing the figure, it can be concluded that, although slightly, the scenario with only 500 simulations at the beginning, due to having fewer observations, presents a higher average variance however, at the end of the 47 iterations, it managed to reduce more in percentage than the scenario with 535 simulations. In order to evaluate the obtained metamodel with the 500 simulations, it was used to predict the results of the 35 simulations performed later. Analysing the predictions with the true values, we can conclude that the metamodel is still not perfect, especially when the number of UAM legs is reduced. However, for high UAM legs values, the metamodel presents a good estimate. In conclusion, due to the reduced test group, it was not possible to reduce the desired mean variance value and, thus, complete the metamodel, in order to have good predictions. Therefore, according to the results, it is important to have more simulations to add to the training group, mainly simulations with low number of UAM legs in the result.

### 5. Conclusions

#### 5.1. Achievements

The results obtained are divided according to the two objectives. The creation of the synthetic population and the network and the UAM service definition were properly done. This allowed to run several simulations in the MATSim framework, with the UAM extension and to evaluate the impact that the introduction of UAM will have in the city of São Paulo, which is the first objective of the thesis. First, regarding the simulation model, it is possible to conclude that the introduction of this new transport mode can be a great driver for the reduction of congestion on the roads and, thus, reduce greenhouse gas emissions, contributing to saving the planet, bearing in mind that UAM vehicles will be electric. Furthermore, based on the results obtained and taking into account that the synthetic population and the network generated are very close to the reality, it can be concluded that the introduction of the UAM service will be well accepted by society. Regarding the results of the simulations of the base scenario, the parameters defined cause a high level of air congestion. However, analyzing the variation of the input parameters, it is possible to conclude that the reduction in the number of stations and the number of UAM vehicles has a very negative effect on the number of trips made by the UAM service. Therefore, having a high number of stations well distributed throughout the city is essential. On the other hand, the increase in cruising
speed and vertical speed have a positive influence. However, there is no big advantage in increasing the cruising speed too much beyond 150km/h, in this case. Regarding the fares and the ground process time, due to the high congestion, no major conclusions can be drawn. With regard to vehicle capacity, it is not possible to determine anything, as the vehicle does not wait for more agents and leaves with just one. Using the updated scenario of the Corsica island with the 10 stations, it was possible to create the metamodel based on Gaussian Processes with active learning, however it needs some improvements. Although the metamodel did not achieve the desired results, the adopted strategy allows to reduce the average variance over the iterations of the metamodel. Furthermore, it is possible to conclude that the metamodel created allows to predict with some quality the results of the simulations, when the number of UAM legs is high. On the contrary, when the simulations results are a reduced value, the metamodel has some difficulty in being able to predict, due to the reduced number of training cases with a reduced output value.

5.2. Future work
In the future, regarding the simulation model, it will be important to adapt the base scenario, in order to adjust it even more to reality. For this, it will be important to obtain information from residents in São Paulo, and adapt the parameters accordingly. It will also be interesting to know which zones will be prohibited for vehicles to circulate and, thus, adjust the network and to understand where the best places for the placement of stations will be. It will also be interesting to allow vehicles to wait at stations, in order to be able to carry more than one passenger on their trips and thus reduce air congestion. Another important improvement would be the individual characterization of each station, since they don’t have all the same parameters. With regard to the metamodel, it will be important to adapt the strategy so that the most informative points that are selected are distanced from each other, a certain value, in order to better explore the input space and speed up the process. In relation to the Corsica scenario, it is necessary to increase the test group so that the metamodel can progress and then be able to improve the predictions in relation to simulations that have a lower number of UAM legs. It will also be necessary to create new strategies, if one intends to apply the metamodel to the São Paulo scenario, as it is impossible to carry out so many simulations in this scenario, due to the long time it takes for each simulation to run.

Acknowledgements
I would like to thank my supervisors, João Barreto and Zafeiris Kokkinogenis, for their support throughout this journey. It was a pleasure working with you. Special thanks to Instituto Superior Técnico and CEiiA for the opportunity given to me to learn from the best.

References


