INSTITUTO TECNOLÓGICO DE AERONÁUTICA

Verônica Rodrighero

MODELING OF A VTOL AIRLINE FOR URBAN TRANSPORT IN THE CITY OF SÃO PAULO

Final Paper
2025

Course of Aerospace Engineering

CDU 629.78:656.7

Verônica Rodrighero

MODELING OF A VTOL AIRLINE FOR URBAN TRANSPORT IN THE CITY OF SÃO PAULO

Advisor

Maj. Av. Lucas Oliveira BARBACOVI (ITA)

Co-advisor

Prof. Dr. Christopher Shneider Cerqueira (ITA)

AEROSPACE ENGINEERING

São José dos Campos
Instituto Tecnológico de Aeronáutica

2025

Cataloging-in Publication Data
Documentation and Information Division

Rodrighero, Verônica
Modeling of a VTOL Airline for Urban Transport in the City of São Paulo / Verônica Rodrighero. São José dos Campos, 2025.
65f.

Final paper (Undergraduation study) – Course of Aerospace Engineering– Instituto Tecnológico de Aeronáutica, 2025. Advisor: Maj. Av. Lucas Oliveira BARBACOVI. Co-advisor: Prof. Dr. Christopher Shneider Cerqueira.

1. Urban Air Mobility. 2. EVTOL. 3. Operational Modeling. I. Instituto Tecnológico de Aeronáutica. II. Title.

BIBLIOGRAPHIC REFERENCE

RODRIGHERO, Verônica. **Modeling of a VTOL Airline for Urban Transport in the City of São Paulo**. 2025. 65f. Final paper (Undergraduation study) – Instituto Tecnológico de Aeronáutica, São José dos Campos.

CESSION OF RIGHTS

Verônica Rodrighero
Rua H8A, 108
12.228-460 – São José dos Campos–SP

MODELING OF A VTOL AIRLINE FOR URBAN TRANSPORT IN THE CITY OF SÃO PAULO

This publication was accepted like Final Work of Undergraduation Study

---

Verônica Rodrighero
Author

Lucas Oliveira BARBACOVI (ITA)
Advisor

Christopher Shneider Cerqueira (ITA)
Co-advisor

Prof^{a}. Dr^{a}. Maisa de Oliveira Terra
Course Coordinator of Aerospace Engineering

São José dos Campos: November 7th, 2025.

Acknowledgments

To God, for allowing the universe to take care of every event that brought me here.

To my mother, Jô, for giving me the best life she could. Thank you for always prioritizing my studies and my happiness. Without you, none of this would have become reality.

To my father, Ricardo, for awakening my curiosity about the “whys” of science with questions and explanations in my childhood. Thank you for believing in me and pursuing this dream with me.

To all my family who, humble and simple as they are, were my emotional support and sincere cheering squad.

To Laura, for being my companion on this journey. Thank you for standing by my side, faithfully, through the worst and best moments of these years.

To Gustavo who, even if recent, brought joy and rest. Thank you for helping me complete this work and for taking such good care of the author.

To all my classmates and peers. Especially Isabel and Fernanda, who were a source of knowledge and encouragement. Thank you for teaching me with such care and patience.

To my dearest friends: Giovanna, Bernardo, Juliana, Caroline, Lucas, Renata, Giovana. You were a home far from home.

To the Verônica of the past, for persisting and believing in herself.

Todas as vitórias ocultam uma abdicação.
— Simone de Beauvoir

Abstract

This work presents the operational modeling of an urban air mobility company based on VTOL aircraft, with a focus on eVTOLs, within the context of São Paulo. The proposal addresses the need for innovative and sustainable short-range transportation solutions in large metropolitan areas, particularly in response to traffic congestion and inefficiencies in ground mobility. The adopted methodology is grounded in a flexible parametric model capable of simulating various operational scenarios based on input variables such as daily passenger volume, aircraft capacity, operational time windows, and cost structure. The analysis includes the definition of optimized urban routes, fleet sizing, calculation of economic metrics (such as revenue, cost, and daily profit), and evaluation of fleet utilization rates. The model was implemented in Python and calibrated using real market data, taking inspiration from the operational structure of Revo, a Brazilian urban helicopter operator, and projections from Eve Air Mobility regarding electric-powered eVTOLs. Simulation results indicate that this type of operation can be both technically feasible and economically sustainable, provided it is supported by adequate vertiport infrastructure, integration with urban air traffic management systems, and public policies promoting sustainable air mobility.

List of Figures

List of Abbreviations and Acronyms

ANAC	National Civil Aviation Agency
CET	Traffic Engineering Company
IBGE	Brazilian Institute of Geography and Statistics
UAM	*Urban Air Mobility*
UATM	*Urban Air Traffic Management*
UAV	*Unmanned Aerial Vehicle*
UTM	*Unmanned Aircraft System Traffic Management*
eVTOL	*Electrical Vertical Take-Off and Landing*
VTOL	*Vertical Take-Off and Landing*

Contents

1 Introduction

The accelerated growth of urbanization in recent decades has posed significant challenges to mobility in large urban centers, making the development of innovative and sustainable solutions for the transportation of people and goods an urgent need. With more than 20 million inhabitants in its metropolitan region, according to census data from IBGE (2022), the city of São Paulo suffers from high levels of congestion, pollutant emissions, and productivity loss resulting from slow and inefficient commuting. Data from CET indicate that during peak hours, the average vehicle speeds can drop to less than 15 km/h on several arterial roads of the capital, compromising not only the movement of people but also the essential logistical flow of the regional economy.

In this context, Vertical Take-Off and Landing aircraft (VTOLs) emerge as a disruptive alternative, offering a new dimension of mobility: the low-altitude urban airspace. VTOLs encompass vehicles with conventional propulsion, such as helicopters, as well as more recent systems with electric propulsion and distributed architectures, such as eVTOLs (Electric VTOLs). The operational versatility of VTOLs allows them to operate in confined areas, reducing the need for large airport infrastructures and enabling the concept of Urban Air Mobility, which is already being explored in several cities around the world (RAJENDRAN; SRINIVAS, 2021); (TAKÁCS; HAIDEGGER, 2023).

São Paulo, with its dense geography and an already consolidated private air transport market — the city has one of the largest helicopter fleets in the world — presents promising structural and social conditions for the progressive introduction of VTOLs on urban routes. Unlike traditional helicopters, new VTOLs may offer lower operating costs, reduced noise, and zero emissions (in the case of eVTOLs), as well as being designed with a focus on automation and integration with urban air traffic control systems (HAGAG; HOEVELER, 2025).

However, the adoption of these vehicles on an urban scale depends on several critical factors: technical and regulatory feasibility, social acceptance, adaptation of urban infrastructure for vertiports and power networks, and, most importantly, the development of economically sustainable operational models. Although manufacturers such as Eve Air Mobility (a subsidiary of Embraer) have already announced partnerships to begin commercial operations with eVTOLs in the state of São Paulo by 2026, there is still a technical-scientific gap concerning the modeling of companies that will operate this new mode of transportation (REUTERS, 2024).

This work aims to propose an operational model for an urban airline based on eVTOLs, focused on the city of São Paulo. The approach includes the analysis of urban demand, selection and simulation of optimized routes, the requirements for physical and digital infrastructure (vertiports, UTM, recharging and supply networks), as well as the economic and environmental parameters involved. The model will be built from reasonable hypotheses adopted as guiding assumptions for the work, as well as regulatory trends under development in collaboration with ANAC.

Thus, this study not only addresses the need to deepen knowledge about urban air operations with eVTOLs but also provides relevant insights for future ventures, public policies, and urban planning. In summary, this work seeks to develop an operational model for an urban airline based on VTOLs in the city of São Paulo, focusing on the technical, operational, and regulatory feasibility of this proposal within the constraints imposed by the current infrastructure and urban environment of the metropolis.

1.1 Hypotheses

Based on studies from published articles on urban air transport using VTOLs, it can be observed that the analysis of the modeling of a future operation, not only in Brazil but also worldwide, is still in the development and learning stage among researchers. However, it is assumed that the results obtained will support the following hypotheses:

1. The eVTOL operation model can be represented by a structure based on discrete event simulation, integrating logistical, energetic, and regulatory aspects;
2. The existing urban infrastructure (heliports and airports) can be partially reused and adapted for initial eVTOL operations in São Paulo;
3. The initial eVTOL operation can be modeled based on priority routes, such as connections between financial centers and airports, using real air and ground traffic data;
4. The application of logistical optimization and urban air traffic management (UATM) algorithms can ensure adequate operational performance at the scale proposed for São Paulo;

1.2 Objectives

The general objective of this work is to develop an operational model for an urban airline based on eVTOL aircraft, considering the technical, logistical, regulatory, and geographical characteristics of the city of São Paulo. To this end, the following specific objectives are highlighted:

1. Develop a parametric operational model to simulate the operation of an urban airline based on eVTOLs, considering technical, logistical, and economic aspects.

2. Assess the feasibility of eVTOL operations in São Paulo, using real demand and infrastructure data, focusing on priority routes and efficient fleet utilization.

3. Analyze the simulation results using indicators such as travel time, cost, profit, service capacity, and utilization rate, to support strategic decisions and future implementations.

2 Literature Review

Since the 2000s, the advancement of composite materials, digital control systems, and, most importantly, electric and distributed propulsion technologies has driven the emergence of a new generation of Vertical Take-Off and Landing (VTOL) aircraft aimed at civil and urban use, consolidating the concept of Urban Air Mobility (RAJENDRAN; SRINIVAS, 2021);(TAKÁCS; HAIDEGGER, 2023). The historical evolution of these vehicles dates back to the Focke-Wulf Fw 61 helicopter in 1936, followed by the Sikorsky R-4 during World War II, considered the first VTOL produced on a large scale (KUHN, 1974). In the following decades, projects such as the tail-sitter Convair XFY-1 and the tiltrotor Bell XV-3 explored new operational configurations, culminating in the 1960s Harrier Jump Jet, which employed thrust vectoring with turbofan engines as a solution for high-speed transitions (FINGER *et al.*, 2017). These advancements gradually shifted the focus of VTOL development from military applications to urban solutions aimed at civil transportation.

Currently, VTOLs range from conventional single-main-rotor helicopters with anti-torque systems to more complex configurations, such as tiltrotors, tilt-wings, tail-sitters, and distributed electric propulsion systems (MOUSAEI *et al.*, 2022);(FINGER *et al.*, 2017). These architectures present significant challenges in terms of aerodynamic stability, energy efficiency, and control during flight transitions — especially in densely built urban environments (BANIK *et al.*, 2024). In this context, the technical understanding of these typologies is essential to guide operational, regulatory, and logistical decisions in the process of modeling an airline focused on urban mobility with VTOL aircraft.

2.1 Analysis of Revo's Business Model

Revo is a company operating in the urban air mobility sector in São Paulo, structuring its business around the sale of individual seats in twin-engine helicopters for urban and regional routes (CNN Brasil, 2024);(Exame, 2024). Its model aims to serve high-income clients with a fast, safe, and integrated travel experience while maintaining controlled costs through a lean and digitized operational architecture (ECONÔMICO, 2024).

2.1.1 Structure and Business Model

Revo's model is based on a digital platform and shared-operation logic. The company does not require full aircraft chartering but allows the purchase of individual seats on flights with predefined routes, schedules, and fares. This approach significantly reduces the entry barrier and increases aircraft occupancy rates.

In addition, Revo uses its own digital infrastructure, through a mobile app and web platform, for the entire process of booking, payment, check-in, and flight tracking. The customer experience is highly personalized, featuring concierge services, on-site hostesses at helipads, and integration with premium ground transportation. The goal is to offer a high-standard, door-to-door journey with competitive travel times compared to São Paulo's road traffic.

Another relevant aspect is operational integration with strategic partners — particularly helicopter operators and companies within its parent group. This allows Revo to operate with an asset-light structure, minimizing fixed costs and maximizing fleet flexibility.

2.1.2 Infrastructure and Operational Standards

Revo uses existing helipads in São Paulo, strategically located in areas of high economic density and near major business centers. Regular routes are planned to connect regions such as Faria Lima, Alphaville, and the Campo de Marte and Guarulhos airports, with flight times ranging from 8 to 12 minutes.

For weekend and holiday destinations, the company also operates seasonal routes to locations such as Ilhabela, Juquehy, Maresias, and Fazenda Boa Vista, using twin-engine helicopters with a capacity of five to eight passengers. This modular structure allows the route network to be adapted according to demand, maximizing occupancy and minimizing aircraft idle time.

All aerial operations are conducted exclusively with twin-engine helicopters, crewed by two pilots on every flight, reinforcing the safety levels required by this customer profile. This choice also anticipates the operational certification requirements for eVTOLs, which will likely need to maintain high safety standards to obtain authorization for large-scale urban operations.

2.1.3 Strategic Partnerships and Adoption of eVTOLs

Revo maintains a partnership with Eve Air Mobility, an Embraer company, with plans to acquire up to 50 eVTOLs (EMBRAER, 2024). This initiative is part of the company's technological transition strategy, anticipating the gradual replacement of its helicopter fleet with low-noise, electric aircraft capable of autonomous or semi-autonomous operation.

In collaboration with Eve and regulatory authorities, Revo has participated in advanced urban air traffic simulations in São Paulo. These tests were conducted using Eve's digital platform, Eve Vector, which can simulate hundreds of simultaneous flights, integrating information about vertiports, aircraft flow, weather, and airspace restrictions. The results of these simulations directly contributed to the design of viable operational scenarios for eVTOLs, helping establish performance metrics, safety protocols, and layouts for urban vertiports.

Beyond partnerships with aircraft manufacturers, Revo relies on the structure of its parent company, Omni Helicopters International (OHI), which provides logistical support, regulatory expertise, and an operational base with dozens of aircraft and certified pilots.

2.1.4 Economic Strategy and Use of Technology

From an economic standpoint, Revo's model is driven by operational efficiency and perceived value generation. The company avoids capital immobilization in fixed assets such as aircraft or helipads, instead opting for operational partnerships and shared infrastructure models. This enables operational leverage with low financial risk.

Additionally, intensive use of data and artificial intelligence algorithms allows for the dynamic adjustment of schedules, routes, and pricing based on observed demand, road traffic, weather conditions, and other external factors. The company also performs predictive analyses to optimize fleet utilization, anticipate operational failures, and identify expansion opportunities.

With over 100% growth in bookings during the first half of 2024, Revo already demonstrates sufficient market maturity to serve as a benchmark for eVTOL operators in cities with similar characteristics.

3 Methodology

The methodology adopted in this work was structured based on three integrated analyses — **Ground Supply**, **City Effect**, and **Economic Feasibility** — which aim to understand the performance of an urban eVTOL airline under three complementary dimensions: (i) operational capacity and infrastructure; (ii) impact on urban mobility; and (iii) financial sustainability.

These analyses were developed in *Python* within the computational environment *Anaconda (Jupyter Notebook)*, which allows interactive, modular, and visually accessible modeling. The use of Jupyter provided code transparency, ease of visualization of results, and traceability of parameters and formulas used.

3.1 General structure and base parameters

The study considers five strategic urban routes in the context of the city of São Paulo: Faria Lima–Guarulhos, Pinheiros–Centro, Paulista–Tatuapé, Faria Lima/Pinheiros–Alphaville, and Santana–Morumbi. These selected routes, shown below in Figure 3.1, represent high-traffic and economically connected corridors, chosen for their logistical relevance and potential integration with existing vertiports.

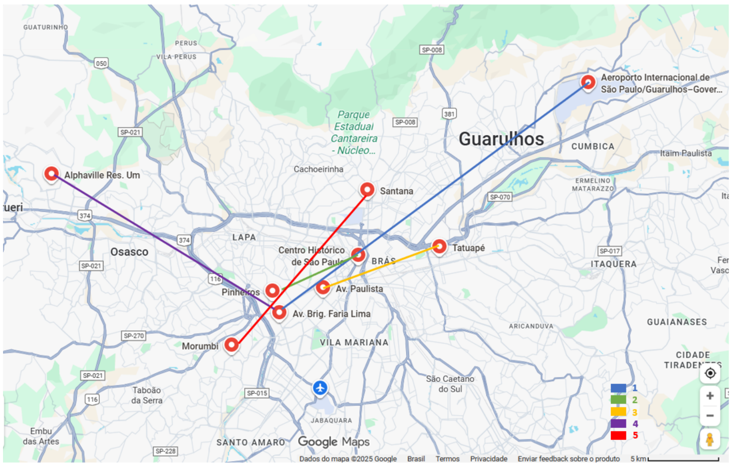


FIGURE 3.1 – Illustrative map of the selected routes for the city of São Paulo.

The input variables (*inputs*) common to the analyses are presented in Table 3.1.

TABLE 3.1 – Main input variables of the model.

Symbol	**Description**	**Unit**
λ	Passenger arrival rate (demand)	pax/h
μ	Passenger service rate	pax/h
c	Number of simultaneous pads	
T_{turn}	Average turnaround time	min
T_{over}	Operational overhead	min
LF	Load factor	dimensionless
P_{ticket}	Ticket fare per passenger	R$/pax
C_{unit}	Unit operating cost per passenger	R$/pax
C_{faixa}	Nominal capacity per road lane	veh/h
N_{faixas}	Number of lanes per direction	
O_{carro}	Average car occupancy	pax/veh

The main expected results (*outputs*) in each analysis include:

- System utilization (ρ);
- Average waiting time (W_q);
- Total service capacity (μ_{total});
- Number of vehicles removed from roads (veh/h);
- Percentage of congestion relief;
- Economic indicators: CASM, RASM, and operating margin.

Furthermore, to facilitate the visualization of the overall code structure, the flowchart shown in Figure 3.2 was designed.

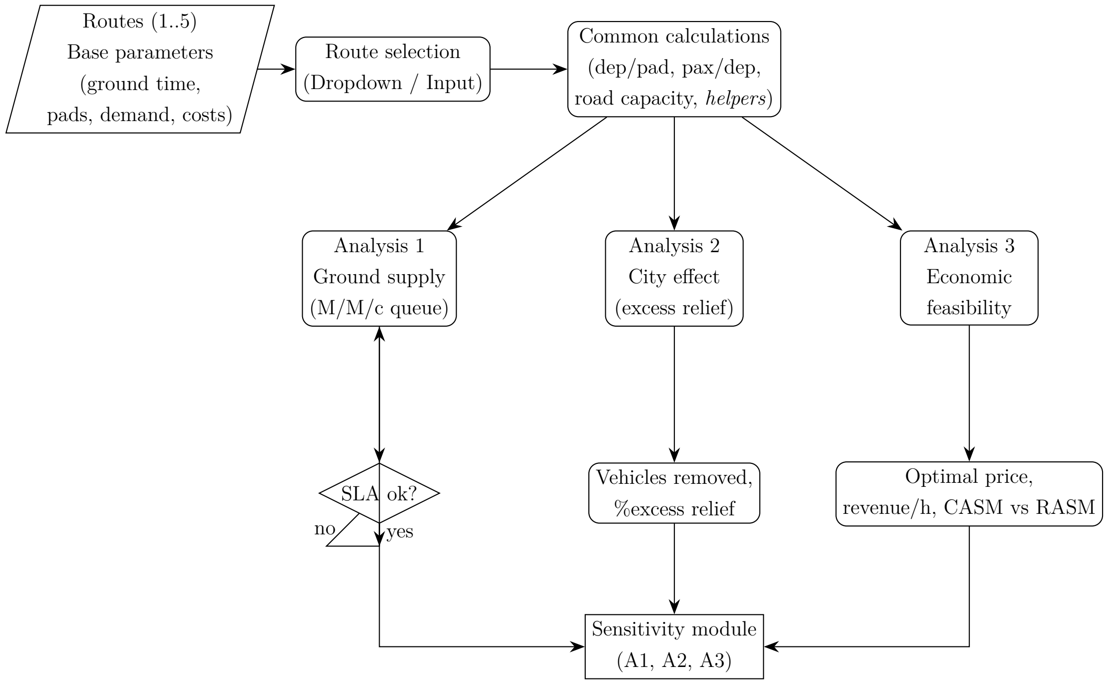


FIGURE 3.2 – General study flow: route selection, common calculations, and the three analysis blocks, with sensitivity module.

3.2 Analysis 1 — Ground Supply

The Ground Supply analysis aims to dimension the operational capacity of a vertiport relative to expected demand, considering the number of pads (c), turnaround time (T_{turn}), and overhead (T_{over}). The system is modeled as an **M/M/c queue**, with exponential arrivals and services and c servers (pads).

Formulas used

The service rate per pad (μ_{pad}) is given by:

$$\mu_{pad} = \frac{60}{T_{turn} + T_{over}} \times n_{pax/flight} \times LF \tag{3.1}$$

The total vertiport service rate is:

$$\mu_{total} = c \times \mu_{pad} \tag{3.2}$$

The average system utilization is defined as:

$$\rho = \frac{\lambda}{\mu_{total}} \tag{3.3}$$

The probability that a passenger waits in the queue is given by the Erlang-C formula:

$$P_c = \frac{\frac{a^c}{c!}\frac{c}{c-a}}{\sum_{k=0}^{c-1}\frac{a^k}{k!} + \frac{a^c}{c!}\frac{c}{c-a}} \tag{3.4}$$

where $a = \frac{\lambda}{\mu_{pad}}$.

The average waiting time (W_q) is calculated by:

$$W_q = \frac{P_c}{c\mu_{pad} - \lambda} \times 60 \tag{3.5}$$

The results of this analysis are presented as **heatmaps**, showing the average waiting time (W_q) as a function of demand (λ) and number of pads (c), allowing identification of combinations that ensure stability ($\rho < 1$) and compliance with the maximum waiting time SLA. The representative flowchart for Analysis 1 is shown below in Figure 3.3.

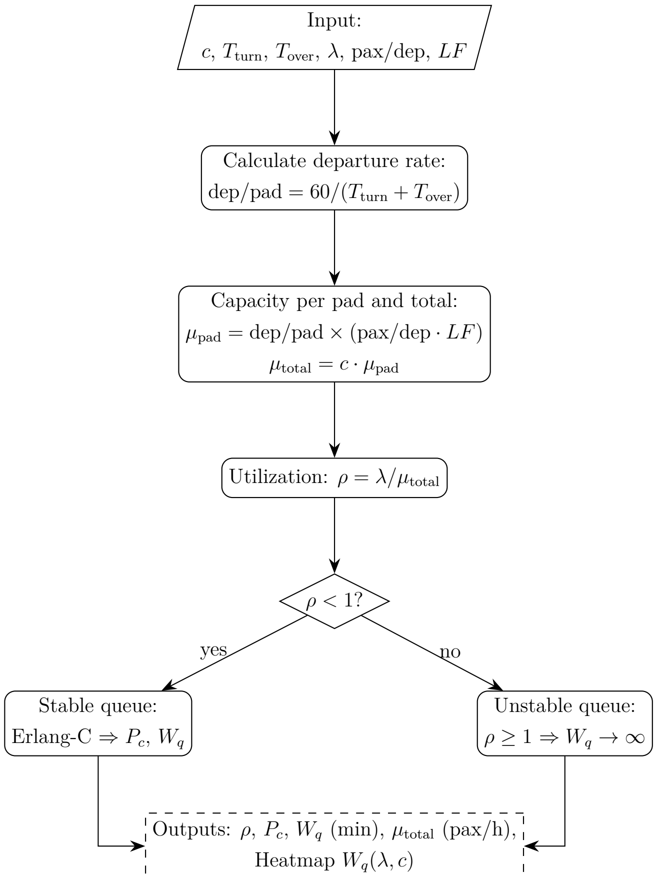


FIGURE 3.3 – Flowchart of Analysis 1 (M/M/c queue) — sequential calculation of capacity, stability, and average waiting time.

3.3 Analysis 2 — City Effect

The City Effect analysis quantifies the impact of eVTOL operations on the urban road system, estimating potential congestion reduction through passenger migration from cars to air transport. Each route was modeled based on traffic parameters: effective road capacity, number of lanes, peak demand, and average car occupancy.

In this project, all routes — except Pinheiros–Centro — were classified as **freeways** (urban expressways), while Pinheiros–Centro is classified as a **main arterial road**. Freeways have controlled access, continuous flow, and **4 lanes per direction**, with an average speed of **80–100 km/h** and capacity of **2,000–2,400 vehicles/hour/lane**, totaling around **8,000–9,000 vehicles/hour/direction**. Considering an average occupancy of 1.4 passengers per vehicle, the potential demand reaches **11,000–12,000 passengers/hour/direction** ((DNIT), 2010; (TRB), 2020; (CET-SP), 2022; (PMSP), 2021).

The **arterial route Pinheiros–Centro** has intersections and urban interferences, operating with **3 lanes per direction**, a speed of **40–60 km/h**, and capacity of **1,600–1,900 vehicles/hour/lane**, equivalent to **5,400–5,700 vehicles/hour/direction**, or approximately **7,500 passengers/hour/direction**. This differentiation was essential for calibrating the capacity and city-effect models adopted.

Formulas used

The effective capacity of the road is calculated as:

$$Cap_{via} = N_{faixas} \times C_{faixa} \times f_{conf} \tag{3.6}$$

where $f_{conf} = 0.85$ is the operational reliability factor.

The excess of vehicles, representing the volume that exceeds the road capacity, is given by:

$$Excesso_{veh/h} = \max(0, Demanda_{veh/h} - Cap_{via}) \tag{3.7}$$

The number of vehicles removed from the road, considering the passengers served by eVTOL, is:

$$Veh_{removidos/h} = \frac{Pax_{serv/h}}{O_{carro}} \tag{3.8}$$

The percentage of congestion relief is calculated as:

$$Relief = \min\left(100, \frac{Veh_{\text{removed/h}}}{Excesso_{\text{veh/h}}} \times 100\right) \tag{3.9}$$

And the minimum number of aircraft required to offset the excess demand is given by:

$$N_{aircraft} = \frac{Pax_{serv/h}}{Cap_{eVTOL,dir} \times LF} \tag{3.10}$$

The results allow estimating, for each route, how many passengers would need to migrate to air transport to significantly reduce congestion and how many aircraft would be needed to serve that demand. The representative flowchart for Analysis 2 is shown below in Figure 3.4.

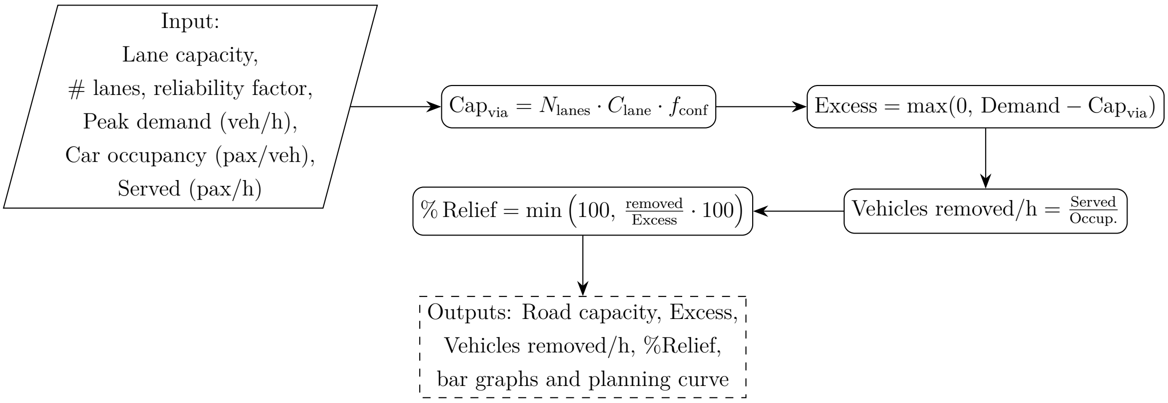


FIGURE 3.4 – Flowchart of Analysis 2 — urban impact (excess relief) given the air transport service.

3.4 Analysis 3 — Economic Feasibility

The Economic Feasibility analysis seeks to evaluate the financial sustainability of the operation, considering costs, fares, and demand. The model relates ticket price, demand elasticity, and capacity constraints, producing typical economic indicators of civil aviation.

To price helicopter operations, we adopted as reference the hourly cost of the AS350/H125 in São Paulo in the range of **R$ 7,000–8,700/h** (base: R$ 7,000/h), according to recent market references (Flapper, 2024); historical data in SP confirms (∼R$ 8,400/h in 2020) (Flapper, 2020), and current premium shuttle operations (Forbes Brasil, 2024). For eVTOLs, we assume a **40–50% discount** applied to the *cost* (CASM), maintaining the *same* tax (τ), fee/platform (f), and margin (m) structure as the helicopter; the customer price results from this reduced base.

$$C \;=\; \text{CASM} \cdot d \tag{3.11}$$

$$P \;=\; C \cdot \Big(1 + \tau + f + m\Big) \tag{3.12}$$

$$C_{\text{eVTOL}} \;=\; C_{\text{helo}} \cdot \Big(1 - \delta\Big), \qquad \delta \in [0.40,\, 0.50] \tag{3.13}$$

$$P_{\text{eVTOL}} \;=\; C_{\text{eVTOL}} \cdot \Big(1 + \tau + f + m\Big) \tag{3.14}$$

$$\frac{P_{\text{eVTOL}}}{P_{\text{helo}}} \;=\; \frac{C_{\text{eVTOL}}}{C_{\text{helo}}} \;=\; 1 - \delta \quad \text{(when } \tau, f, m \text{ are equal in both operations).} \tag{3.15}$$

Where: CASM is the cost per passenger-kilometer (R$/pax-km); d is the route distance (km); C is the seat cost; P is the ticket price per seat; τ is the tax fraction; f the platform fee fraction; m the operating margin fraction; and δ is the relative cost discount of the eVTOL versus the helicopter.

Formulas used

The revenue per hour of operation is given by:

$$R = P_{ticket} \times \min(Demand(P_{ticket}), Cap_{total}) \tag{3.16}$$

The total hourly cost is expressed as:

$$C = (C_{unit} \times n_{pax/flight}) \times Dep/h \tag{3.17}$$

The margin per minute of vertiport utilization is:

$$Margin_{pad/min} = \frac{R - C}{Dep/h \times T_{turn}} \tag{3.18}$$

The cost per available seat kilometer (CASM) is defined as:

$$CASM = \frac{C_{flight}}{Dist_{km} \times n_{seats}} \tag{3.19}$$

And the revenue per available seat kilometer (RASM) as:

$$RASM = \frac{P_{ticket} \times LF}{Dist_{km}} \tag{3.20}$$

The headway and average waiting time (SLA) are determined by:

$$h = \frac{60}{Dep/h}, \quad SLA = \frac{h}{2} \tag{3.21}$$

These metrics make it possible to identify the equilibrium point between revenue, cost, and service quality. The model allows testing different fares and operating costs to determine the optimal price that maximizes margin while respecting the maximum waiting time established by the ground analysis. The representative flowchart for Analysis 3 is shown below in Figure 3.5.

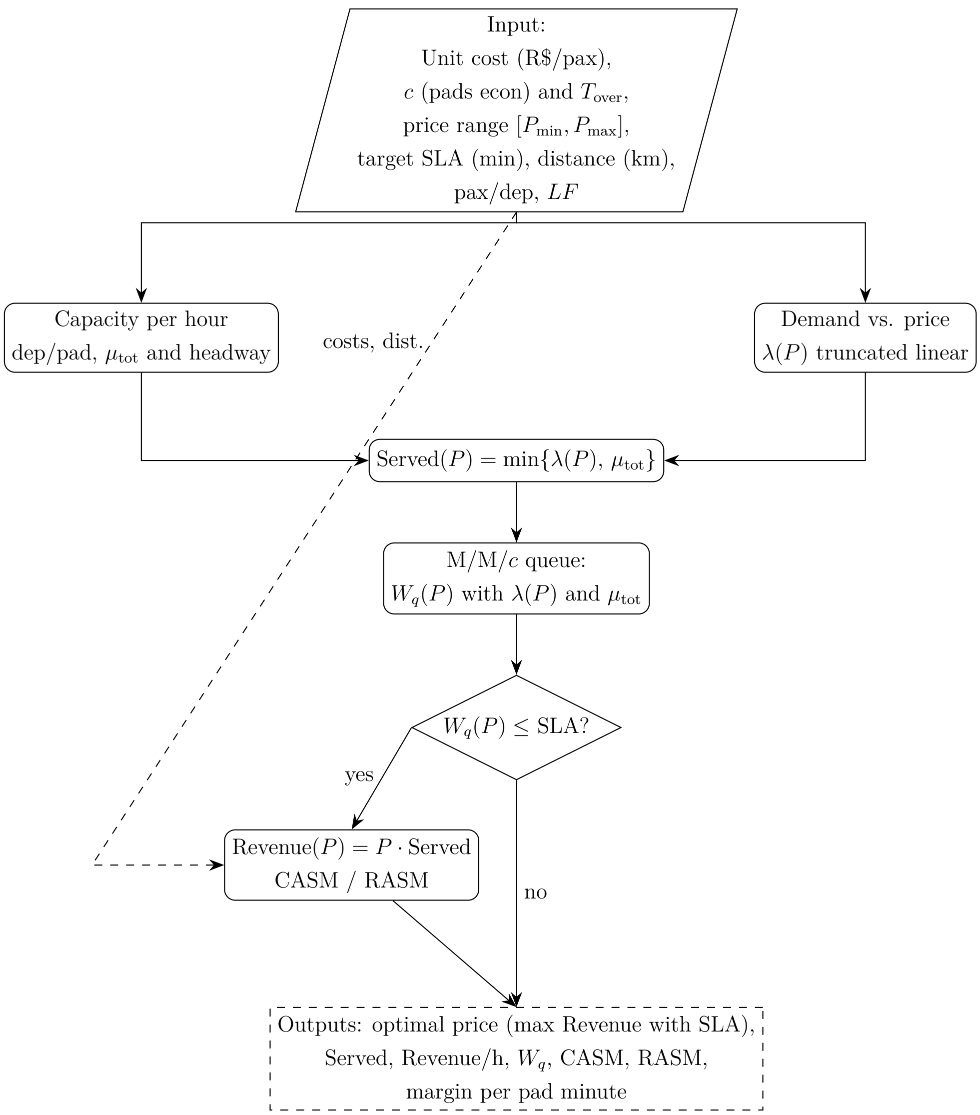


FIGURE 3.5 – Flowchart of Analysis 3 — price, demand, and capacity constraint with SLA criterion. The dashed line represents cost and distance parameters feeding the economic calculation.

3.5 Sensitivity Analysis

In addition to the three main analyses, a sensitivity analysis module was implemented to investigate the influence of key parameters on the operational, urban, and economic results of the system. This module, illustrated in Figure 3.6, performs systematic sweeps over variables such as number of pads (c), passenger arrival rate (λ), average car occupancy, effective road capacity, and price range, automatically recalculating the metrics of Analyses 1, 2, and 3. The procedure allows identifying regions of stability and critical operating zones — such as the $\rho \approx 1$ threshold in M/M/c models, road saturation points, and the optimal price under capacity constraint. Thus, the sensitivity analysis acts as a cross-validation step, demonstrating the robustness and limits of each simulated scenario before final interpretation of the results.

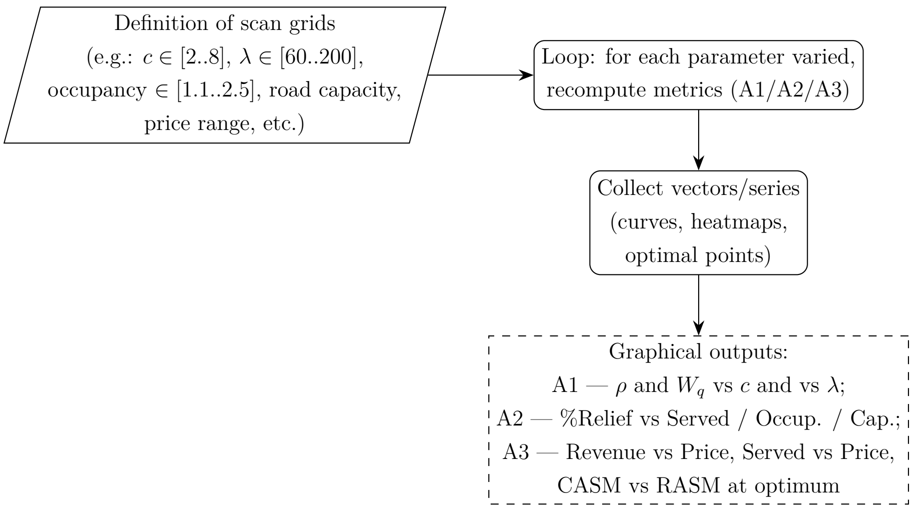


FIGURE 3.6 – Flowchart of the sensitivity module used for A1, A2, and A3.

3.6 Computational environment and simulation structure

The three analyses were implemented in **Python** within the **Anaconda** environment, using **Jupyter Notebook** as the interactive interface.

The main advantages of this choice include:

- **Interactivity:** use of `ipywidgets` to modify input parameters and visualize results in real time;
- **Visualization:** `matplotlib` and `numpy` libraries for generating graphs, heatmaps, and performance curves;

- **Modularity:** code separation by cells (definitions, calculations, and visualization), facilitating auditability and reproducibility;
- **Integration:** unification of operational, urban, and economic analyses in a single analytical tool.

This methodological architecture ensures didactic clarity, analytical robustness, and flexibility in evaluating eVTOL operating scenarios in urban environments.

4 Results

For the results presented below, a **Case Study** was conducted using Route 1 — Faria Lima–Guarulhos with the aim of demonstrating the three analyses chosen and described in Chapter 3. The values adopted for the analysis parameters are presented in each section. The other routes are equally suitable for studies like this; however, for the sake of summary, they were not included in the results. For the purposes of assessing the model's feasibility and the coherence of the parameters, the choice of a single route was considered sufficient.

4.1 Analysis 1 — Ground Supply

With the Faria Lima–Guarulhos route configured to have three pads, arrivals of 120 passengers per hour, and ground time composed of 8 minutes of *turnaround* and 1.5 minutes of *overhead*, the system enters an unstable regime, characterized by $\rho > 1$. The service rate per pad is limited by the turnaround time, given the maximum eVTOL capacity considered of 4 passengers: each pad can achieve approximately 6.32 departures per hour and, with four passengers per departure, delivers about 25.3 passengers per hour. With three pads, the total service capacity reaches approximately 75.8 passengers per hour, a value significantly lower than the imposed demand (120 passengers per hour). This relationship produces a utilization of $\rho \approx 1.58$, indicating that the system is operating above its service capacity. Operationally, this means that more passengers arrive than the system can board per unit time, and the queue grows indefinitely. This condition is reflected in the performance metrics shown in Table 4.1: the average waiting time (W_q) tends to infinity and the probability of waiting (P_c) approaches 1, indicating that all passengers will have to wait for service.

TABLE 4.1 – Numerical results — Analysis 1 (Ground Supply), Route 1: Faria Lima–GRU

Metric	**Symbol**	**Value**	**Unit**	**Remark**
Departures per pad	—	6,32	dep/h	$60/(T_{\text{turn}} + T_{\text{over}})$ with 8+1,5 min
Capacity per pad	μ_{pad}	25,26	pax/h	$6{,}32 \times 4$
Total capacity	μ_{total}	75,79	pax/h	$c \times \mu_{\text{pad}}$ with $c = 3$
Peak demand	λ	120,00	pax/h	Input parameter (scenario)
Utilization	ρ	1,58	—	$\lambda/\mu_{\text{total}}$ (unstable system)
Probability of waiting	P_c	1,00	—	Tending to 1 when $\rho \geq 1$
Average queueing time	W_q	∞	min	Undefined (infinite) when $\rho \geq 1$

In the heatmap obtained in Figure 4.1, this imbalance appears as a region without finite values in the area corresponding to three pads for demands above approximately 75–80 passengers per hour. As the number of pads increases or the demand decreases, colored areas with finite W_q values emerge, representing the system's stability frontier. In practical terms, the operation begins to stabilize for $\lambda = 120$ only when the number of pads is increased from three to five, at which point ρ drops close to 1 and W_q, although still high, becomes finite. When moving to six pads, the capacity margin grows ($\rho \approx 0.8$) and the average waiting time falls nonlinearly — typical behavior of M/M/c queueing systems, in which small capacity margins near $\rho = 1$ still produce long waits, but moderate margins reduce W_q significantly.

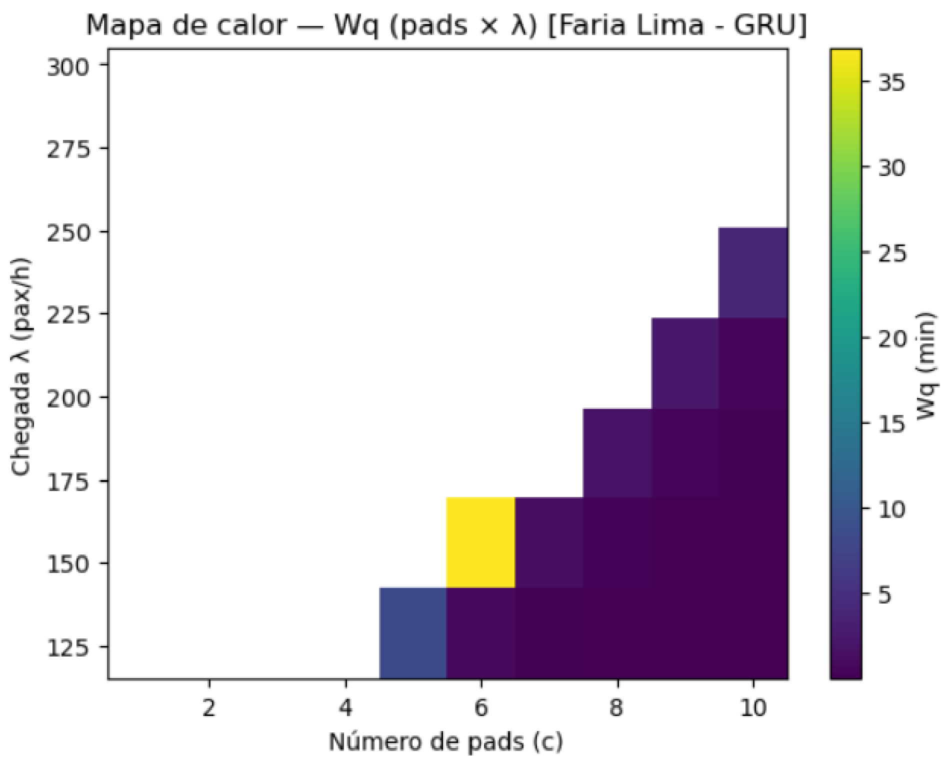


FIGURE 4.1 – Heatmap of the ground supply analysis for the Faria Lima–GRU route.

If the infrastructure remains fixed (three pads) and the effective demand (λ) is reduced, the same phenomenon is observed in reverse. As λ approaches the system's total capacity (around 75 passengers per hour), the average waiting time remains high. Only when demand drops to significantly lower values, around 50–60 passengers per hour, does W_q reach levels compatible with good service quality. This sensitivity shows that operating with utilization very close to 1 is structurally risky for user experience, as the system becomes unstable even with minimal demand variations.

Changes in ground times (*turnaround* and *overhead*) have a direct impact on total capacity. An increase in *overhead* from 1.5 to 3.0 minutes, keeping the same *turnaround* time, reduces the number of departures per pad and, consequently, the vertiport's capacity. This means that more pads would be required to keep the same demand stable. The opposite effect also occurs: a reduction of one or two minutes in ground times increases the number of departures per pad and can offset the need for infrastructure expansion. On short routes, such as those chosen in this work, the bottleneck tends to be the ground time, which makes operational gains of a few minutes as valuable as building new pads.

Increasing the number of passengers per departure (for example, from four to five) or a higher *load factor* has a similar effect: it raises the service rate per pad, reduces ρ, and lowers W_q. This variable represents an internal product factor — more seats or higher occupancy — while ground times and number of pads represent structural variables of infrastructure and process. The balance among these three elements determines efficient operation. Thus, the optimal strategy combines sufficient infrastructure (pads), lean processes (reducing *turnaround* and *overhead*), and high occupancy per flight to keep the system stable and with waiting times compatible with the established SLA.

In summary, the analyzed scenario highlights the importance of balancing supply and demand. With three pads and arrivals of 120 passengers per hour, the system is saturated and unable to meet demand. With five pads, the operation becomes stable, and with six, it reaches adequate performance levels. Alternatively, stability could be achieved while maintaining three pads, provided that demand were reduced, ground times optimized, or per-flight occupancy increased. Thus, the analysis shows that there are multiple paths to achieving operational efficiency, but all depend on balancing capacity, turnaround time, and passenger volume served.

4.2 Analysis 2 — City Effect

For the second analysis, a scenario was adopted with an effective road capacity of 4,500 vehicles per hour (due to the choice of the Faria Lima–GRU route), a peak demand of 6,000 vehicles per hour, and a rate of 120 passengers per hour served by the air system.

In addition, an average car occupancy of 1.6 passengers per vehicle was adopted, a value consistent with urban peak measurements in São Paulo.

From these data, it is calculated that the excess vehicles on the road — that is, the volume that exceeds the discharge capacity — corresponds to approximately 1,410 vehicles per hour. The proposed air operation, serving 120 passengers per hour, would remove about 75 vehicles from the road each hour, considering the average occupancy of 1.6 passengers per car. This represents a reduction of approximately 5.32% of the excess traffic, as indicated in Table 4.2.

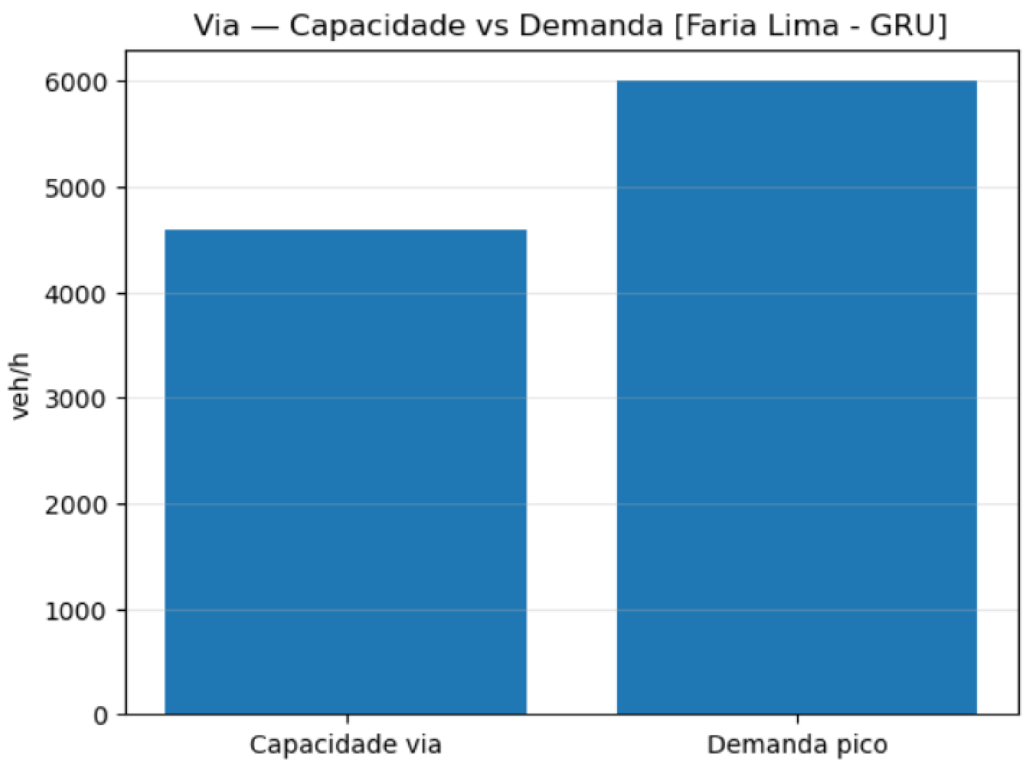


FIGURE 4.2 – Road capacity vs. demand on the Faria Lima–GRU route

The generated charts highlight the structural imbalance between road capacity and peak demand. In Figure 4.2, it is observed that demand (6,000 vehicles/h) exceeds effective capacity (4,500 vehicles/h) by 33%, characterizing a saturation situation typical of access corridors to Guarulhos Airport. Figure 4.3 shows the linear relationship between the number of passengers served per hour and the percentage of congestion relief: the greater the volume of passengers transported by the air system, the higher the percentage reduction of road excess, in direct proportion.

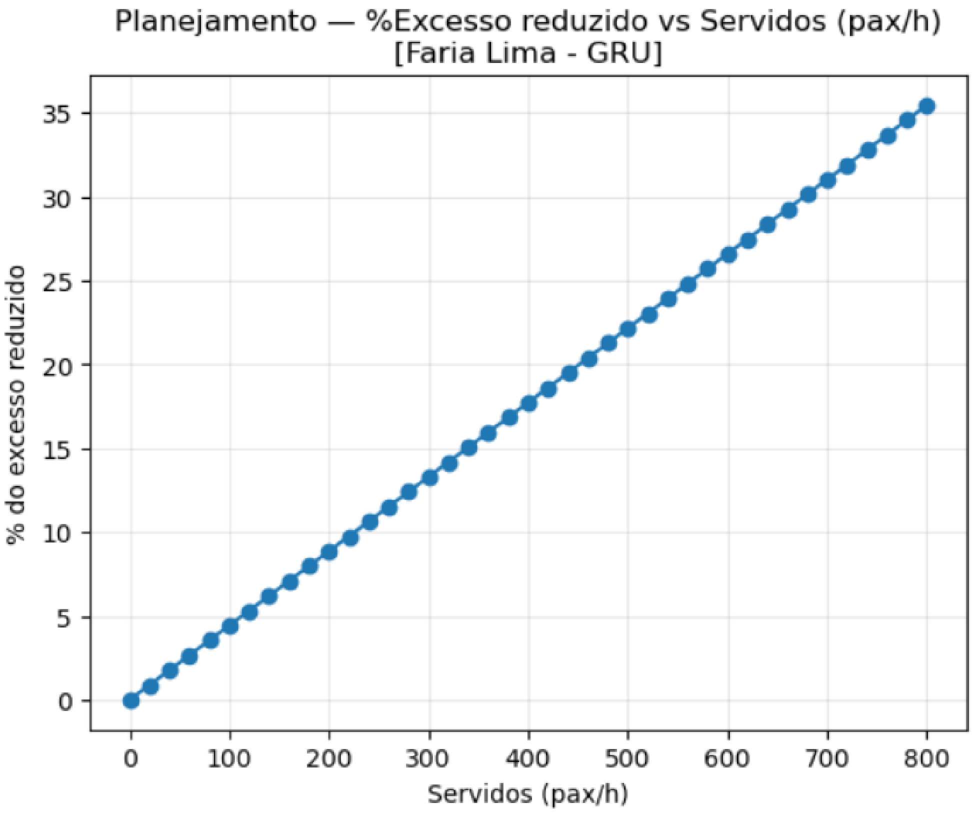


FIGURE 4.3 – Decrease in congestion volume as a function of the number of passengers served per hour

This linear relationship is explained by the equation used to calculate the percentage relief, defined as:

$$\text{Relief} = \frac{(\text{Served (pax/h)}/\text{Average occupancy (pax/veh)})}{\text{Excess (veh/h)}} \times 100 \tag{4.1}$$

According to this expression, each passenger transported by the air system represents a fractional removal of a vehicle from road traffic. Thus, the total effect is directly proportional to the number of passengers served and inversely proportional to the average occupancy of private vehicles.

In an operational context, this sensitivity indicates that the eVTOL system exerts increasing influence on urban flow as its capacity expands. If the number of passengers served per hour doubled to 240, the number of vehicles removed would also double to 150 vehicles per hour, raising the excess relief to about 10.6%. Similarly, a reduction in average car occupancy (for example, to 1.2 passengers per vehicle) would result in a significant gain in the percentage of relief, since the passenger–vehicle equivalence becomes more favorable to air transport.

The model also allows for estimating operational targets. To reduce the traffic excess by 20%, approximately 451 passengers per hour would need to be served by the air system; to eliminate one third of the excess (33%), the number rises to around 753 passengers per hour; and to fully offset the current excess of 1,410 vehicles per hour, a service rate of about 2,256 passengers per hour would be required. These values provide sizing parameters for fleet planning, infrastructure, and operating frequency.

Finally, it is important to note that the system's percentage impact also varies with road conditions. Increases in capacity (for example, from 4,500 to 5,000 vehicles/h) reduce the excess and, consequently, amplify the percentage relief produced by the same air operation. In scenarios of worsening demand (for example, from 6,000 to 6,500 vehicles/h), the excess grows and the relative effect of eVTOLs is diluted. Thus, the efficiency of the air operation depends not only on its own scale but also on the dynamic behavior of the road system, reinforcing the importance of integrating ground and urban air mobility policies.

In summary, the analyzed scenario shows that, even with an initial operation of 120 passengers per hour, the eVTOL system already produces a measurable and positive impact on the traffic flow of the Faria Lima–Guarulhos route, mitigating about 5% of the excess traffic. This percentage tends to grow linearly with the expansion of air capacity, indicating that scalable urban vertical mobility operations can become effective decongestion tools when integrated into critical city corridors.

TABLE 4.2 – Numerical results — Analysis 2 (City Effect), Faria Lima–Guarulhos Route

Metric	**Symbol**	**Value**	**Unit**	**Remark**
Effective road capacity	—	4,500	veh/h	Number of lanes × capacity/lane × reliability factor
Peak demand	—	6,000	veh/h	Total traffic volume at the critical hour
Estimated excess	—	1,410	veh/h	Difference between demand and capacity
Passengers served	Served	120	pax/h	Demand served by the eVTOL system
Average vehicle occupancy	O_{carro}	1,6	pax/veh	Average observed at urban peak
Vehicles removed	—	75,0	veh/h	Served/O_{carro}
Percentage relief of excess	Relief	5,32	%	(75/1410) × 100

4.3 Analysis 3 — Economic Feasibility

The scenario considered adopts the following parameters: unit operating cost of R$650.00 per passenger, three available pads, *overhead* time of 1.5 minutes, *turnaround* time of 8 minutes, four passengers per departure, and 6.32 departures per pad per hour. The price limits were defined between R$300.00 and R$1,500.00, distributed across 25 points, and the maximum acceptable waiting-time SLA was set at 10 minutes. The simulation estimates a demand decreasing as a function of price, with linear behavior up to the saturation point, and seeks the balance between price and passenger volume to maximize hourly revenue.

The results shown in Table 4.3 indicate that the optimal price for the route is approximately R$950.00, at which the air operation reaches its highest hourly revenue of about R$24,000.00. At this price, demand is approximately 30 passengers per hour, of whom 25.3 are effectively served, considering the system's capacity and the operational limitation of three pads. The calculated average waiting time was 4.8 minutes, meeting the SLA requirement (less than 10 minutes). The margin per pad-minute reached R$150.00, indicating good use of the available infrastructure.

The economic efficiency metrics show an average operating cost (*Cost per Available Seat Kilometer* — CASM) of R$27.08 per askm and an average revenue per available seat-kilometer (*Revenue per Available Seat Kilometer* — RASM) of R$39.58 per askm. The fact that RASM exceeds CASM confirms the profitability of the operation at this equilibrium point.

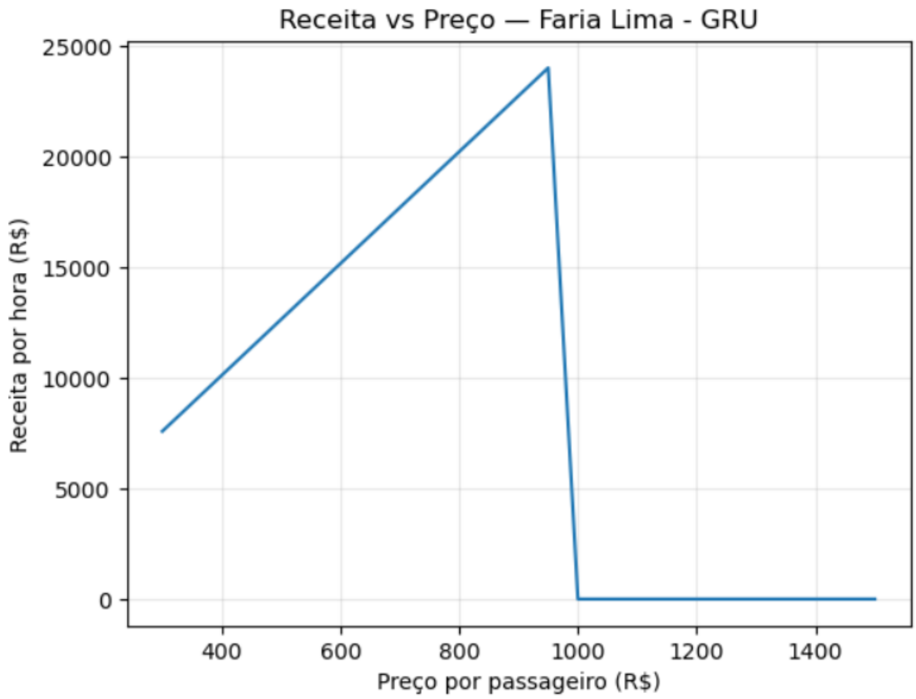


FIGURE 4.4 – Revenue behavior by fare charged to the passenger

Figure 4.4 highlights the classic price elasticity of demand. The curve increases up to the point of R$950.00, where total revenue is maximal, and drops sharply after this value, reflecting the upper limit of users' willingness to pay. From this point on, the price increase reduces the number of passengers served more quickly than the unit gain per passenger, leading to a steep decline in revenue.

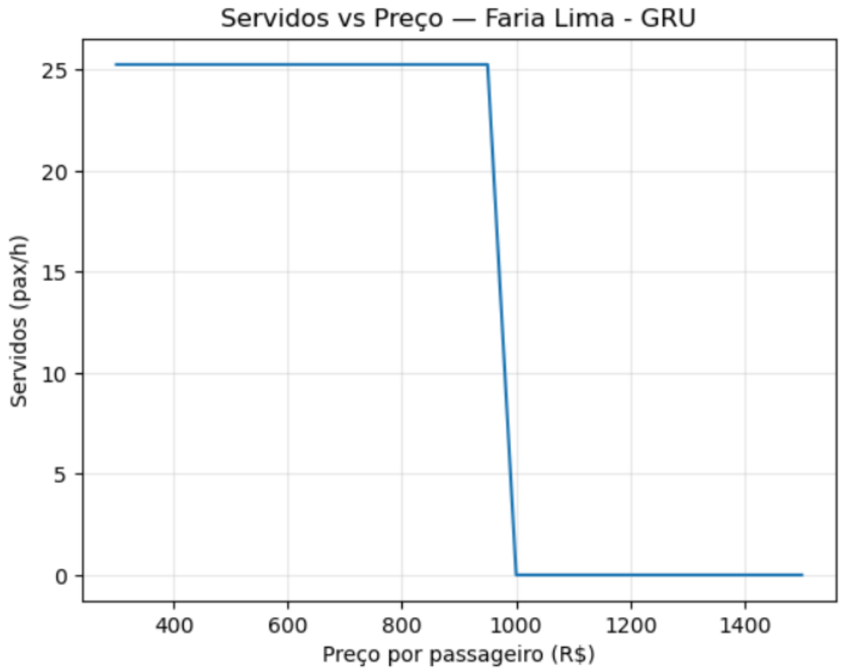


FIGURE 4.5 – Number of passengers served per hour as a function of the fare charged per passenger

Figure 4.5 shows the number of passengers effectively served per hour as a function of price. The curve remains practically constant up to the optimal price and plummets thereafter, demonstrating the point at which the perceived cost exceeds the market's tolerance. This indicates a price boundary for the urban air product — beyond which demand elasticity becomes very high, making it impossible to balance occupancy and price.

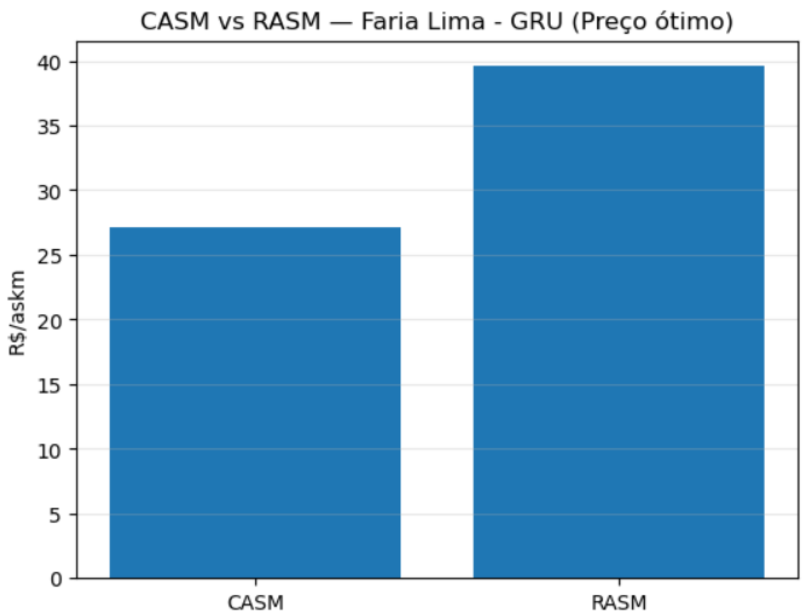


FIGURE 4.6 – Relationship between cost and revenue at the optimal price

Finally, Figure 4.6 presents, for the optimal price, the direct comparison between cost and revenue per seat-kilometer. RASM remains significantly above CASM, confirming that the system is economically viable at this point. This positive difference between the two metrics represents the operating margin, which is essential to sustain investment in fleet and infrastructure.

The analysis also allows us to infer the effect of parameter variations on the results. An increase in unit cost would shift the equilibrium point to higher prices, reducing the operating margin. Expanding the number of pads or reducing ground times (*turnaround* and *overhead*) would increase service capacity, boosting potential revenue and reducing average waiting time, which could allow for more affordable pricing. Conversely, raising the maximum SLA (for example, from 10 to 15 minutes) would increase system tolerance, enabling operation with a higher load and, therefore, higher marginal revenue — albeit with potential impact on user experience.

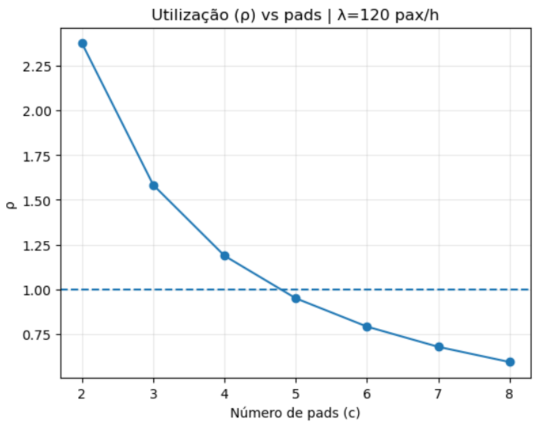
In summary, the model shows that the Faria Lima–Guarulhos route presents favorable economic conditions for eVTOL operations, with a balance between price and demand that ensures profitability and adequate service levels. The system's sensitivity to cost, price, and capacity variations reinforces the importance of integrated management among pricing, operational efficiency, and infrastructure sizing.

TABLE 4.3 – Numerical results — Analysis 3 (Economic Feasibility), Faria Lima–Guarulhos Route

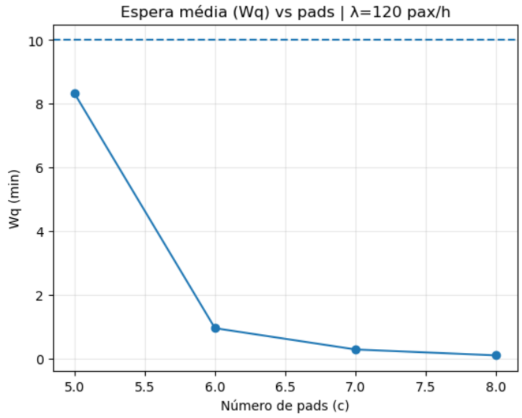
Metric	**Symbol**	**Value**	**Unit**	**Remark**
Unit cost per passenger	C_{pax}	650,00	R$/pax	Estimated average operating cost
Available pads	c	3	—	Number of simultaneous positions
Turnaround	T_{turn}	8,0	min	Ground time between cycles
Overhead	T_{over}	1,5	min	Additional time per operation
Departures per pad	—	6,32	dep/h	Calculated as $60/(T_{turn} + T_{over})$
Passengers per departure	—	4	pax/dep	Effective capacity
Optimal price	P_{opt}	950,00	R$	Point of maximum revenue
Estimated demand at this price	λ_p	30,0	pax/h	Equilibrium demand
Passengers served	—	25,3	pax/h	Limited by operational capacity
Total hourly revenue	R_h	24.000,0	R$/h	$P_{opt} \times$ passengers served
Average waiting time	W_q	4,8	min	Meets the 10 min SLA
Margin per pad-minute	—	150,0	R$/min	Net operating revenue
CASM	—	27,08	R$/askm	Cost per available seat-km
RASM	—	39,58	R$/askm	Revenue per available seat-km

4.4 Sensitivity Analysis

4.4.1 Ground Supply: sensitivity to the number of *pads* and to demand



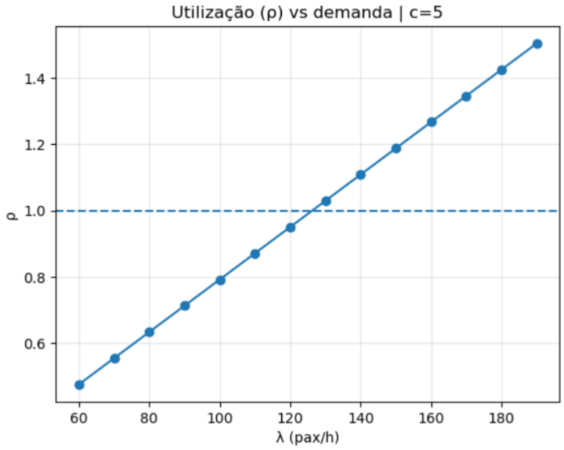
(a) Utilization (ρ) as a function of the number of *pads*.



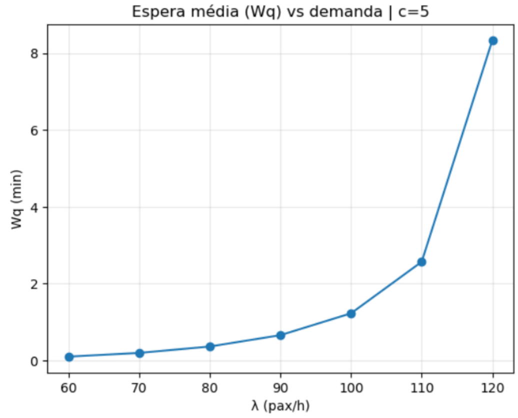
(b) Average waiting time (W_q) as a function of the number of *pads*.

FIGURE 4.7 – Sensitivity of ground capacity to changes in the number of *pads*.

Figures 4.7a and 4.7b link the vertiport's service capacity to the number of *pads*, keeping the hourly arrival rate and ground times constant. It is observed that the utilization curve ρ decreases monotonically as c increases. For very low c, ρ exceeds one, characterizing the instability region of the M/M/c system. The dotted line at $\rho = 1$ highlights this boundary: to its left, the queue diverges; to its right, the system stabilizes. The behavior of W_q reflects the same dynamics, but in a nonlinear fashion and with greater sensitivity when the operation is near the limit. For reduced c values (for example, $c = 2$ or $c = 3$), the average waiting time is high or even unworkable; with the progressive increase in *pads*, W_q drops quickly until reaching values close to zero. The presence of the dotted line in the W_q panel signals a target SLA (e.g., 10 minutes) and makes it clear that a few additional units of c are enough to move from noncompliance to fully SLA-compliant operation, reinforcing the role of c as a primary leverage variable when ground times have already been streamlined.



(a) Utilization (ρ) as a function of demand



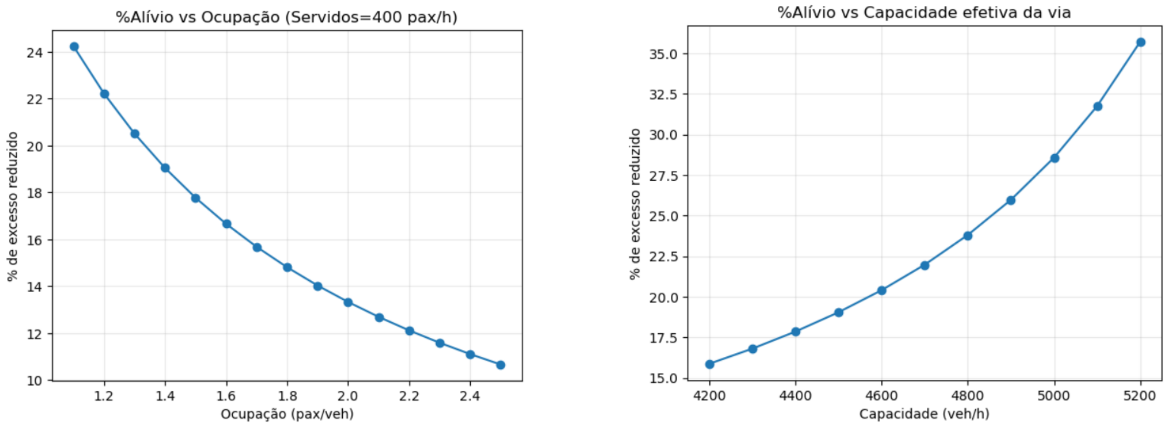
(b) Average waiting time (W_q) as a function of demand.

FIGURE 4.8 – Sensitivity of ground capacity to demand variations

Figures 4.8a and 4.8b repeat the exercise, but now fix c and vary demand λ. The ρ curve shows the expected behavior: approximately linear growth of utilization with arrivals, crossing the $\rho = 1$ line at a certain critical point. The integrated reading with the W_q panel shows the "cliff" effect typical of the M/M/c regime: while the operation remains away from saturation, W_q increases moderately; as ρ approaches one, small variations in λ induce disproportionate increases in W_q, until reaching the region in which the SLA is no longer met. This pair of charts confirms that, for fixed c, there is a safe demand range in which waiting time remains low; outside it, the system enters a highly sensitive zone, justifying demand throttling policies, boarding window management, or contingency plans to activate temporary *pads*.

4.4.2 City Effect: sensitivity to average occupancy and effective road capacity

The sensitivity results for the urban effect relate the effectively delivered air capacity (in passengers per hour) to the percentage reduction of excess vehicles on a road corridor. In Figure 4.9a, the controlled variable is average car occupancy (passengers per vehicle), keeping the number of passengers served per hour fixed. The decreasing curve illustrates the inverse relationship between occupancy and impact: the lower the occupancy, the greater the number of equivalent vehicles removed per passenger transported by air, increasing the fraction of excess reduced. In practical terms, corridors with low average occupancy exhibit greater sensitivity to the introduction of the eVTOL mode, as each air passenger replaces, on average, a more expressive fraction of individual vehicles. As occupancy increases, each removed passenger corresponds to a smaller quotient of vehicles, and the percentage gain narrows, although it remains relevant when the air operation scales.



(a) Road relief as a function of vehicle occupancy

(b) Road relief as a function of effective road capacity

FIGURE 4.9 – Sensitivity of the expected urban effect given air capacity.

Figure 4.9b analyzes the sensitivity of the same indicator to effective road capacity, keeping peak demand and passengers served per hour constant. The increasing curve reveals that relative relief rises when the road accommodates more vehicles per hour, because the excess is calculated as the difference between demand and capacity. When capacity is higher, the absolute excess is smaller, and the same number of removed vehicles represents a larger fraction of this excess. This effect has a direct implication for public policy: moderate road management interventions (signal synchronization, dynamic lanes, peak-hour restrictions) can, when combined with air operations, produce percentage gains significantly greater than those observed when capacity is depressed. Taken together, the two sensitivities show that the urban gain is the product of three multiplying forces: the scale in passengers served, the average car occupancy, and the corridor's effective capacity. Small improvements in any of these axes amplify the overall result.

4.4.3 Economic Feasibility: revenue vs. price sensitivity with capacity constraint

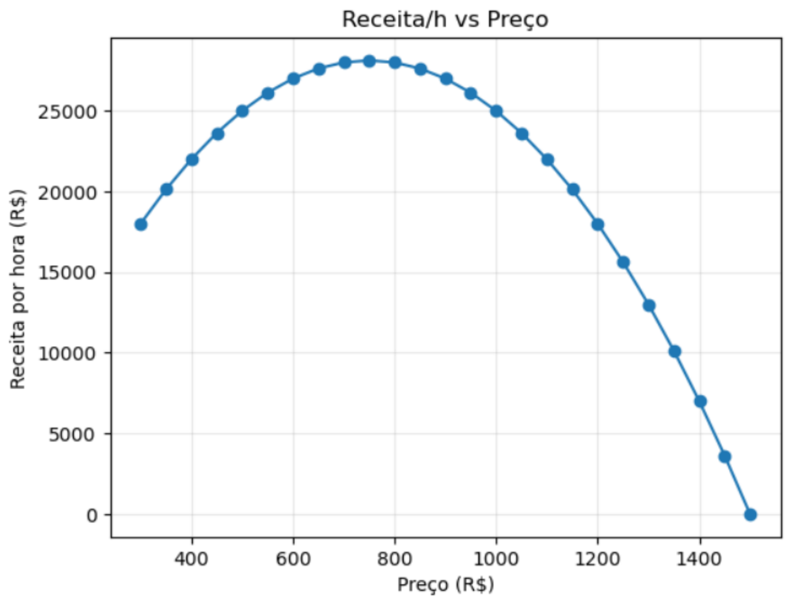


FIGURE 4.10 – Price optimization given demand and service capacity

The hourly revenue curve as a function of price, shown in Figure 4.10, synthesizes the interaction between demand elasticity and the vertiport's service capacity. A typical arc is observed: revenue grows with price while the loss of demand is offset by the higher ticket, reaches a maximum around the balance point between willingness to pay and servable volume, and declines after the "knee," when effective demand collapses. In the increasing section, the system's total capacity is not the limiting factor; the operational constraint emerges when demand at the optimal price approaches the vertiport's processing ceiling, a situation in which increasing *pads* or reducing ground times can shift the revenue maximum to the right or raise the plateau itself. In the decreasing section, the drop is dominated by elasticity: prices above users' acceptability limit make effective demand tend toward zero, and revenue follows this trajectory.

From a decision-making standpoint, the reading of the curve should be combined with the average waiting-time chart and the SLA from the ground analysis. If, at the optimal price, the average waiting time exceeded the SLA, there would be a conflict between economic and service objectives, requiring either a re-optimization of the price toward a region with lower $\lambda(P)$ or marginal investments in ground capacity to increase μ_{total}. If the optimal price is operationally viable and the difference between RASM and CASM remains positive, a sustainable operating zone is created; further reductions in *turnaround* or *overhead* raise the safety margin, allowing operation with slightly lower prices to capture market mass without degrading the SLA.

5 Conclusions

5.1 Conclusions: decision-oriented synthesis

The integrated analysis used in the case study confirms that the system's core parameters — ground times, number of *pads*, effective load per departure, price–demand elasticity, waiting-time SLA, average car occupancy, and road capacity — vary coherently with queueing theory and operational practice, and that their combined effects define a stable, profitable operating region with a measurable urban benefit for São Paulo.

5.1.1 Ground capacity and operational stability

It is concluded that reducing *turnaround* and *overhead* is the most efficient lever to expand capacity: minutes removed on the ground convert directly into more departures per *pad* and shift the system away from the instability threshold ($\rho \approx 1$), reducing W_q's sensitivity to demand variations. Increasing c (*pads*) is confirmed as a structural shock absorber: its variation shifts the entire stability frontier and ensures SLA compliance under peaks. However, the marginal gain from new *pads* is lower than that of the first minutes removed from the process; therefore, the optimal sequence is first **lean process**, then **physical capacity**.

5.1.2 Per-flight supply and window management

Effective load per departure, when increased while maintaining an agile process, raises the service rate and smooths the relative variability of arrivals, reducing queues at constant demand. The opposite effect occurs when the cabin grows without compatible *turnaround*: the theoretical gain in seats is consumed by higher *overheads*. Thus, the practical conclusion is that cabins and process must be co-designed so that the seat increase actually results in delivered capacity. Moreover, demand profiles with strong hourly concentration require disciplined use of windows/slots; scheduling variation is concluded to be essential to keep $\rho < 1$ precisely when the system is most sensitive.

5.1.3 SLA as a design criterion

A stricter SLA is concluded to be feasible provided it is supported by operational slack (lean process + sufficient c). Its effect is twofold: it raises perceived quality and sustains willingness to pay, at the cost of requiring greater planning discipline. When relaxed, it improves asset utilization but reduces differentiation versus premium alternatives. In São Paulo, for airport and corporate corridors, it is concluded preferable to work with rigorous SLAs and installed capacity sized for the peak, since willingness to pay in these markets depends heavily on predictability.

5.1.4 Price, elasticity, and revenue

The price–demand relationship shows the expected “knee”: below it, price variation increases revenue; above it, it precipitates a drop in volume and revenue. It is concluded that the optimal price depends both on elasticity and on the vertiport’s service ceiling: raising capacity (via ground minutes and/or *pads*) shifts the economic optimum, allowing more demand to be captured without violating the SLA. Thus, price ceases to be merely a commercial variable and becomes an **instrument of load control**, stabilizing the operation around the region of maximum revenue compatible with the SLA.

5.1.5 City effect and corridor prioritization

The case study shows that the eVTOL’s urban efficiency grows when average car occupancy is low and when effective road capacity is reasonable. It is concluded that these conditions maximize the percentage of excess reduced per passenger served. In São Paulo, this guides the prioritization of corridors with high value of time, low automotive occupancy, and road capacity susceptible to management gains — for example, corporate and airport connections. In these corridors, variations in eVTOL operational parameters convert more intensely into urban benefit.

5.1.6 Interdependencies and safe zones

The sensitivity curves converge to the same conclusion: operating near $\rho = 1$ is structurally risky, because small variations in any parameter — demand, ground minutes, effective seats — amplify W_q and break the SLA. The safe zone, therefore, is not a point, but a capacity *buffer*, maintained by a stable process, a prudent number of *pads*, and the use of price/scheduling as demand “thermostats.” Projects that combine these three fronts tend to remain stable even under stress.

5.1.7 Model efficiency and feasibility in São Paulo

The model proved efficient for decision support: it isolates levers, quantifies directions of effect, and allows, from a few parameters with clear operational interpretation, the derivation of implications for SLA, capacity, revenue, and urban impact. The assumptions adopted — M/M/c for boarding, truncated linear demand, and average costs per passenger — do not preclude the central conclusion: **there is technical and economic feasibility** to deploy an initial eVTOL network in São Paulo, provided the prioritized routes meet the above criteria and the operation is anchored in a lean ground process, moderate buffers of c, and active management of windows and prices. In summary, when parameters move in the recommended direction — fewer ground minutes, sufficient *pads*, effective seats compatible with the process, price modulating load, and routes with a strong "city effect" — the system converges to a stable operating region, with SLA met, revenue maximized, and verifiable urban benefit. This region conclusively defines the attractiveness and executability of implementation in São Paulo.

Bibliography

BANIK, D. *et al.* Advanced flight control systems for distributed electric propulsion. **Aerospace Science and Technology**, 2024.

(CET-SP), C. de Engenharia de Tráfego de S. P. **Relatório Técnico de Desempenho Viário e Capacidade Urbana – 2022**. São Paulo, 2022. Disponível em: https://www.cetsp.com.br. Acesso em: 11 out. 2025.

CNN Brasil. **Revo: segurança e exclusividade em deslocamentos com helicóptero bimotor**. 2024. https://www.cnnbrasil.com.br/branded-content/nacional/revo-seguranca-e-exclusividade-em-deslocamentos-com-helicoptero-bimotor/. Disponível em: link acima. Acesso em: 7 jun. 2025.

(DNIT), D. N. de Infraestrutura de T. **Manual de Projeto Geométrico de Vias Urbanas**. Brasília, 2010. Disponível em: https://www.dnit.gov.br/engenharia/manual-de-projeto-geometrico-de-vias-urbanas. Acesso em: 11 out. 2025.

ECONÔMICO, V. **Revo inova em transporte por helicóptero bimotor em São Paulo**. 2024. https://valor.globo.com/conteudo-de-marca/revo/noticia/2024/05/29/revo-inova-em-transporte-por-helicoptero-bimotor-em-sao-paulo.ghtml. Disponível em: link acima. Acesso em: 7 jun. 2025.

EMBRAER. **Eve, OHI e Revo colaboram para desenvolver ecossistema de mobilidade aérea urbana**. 2024. https://embraer.com/br/pt/noticias?slug=1207476-eve-ohi-e-revo-colaboram-para-desenvolver-ecossistema-de-mobilidade-aerea-
Disponível em: link acima. Acesso em: 7 jun. 2025.

Exame. **Revo: a plataforma de mobilidade aérea que conquistou Felipe Massa**. 2024. https://exame.com/negocios/revo-a-plataforma-de-mobilidade-aerea-que-conquistou-felipe-massa/. Disponível em: link acima. Acesso em: 7 jun. 2025.

FINGER, D. *et al.* Tiltrotor aerodynamic configurations and historical performance review. **Journal of the American Helicopter Society**, 2017.

Flapper. **Quanto realmente custa para fretar um helicóptero**. maio 2020. https://flyflapper.com/stories/pt-br/quanto-custa-para-fretar-um-helicoptero/. Inclui custo-hora de AS350 em rota curta em São Paulo (R$ 8,400/h em 10/05/2020). Disponível em: link acima. Acesso em: 11 out. 2025.

Flapper. **Quanto custa voar de helicóptero em São Paulo?** 2024. https://flyflapper.com/stories/pt-br/quanto-custa-voar-de-helicoptero-em-sao-paulo/. Preços estimados em jan/2024 para frota monomotor a turbina (AS350/H125) em São Paulo. Disponível em: link acima. Acesso em: 11 out. 2025.

Forbes Brasil. **Conexão Pelos Ares**. out. 2024. https://forbes.com.br/forbeslife/2024/10/conexao-pelos-ares/. Revo: assento a partir de R$ 2,500 em rotas GRU ↔ Faria Lima/Cidade Jardim. Disponível em: link acima. Acesso em: 11 out. 2025.

HAGAG, N.; HOEVELER, B. The feasibility of electric air taxis: balancing time-savings and co2 emissions — a joint case study of respective plans in paris. **CEAS Aeronautical Journal**, Springer, v. 16, p. 169–185, 2025.

KUHN, H. **The Evolution of VTOL Concepts: 1900–1974**. [*S.l.*], 1974.

MOUSAEI, M. *et al.* Control challenges in eVTOL flight transitions. **International Journal of Aerospace Engineering**, 2022.

(PMSP), P. do Município de S. P. **Plano Diretor Estratégico do Município de São Paulo: Lei nº 16.050/2014 com alterações até 2021**. São Paulo: Secretaria Municipal de Urbanismo e Licenciamento, 2021. Disponível em: https://gestaourbana.prefeitura.sp.gov.br. Acesso em: 11 out. 2025.

RAJENDRAN, K.; SRINIVAS, K. Urban air mobility: Design trends and safety considerations. **Journal of Aircraft Systems**, 2021.

REUTERS. **Embraer's Eve to launch eVTOL operations in Brazil by 2026**. 2024. https://www.reuters.com. Disponível em: link acima. Acesso em: 7 jun. 2025.

TAKÁCS, A.; HAIDEGGER, T. Autonomy levels and flight control in UAM vehicles: A review. **Aerospace Systems Engineering**, 2023.

(TRB), T. R. B. **Highway Capacity Manual 7th Edition: A Guide for Multimodal Mobility Analysis**. Washington, D.C.: National Academies of Sciences, Engineering, and Medicine, 2020. Acesso em: 11 out. 2025.

Annex A - Codes

A.1 Imports, data and functions common to the 3 analyses

```python
# CELL 1 - Imports, data and functions common to the 3 analyses

from dataclasses import dataclass
import math
import numpy as np
import matplotlib.pyplot as plt
import ipywidgets as w
from IPython.display import display, clear_output

# ===== Single route model (serves all 3 analyses) =====
@dataclass
class RouteParams:
    name: str
    distance_km_air: float
    evtol_time_min_oneway: float
    heli_time_min_oneway: float
    turnaround_min: float
    car_time_peak_min_low: float
    car_time_peak_min_high: float
    road_type: str                  # 'freeway' or 'arterial'
    lanes_gargalo: int
    demand_peak_veh_h: int
    car_occupancy: float = 1.3
    cap_per_lane_freeway: int = 1800
    cap_per_lane_arterial: int = 1200
    reliability_factor: float = 0.85
    load_factor: float = 1.0
```

```python
    pax_per_leg: int = 4

# ===== Numbered routes (1..5) =====
ROUTES = {
    1: RouteParams("Faria Lima - GRU", 24.0, 13.0, 10.0, 8.0, 70.0, 85.0, "freeway", 3, 6000),
    2: RouteParams("Pinheiros - Centro (Sé)", 6.0, 6.0, 6.0, 8.0, 20.0, 35.0, "arterial", 2, 2600),
    3: RouteParams("Paulista - Tatuapé", 9.0, 7.0, 6.0, 8.0, 30.0, 45.0, "freeway", 4, 8000),
    4: RouteParams("Faria Lima/Pinheiros - Alphaville", 24.0, 13.0, 10.0, 8.0, 55.0, 75.0, "freeway", 3, 6500),
    5: RouteParams("Santana - Morumbi", 14.0, 9.0, 8.0, 8.0, 55.0, 70.0, "freeway", 3, 5200),
}

# ===== Common helpers =====
def dep_per_pad(turnaround_min, overhead_min=2.0):
    # departures/h per pad
    return 60.0 / (turnaround_min + overhead_min)

def mmc_metrics(arrival_rate_pax_h, departures_per_hour, pax_per_departure, c):
    # M/M/c queue metrics: rho, Pc (Erlang-C), Wq_min (min), mu_total (pax/h)
    mu_pax_h = departures_per_hour * pax_per_departure
    mu_total = c * mu_pax_h
    rho = arrival_rate_pax_h / mu_total if mu_total > 0 else math.inf
    if rho >= 1.0:
        return {"rho": rho, "Pc": 1.0, "Wq_min": math.inf, "mu_total_pax_h": mu_total}

    a = arrival_rate_pax_h / mu_pax_h if mu_pax_h > 0 else math.inf
    sum_terms = sum((a**k)/math.factorial(k) for k in range(c))
    erlang_c_num = (a**c)/math.factorial(c) * (c/(c - a))
    P0 = 1.0 / (sum_terms + erlang_c_num)
    Pc = erlang_c_num * P0
    Lq = Pc * (arrival_rate_pax_h / (mu_total - arrival_rate_pax_h))
    Wq_min = (Lq / arrival_rate_pax_h) * 60.0
    return {"rho": rho, "Pc": Pc, "Wq_min": Wq_min, "mu_total_pax_h": mu_total}
```

```

def capacity_of_road(p: RouteParams) -> int:
    cap_lane = p.cap_per_lane_freeway if p.road_type == "freeway" else p.cap_per_lane_arterial
    return int(p.lanes_gargalo * cap_lane * p.reliability_factor)

def excess_flow(p: RouteParams) -> int:
    cap = capacity_of_road(p)
    return max(0, p.demand_peak_veh_h - cap)

def casm_rasm(distance_km: float, pax_per_departure: int, load_factor: float,
              cost_per_departure: float, price_per_pax: float):
    ask = distance_km * pax_per_departure
    casm = cost_per_departure / max(1e-6, ask)
    rasm = (price_per_pax * pax_per_departure * load_factor) / max(1e-6, ask)
    return casm, rasm

def margin_per_minute(price_per_pax, served_pax_h, cost_per_departure, departures_per_hour, pad_minutes_used_per_hour):
    revenue_h = price_per_pax * served_pax_h
    cost_h = cost_per_departure * departures_per_hour
    return (revenue_h - cost_h) / max(1e-6, pad_minutes_used_per_hour)

# Demand × price placeholder (replace with empirical curve when available)
def demand_at_price(price):
    a, b = 600.0, 0.6
    return max(0.0, a - b*price)

```

A.2 Global route selection

```
# CELL 2 - Global route selection (1..5)
rota_dd = w.Dropdown(
    options=[(f"{i}) {ROUTES[i].name}", i) for i in ROUTES],
    value=1, description='Route:'
)
```

```
display(w.VBox([w.HTML("<h3>Select the route</h3>"), rota_dd]))

```

A.3 Ground Supply

```
# CELL 3 - ANALYSIS 1: Ground Supply (wizard with pauses) + heatmap

# Steps:
# 1) Select pads
# 2) Select lambda (arrivals pax/h)
# 3) Select overhead
# 4) Run analysis and show results + heatmap

pads_step    = w.Dropdown(options=[2,3,4,6,8,10], value=3,
↪ description='Pads:')
arrival_step = w.Dropdown(options=[120,150,180,210,240,270,300],
↪ value=180, description='lambda (pax/h):')
over_step    = w.Dropdown(options=[1.5,2.0,2.5,3.0], value=2.0,
↪ description='Overhead (min):')

next1 = w.Button(description='Next ->', button_style='info')
next2 = w.Button(description='Next ->', button_style='info')
next3 = w.Button(description='Run analysis', button_style='success',
↪ icon='play')
back2 = w.Button(description='<- Back', button_style='')
back3 = w.Button(description='<- Back', button_style='')

out1 = w.Output()

step1_box = w.VBox([w.HTML("<b>Step 1/3 - Number of pads</b>"), pads_step,
↪ next1])
step2_box = w.VBox([w.HTML("<b>Step 2/3 - Arrival lambda (pax/h)</b>"),
↪ arrival_step, w.HBox([back2, next2])])
step3_box = w.VBox([w.HTML("<b>Step 3/3 - Ground overhead (min)</b>"),
↪ over_step, w.HBox([back3, next3])])

step2_box.layout.display = 'none'
step3_box.layout.display = 'none'
```

```

def go_step2(_):
    step1_box.layout.display = 'none'
    step2_box.layout.display = 'block'
def back_to_step1(_):
    step2_box.layout.display = 'none'
    step1_box.layout.display = 'block'
def go_step3(_):
    step2_box.layout.display = 'none'
    step3_box.layout.display = 'block'
def back_to_step2(_):
    step3_box.layout.display = 'none'
    step2_box.layout.display = 'block'

next1.on_click(go_step2)
back2.on_click(back_to_step1)
next2.on_click(go_step3)
back3.on_click(back_to_step2)

# ---- heatmap (kept) ----
def heatmap_wq(p, arrivals=[120,150,180,210,240,270,300], pads_list=range(1,11), overhead_min=2.0):
    dep = dep_per_pad(p.turnaround_min, overhead_min)
    pax_dep = int(p.pax_per_leg * p.load_factor)
    W = np.zeros((len(arrivals), len(pads_list)))
    W[:] = np.nan
    for i, lam in enumerate(arrivals):
        for j, c in enumerate(pads_list):
            mu_pax_h = dep * pax_dep
            mu_total = c * mu_pax_h
            rho = lam / mu_total if mu_total > 0 else math.inf
            if rho >= 1.0:
                continue
            a = lam / mu_pax_h
            s = sum((a**k)/math.factorial(k) for k in range(c))
            ec = (a**c)/math.factorial(c) * (c/(c - a))
            P0 = 1.0 / (s + ec)
            Pc = ec * P0
            Lq = Pc * (lam / (mu_total - lam))
            Wq = (Lq / lam) * 60.0
            W[i, j] = Wq
```

```python

    plt.figure()
    plt.imshow(W, aspect='auto', origin='lower',
               extent=[min(pads_list)-0.5, max(pads_list)+0.5, min(arrivals)-5, max(arrivals)+5])
    plt.colorbar(label="Wq (min)")
    plt.title(f"Heatmap - Wq (pads × lambda) [{p.name}]")
    plt.xlabel("Number of pads (c)")
    plt.ylabel("Arrival lambda (pax/h)")
    plt.show()

# ---- analysis execution ----
def run_analysis_1(_):
    with out1:
        clear_output()
        p = ROUTES[rota_dd.value]
        pads = pads_step.value
        lam  = arrival_step.value
        over = over_step.value
        dep  = dep_per_pad(p.turnaround_min, over)
        pax_dep = int(p.pax_per_leg * p.load_factor)
        mmc = mmc_metrics(lam, dep, pax_dep, pads)

        print(f"[Analysis 1] Ground Supply - Route: {p.name}")
        print(f"Pads (c): {pads}")
        print(f"Departures per pad (dep/h): {dep:.2f}")
        print(f"Pax per departure: {pax_dep}")
        print(f"Utilization (p): {mmc['rho']:.2f}")
        print(f"Probability of waiting (Pc): {mmc['Pc']:.2f}")
        print(f"Average wait (Wq, min): {mmc['Wq_min']}")
        print(f"Total capacity (mu_total, pax/h): {mmc['mu_total_pax_h']:.1f}")

        # Heatmap only
        heatmap_wq(p, arrivals=[120,150,180,210,240,270,300],
                   pads_list=range(1,11), overhead_min=over)

next3.on_click(run_analysis_1)

display(w.VBox([
    w.HTML("<h3>Analysis 1 - Ground Supply (M/M/c)</h3>"),
```

```
    step1_box, step2_box, step3_box,
    out1
]))

```

A.4 City Effect

```
# CELL 4 - ANALYSIS 2: City Effect (wizard with pauses) + charts

# Steps:
# 1) Manually define served (pax/h)
# 2) (Optional) Adjust average car occupancy
# 3) Run analysis and charts

served_step = w.IntSlider(value=120, min=0, max=800, step=10, description='Served (pax/h):', continuous_update=False)
occ_step    = w.Dropdown(options=[1.2, 1.3, 1.4, 1.6], value=1.3, description='Car occup:')

nextA = w.Button(description='Next ->', button_style='info')
nextB = w.Button(description='Run analysis', button_style='success', icon='play')
backA = w.Button(description='<- Back', button_style='')

out2 = w.Output()

stepA_box = w.VBox([w.HTML("<b>Step 1/2 - Define served (pax/h)</b>"), served_step, nextA])
stepB_box = w.VBox([w.HTML("<b>Step 2/2 - Average car occupancy</b>"), occ_step, w.HBox([backA, nextB])])

stepB_box.layout.display = 'none'

def go_stepB(_):
    stepA_box.layout.display = 'none'
    stepB_box.layout.display = 'block'
def back_to_stepA(_):
    stepB_box.layout.display = 'none'
    stepA_box.layout.display = 'block'
```

```python

nextA.on_click(go_stepB)
backA.on_click(back_to_stepA)

def run_analysis_2(_):
    with out2:
        clear_output()
        p = ROUTES[rota_dd.value]
        # apply the chosen occupancy in this analysis (without changing
        ↪ the route's base data)
        p2 = RouteParams(**{**p.__dict__, "car_occupancy":
        ↪ float(occ_step.value)})

        cap_via = capacity_of_road(p2)
        excess  = excess_flow(p2)
        served  = served_step.value
        veh_removed_h = served / p2.car_occupancy
        pct_red = 0.0 if excess==0 else min(100.0, 100.0 * veh_removed_h /
        ↪ excess)

        print(f"[Analysis 2] City Effect - Route: {p2.name}")
        print(f"Road capacity (veh/h): {cap_via}")
        print(f"Peak demand (veh/h): {p2.demand_peak_veh_h}")
        print(f"Estimated excess (veh/h): {excess}")
        print(f"Served (pax/h): {served}")
        print(f"Vehicles removed (veh/h): {veh_removed_h:.1f}  (occupancy
        ↪ {p2.car_occupancy:.1f} pax/veh)")
        print(f"% of excess reduced: {pct_red:.2f}%")

        # Chart A: bars - capacity vs demand
        plt.figure()
        plt.bar([0,1], [cap_via, p2.demand_peak_veh_h])
        plt.xticks([0,1], ["Road capacity", "Peak demand"])
        plt.ylabel("veh/h")
        plt.title(f"Road - Capacity vs Demand [{p2.name}]")
        plt.grid(True, axis='y', alpha=0.3)
        plt.show()

        # Chart B: planning curve %excess vs served
        xs = np.arange(0, 801, 20)
        ys = []
```

```
        for s in xs:
            rem = s / p2.car_occupancy
            y = 0.0 if excess==0 else min(100.0, 100.0 * rem / excess)
            ys.append(y)
        plt.figure()
        plt.plot(xs, ys, marker='o')
        plt.title(f"Planning - %Excess reduced vs Served (pax/h)\n[{p2.name}]")
        plt.xlabel("Served (pax/h)")
        plt.ylabel("% of excess reduced")
        plt.grid(True, alpha=0.3)
        plt.show()

nextB.on_click(run_analysis_2)

display(w.VBox([
    w.HTML("<h3>Analysis 2 - City Effect</h3>"),
    stepA_box, stepB_box, out2
]))

```

A.5 Economic Feasibility

```
# CELL 5 - ANALYSIS 3: Economic Feasibility (wizard with pauses) + charts

# Steps:
# 1) Define unit cost (R$/pax)
# 2) Define wait SLA (min)
# 3) Define price range (min, max, points)
# 4) Define economic pads (capacity) and overhead
# 5) Run analysis and charts

cost_step   = w.Dropdown(options=[400.0, 650.0, 900.0], value=650.0, description='Cost R$/pax:')
sla_step    = w.Dropdown(options=[5.0, 7.5, 10.0, 12.5, 15.0], value=10.0, description='SLA (min):')
pmin_step   = w.IntSlider(value=300, min=100, max=1200, step=50, description='Min price:')
```

```python
pmax_step   = w.IntSlider(value=1500, min=800, max=3000, step=100, description='Max price:')
pnum_step   = w.IntSlider(value=25, min=5, max=60, step=5, description='# points:')
padsE_step  = w.Dropdown(options=[1,2,3,4,6], value=2, description='Pads (econ):')
overE_step  = w.Dropdown(options=[1.5,2.0,2.5,3.0], value=2.0, description='Overhead (min):')

nextE1 = w.Button(description='Next ->', button_style='info')
nextE2 = w.Button(description='Next ->', button_style='info')
nextE3 = w.Button(description='Next ->', button_style='info')
nextE4 = w.Button(description='Run analysis', button_style='success', icon='play')
backE2 = w.Button(description='<- Back', button_style='')
backE3 = w.Button(description='<- Back', button_style='')
backE4 = w.Button(description='<- Back', button_style='')

out3 = w.Output()

stepE1_box = w.VBox([w.HTML("<b>Step 1/4 - Unit cost (R$/pax)</b>"), cost_step, nextE1])
stepE2_box = w.VBox([w.HTML("<b>Step 2/4 - Wait SLA (min)</b>"), sla_step, w.HBox([backE2, nextE2])])
stepE3_box = w.VBox([w.HTML("<b>Step 3/4 - Price range (min, max, points)</b>"),
                     w.HBox([pmin_step, pmax_step, pnum_step]), w.HBox([backE3, nextE3])])
stepE4_box = w.VBox([w.HTML("<b>Step 4/4 - Capacity (economic pads) and overhead</b>"),
                     w.HBox([padsE_step, overE_step]), w.HBox([backE4, nextE4])])

# initial visibility
stepE2_box.layout.display = 'none'
stepE3_box.layout.display = 'none'
stepE4_box.layout.display = 'none'

def go_E2(_):
```

```
    stepE1_box.layout.display = 'none'; stepE2_box.layout.display =
    ↪ 'block'
def back_to_E1(_):
    stepE2_box.layout.display = 'none'; stepE1_box.layout.display =
    ↪ 'block'
def go_E3(_):
    stepE2_box.layout.display = 'none'; stepE3_box.layout.display =
    ↪ 'block'
def back_to_E2(_):
    stepE3_box.layout.display = 'none'; stepE2_box.layout.display =
    ↪ 'block'
def go_E4(_):
    stepE3_box.layout.display = 'none'; stepE4_box.layout.display =
    ↪ 'block'
def back_to_E3(_):
    stepE4_box.layout.display = 'none'; stepE3_box.layout.display =
    ↪ 'block'

nextE1.on_click(go_E2)
backE2.on_click(back_to_E1)
nextE2.on_click(go_E3)
backE3.on_click(back_to_E2)
nextE3.on_click(go_E4)
backE4.on_click(back_to_E3)

def dynamic_pricing_table(price_min_val, price_max_val, n_points,
↪ dep_per_pad_val, pax_per_dep):
    grid = np.linspace(price_min_val, price_max_val, n_points)
    rows = []
    for price in grid:
        demand = demand_at_price(price)
        headway_min = 60.0 / dep_per_pad_val
        expected_wait = headway_min / 2.0
        capacity_h = dep_per_pad_val * pax_per_dep
        served = min(demand, capacity_h)
        revenue = served * price
        rows.append((price, demand, served, revenue, expected_wait))
    rows.sort(key=lambda r: r[3], reverse=True)
    return rows, grid

def run_analysis_3(_):
```

```
    with out3:
        clear_output()
        p = ROUTES[rota_dd.value]

        unit_cost = float(cost_step.value)
        sla_wait  = float(sla_step.value)
        pmin, pmax, pnum = int(pmin_step.value), int(pmax_step.value), int(pnum_step.value)
        pads_econ, over_econ = int(padsE_step.value), float(overE_step.value)

        dep = dep_per_pad(p.turnaround_min, over_econ)      # departures/h per pad
        pax_dep = int(p.pax_per_leg * p.load_factor)        # pax per departure
        cost_per_departure = unit_cost * p.pax_per_leg

        rows, grid = dynamic_pricing_table(pmin, pmax, pnum, dep, pax_dep)
        best = None
        for r in rows:
            if r[4] <= sla_wait:
                best = r
                break
        if best is None:
            best = rows[0]   # best overall revenue if none meets the SLA

        best_price, demand, served, revenue, exp_wait = best

        mpm = margin_per_minute(
            price_per_pax=best_price,
            served_pax_h=served,
            cost_per_departure=cost_per_departure,
            departures_per_hour=dep,
            pad_minutes_used_per_hour=dep * p.turnaround_min
        )
        casm, rasm = casm_rasm(
            distance_km=p.distance_km_air,
            pax_per_departure=p.pax_per_leg,
            load_factor=p.load_factor,
            cost_per_departure=cost_per_departure,
            price_per_pax=best_price
```

```
        )

        print(f"[Analysis 3] Economic Feasibility - Route: {p.name}")
        print(f"Unit cost (R$/pax): {unit_cost:.2f}")
        print(f"Pads (econ): {pads_econ}  |  Overhead(min): {over_econ}  |  Turnaround(min): {p.turnaround_min}")
        print(f"Departures per pad (dep/h): {dep:.2f}  |  Pax/dep: {pax_dep}")
        print(f"Optimal price (R$): {best_price:.2f}")
        print(f"Demand at this price (pax/h): {demand:.1f}")
        print(f"Served (pax/h): {served:.1f}")
        print(f"Revenue/h (R$): {revenue:.2f}")
        print(f"Average wait (min): {exp_wait:.1f}  |  SLA (min): {sla_wait}  -> OK? {exp_wait <= sla_wait}")
        print(f"Margin per pad-minute (R$): {mpm:.2f}")
        print(f"CASM (R$/askm): {casm:.4f}  |  RASM (R$/askm): {rasm:.4f}")

        # Chart 1: Revenue vs Price
        revs = []
        for price in grid:
            demand_g = demand_at_price(price)
            served_g = min(demand_g, dep * pax_dep)
            revs.append(served_g * price)
        plt.figure()
        plt.plot(grid, revs)
        plt.title(f"Revenue vs Price - {p.name}")
        plt.xlabel("Price per passenger (R$)")
        plt.ylabel("Revenue per hour (R$)")
        plt.grid(True, alpha=0.3)
        plt.show()

        # Chart 2: Served vs Price
        servs = []
        for price in grid:
            demand_g = demand_at_price(price)
            servs.append(min(demand_g, dep * pax_dep))
        plt.figure()
        plt.plot(grid, servs)
        plt.title(f"Served vs Price - {p.name}")
        plt.xlabel("Price per passenger (R$)")
```

```python
        plt.ylabel("Served (pax/h)")
        plt.grid(True, alpha=0.3)
        plt.show()

        # Chart 3: CASM and RASM (bars)
        plt.figure()
        plt.bar([0,1], [casm, rasm])
        plt.xticks([0,1], ["CASM", "RASM"])
        plt.title(f"CASM vs RASM - {p.name} (Optimal price)")
        plt.ylabel("R$/askm")
        plt.grid(True, axis='y', alpha=0.3)
        plt.show()

nextE4.on_click(run_analysis_3)

display(w.VBox([
    w.HTML("<h3>Analysis 3 - Economic Feasibility</h3>"),
    stepE1_box, stepE2_box, stepE3_box, stepE4_box,
    out3
]))

```

A.6 Sensitivity Analysis

```python
# -*- coding: utf-8 -*-
import math
import numpy as np
import matplotlib.pyplot as plt

%matplotlib inline

# ============================================================
# Common core (base calculations) - used by A1, A2 and A3
# ============================================================

def dep_per_pad(turn_min, over_min):
    """Departures per pad per hour based on ground times."""
    return 60.0 / (turn_min + over_min)

```

```python
def mmc_metrics(lam_pax_h, dep_per_pad_h, pax_per_dep_eff, c, eps=1e-9):
    """
    M/M/c metrics for the boarding queue at the vertiport (robust to lambda=0).
    Returns: rho, Pc, Wq_min (min), mu_total (pax/h)
    """
    mu_pad   = dep_per_pad_h * pax_per_dep_eff
    mu_total = c * mu_pad

    # Degenerate case: no arrivals → no queue
    if lam_pax_h <= eps:
        return {"rho": 0.0, "Pc": 0.0, "Wq_min": 0.0, "mu_total": mu_total}

    # No capacity
    if mu_total <= eps:
        return {"rho": np.inf, "Pc": 1.0, "Wq_min": np.inf, "mu_total": mu_total}

    rho = lam_pax_h / mu_total
    # Unstable (or nearly)
    if rho >= 1.0 - eps:
        return {"rho": rho, "Pc": 1.0, "Wq_min": np.inf, "mu_total": mu_total}

    # Erlang-C formulas
    a = lam_pax_h / max(mu_pad, eps)
    s = sum((a**k)/math.factorial(k) for k in range(c))
    ec = (a**c)/math.factorial(c) * (c / max(c - a, eps))
    P0 = 1.0 / (s + ec)
    Pc = ec * P0
    Lq = Pc * (lam_pax_h / max(mu_total - lam_pax_h, eps))
    Wq_min = (Lq / lam_pax_h) * 60.0
    return {"rho": rho, "Pc": Pc, "Wq_min": Wq_min, "mu_total": mu_total}

def demand_linear(price, p0, p1, lam_at_p0, lam_at_p1):
    """Linear demand lambda(P) between p0 and p1, truncated to [0, lam_at_p0]."""
    if price <= p0:
        return lam_at_p0
    if price >= p1:
```

```
        return max(0.0, lam_at_p1)
    slope = (lam_at_p1 - lam_at_p0) / (p1 - p0)
    return lam_at_p0 + slope * (price - p0)

# ============================================================
# Analysis 1 - Sensitivities and charts
# ============================================================

def a1_sens_vs_pads(lambda_val, turn_min, over_min, pax_dep, lf,
↪   pads_grid=range(1, 11), sla_target=None):
    dep = dep_per_pad(turn_min, over_min)
    pax_eff = pax_dep * lf

    xs, rhos, wqs = [], [], []
    for c in pads_grid:
        m = mmc_metrics(lambda_val, dep, pax_eff, c)
        xs.append(c)
        rhos.append(m["rho"])
        wqs.append(m["Wq_min"] if m["rho"] < 1.0 else np.nan)

    plt.figure()
    plt.plot(xs, rhos, marker='o')
    plt.axhline(1.0, ls='--')
    plt.title(f"Utilization (p) vs pads | lambda={lambda_val:.0f} pax/h")
    plt.xlabel("Number of pads (c)")
    plt.ylabel("p")
    plt.grid(True, alpha=0.3)
    plt.show()

    plt.figure()
    plt.plot(xs, wqs, marker='o')
    if sla_target is not None:
        plt.axhline(sla_target, ls='--')
    plt.title(f"Average wait (Wq) vs pads | lambda={lambda_val:.0f}
    ↪   pax/h")
    plt.xlabel("Number of pads (c)")
    plt.ylabel("Wq (min)")
    plt.grid(True, alpha=0.3)
    plt.show()

```

```
def a1_sens_vs_lambda(c, turn_min, over_min, pax_dep, lf,
↪ lam_grid=np.arange(40, 240, 10), sla_target=None):
    dep = dep_per_pad(turn_min, over_min)
    pax_eff = pax_dep * lf

    xs, rhos, wqs = [], [], []
    for lam in lam_grid:
        m = mmc_metrics(lam, dep, pax_eff, c)
        xs.append(lam)
        rhos.append(m["rho"])
        wqs.append(m["Wq_min"] if m["rho"] < 1.0 else np.nan)

    plt.figure()
    plt.plot(xs, rhos, marker='o')
    plt.axhline(1.0, ls='--')
    plt.title(f"Utilization (p) vs demand | c={c}")
    plt.xlabel("lambda (pax/h)")
    plt.ylabel("p")
    plt.grid(True, alpha=0.3)
    plt.show()

    plt.figure()
    plt.plot(xs, wqs, marker='o')
    plt.title(f"Average wait (Wq) vs demand | c={c}")
    plt.xlabel("lambda (pax/h)")
    plt.ylabel("Wq (min)")
    plt.grid(True, alpha=0.3)
    plt.show()

# ============================================================
# Analysis 2 - Sensitivities and charts
# ============================================================

def a2_excesso(cap_via_veh_h, dem_pico_veh_h):
    return max(0.0, dem_pico_veh_h - cap_via_veh_h)

def a2_veh_removed(served_pax_h, occ_pax_per_veh):
    return served_pax_h / max(occ_pax_per_veh, 1e-6)

def a2_pct_relief(served_pax_h, occ_pax_per_veh, excesso_veh_h):
    if excesso_veh_h <= 0:
```

```
            return 100.0
        return (a2_veh_removed(served_pax_h, occ_pax_per_veh) / excesso_veh_h) * 100.0

def a2_sens_vs_served(excesso_veh_h, occ, served_grid=np.arange(0, 1200+1, 50)):
    xs = list(served_grid)
    ys = [a2_pct_relief(s, occ, excesso_veh_h) for s in xs]
    plt.figure()
    plt.plot(xs, ys, marker='o')
    plt.title("%Relief vs Served")
    plt.xlabel("Served (pax/h)")
    plt.ylabel("% of excess reduced")
    plt.grid(True, alpha=0.3)
    plt.show()

def a2_sens_vs_occup(served_pax_h, excesso_veh_h, occ_grid=np.arange(1.1, 2.6, 0.1)):
    xs = list(occ_grid)
    ys = [a2_pct_relief(served_pax_h, o, excesso_veh_h) for o in xs]
    plt.figure()
    plt.plot(xs, ys, marker='o')
    plt.title(f"%Relief vs Occupancy (Served={served_pax_h:.0f} pax/h)")
    plt.xlabel("Occupancy (pax/veh)")
    plt.ylabel("% of excess reduced")
    plt.grid(True, alpha=0.3)
    plt.show()

def a2_sens_vs_capacity(served_pax_h, occ, dem_pico_veh_h, cap_grid=np.arange(3800, 5200+1, 100)):
    xs, ys = [], []
    for cap in cap_grid:
        ex = a2_excesso(cap, dem_pico_veh_h)
        ys.append(a2_pct_relief(served_pax_h, occ, ex))
        xs.append(cap)
    plt.figure()
    plt.plot(xs, ys, marker='o')
    plt.title("%Relief vs Effective road capacity")
    plt.xlabel("Capacity (veh/h)")
    plt.ylabel("% of excess reduced")
    plt.grid(True, alpha=0.3)
```

```python
    plt.show()

# ============================================================
# Analysis 3 - Sensitivities and charts
# ============================================================

def a3_revenue_at_price(price, c, turn_min, over_min, pax_dep, lf, sla_target_min,
                        p0, p1, lam_at_p0, lam_at_p1, cost_per_pax,
                        dist_km, ask_seats):
    dep = dep_per_pad(turn_min, over_min)
    pax_eff = pax_dep * lf
    lam_price = demand_linear(price, p0, p1, lam_at_p0, lam_at_p1)
    m = mmc_metrics(lam_price, dep, pax_eff, c)
    served = min(lam_price, m["mu_total"])
    Wq = m["Wq_min"]
    sla_ok = (Wq <= sla_target_min) if np.isfinite(Wq) else False
    revenue_h = price * served
    dep_total_h = dep * c
    askm_h = ask_seats * dep_total_h * dist_km
    cost_h = cost_per_pax * served
    casm = cost_h / askm_h if askm_h > 0 else np.nan
    rasm = (price * served) / askm_h if askm_h > 0 else np.nan
    return {
        "price": price, "served": served, "revenue_h": revenue_h,
        "Wq_min": Wq, "sla_ok": sla_ok, "casm": casm, "rasm": rasm, "rho": m["rho"]
    }

def a3_sens_price_curve(c, turn_min, over_min, pax_dep, lf, sla_target_min,
                        p0, p1, lam_at_p0, lam_at_p1, cost_per_pax,
                        dist_km, ask_seats,
                        price_grid=np.linspace(300, 1500, 25)):
    rows = [a3_revenue_at_price(p, c, turn_min, over_min, pax_dep, lf, sla_target_min,
                                p0, p1, lam_at_p0, lam_at_p1, cost_per_pax,
                                dist_km, ask_seats)
            for p in price_grid]

```

```
    prices = [r["price"] for r in rows]
    revs   = [r["revenue_h"] for r in rows]
    serv   = [r["served"] for r in rows]

    plt.figure()
    plt.plot(prices, revs, marker='o')
    plt.title("Revenue/h vs Price")
    plt.xlabel("Price (R$)")
    plt.ylabel("Revenue per hour (R$)")
    plt.grid(True, alpha=0.3)
    plt.show()

    plt.figure()
    plt.plot(prices, serv, marker='o')
    plt.title("Served/h vs Price")
    plt.xlabel("Price (R$)")
    plt.ylabel("Served (pax/h)")
    plt.grid(True, alpha=0.3)
    plt.show()

    feasible = [r for r in rows if r["sla_ok"]]
    best = max(feasible, key=lambda x: x["revenue_h"]) if feasible else
    ↪ max(rows, key=lambda x: x["revenue_h"])

    plt.figure()
    plt.bar(["CASM", "RASM"], [best["casm"], best["rasm"]])
    plt.title("CASM vs RASM (at the optimal price)")
    plt.ylabel("R$/askm")
    plt.grid(True, axis='y', alpha=0.3)
    plt.show()

    return rows, best

# ============================================================
# Direct runs - already uncommented
# ============================================================

# ==== A1 ====
a1_sens_vs_pads(lambda_val=120, turn_min=8, over_min=1.5, pax_dep=4,
↪ lf=1.0, pads_grid=range(2, 9), sla_target=10)
```

```
a1_sens_vs_lambda(c=5, turn_min=8, over_min=1.5, pax_dep=4, lf=1.0, lam_grid=np.arange(60, 200, 10), sla_target=10)

# ==== A2 ====
excesso = a2_excesso(4500, 6000)
a2_sens_vs_served(excesso, occ=1.6, served_grid=np.arange(0, 1200+1, 50))
a2_sens_vs_occup(served_pax_h=400, excesso_veh_h=excesso, occ_grid=np.arange(1.1, 2.6, 0.1))
a2_sens_vs_capacity(served_pax_h=400, occ=1.4, dem_pico_veh_h=6000, cap_grid=np.arange(4200, 5200+1, 100))

# ==== A3 ====
rows, best = a3_sens_price_curve(
    c=3, turn_min=8, over_min=1.5, pax_dep=4, lf=1.0, sla_target_min=10,
    p0=300, p1=1500, lam_at_p0=60, lam_at_p1=0,
    cost_per_pax=650, dist_km=32, ask_seats=4,
    price_grid=np.linspace(300, 1500, 25)
)
print("Best price:", best["price"], "| Revenue/h:", best["revenue_h"],
      "| Served:", best["served"], "| Wq:", best["Wq_min"], "| SLA ok?", best["sla_ok"])

```

FOLHA DE REGISTRO DO DOCUMENTO

1. CLASSIFICAÇÃO/TIPO	2. DATA	3. REGISTRO N°	4. N° DE PÁGINAS
TC	13 de novembro de 2025	DCTA/ITA/TC-062/2025	70

5. TÍTULO E SUBTÍTULO:

Modelagem de uma Companhia Aérea de VTOL para Transporte Urbano na Cidade de São Paulo

6. AUTOR(ES):

Verônica Rodrighero

7. INSTITUIÇÃO(ÕES)/ÓRGÃO(S) INTERNO(S)/DIVISÃO(ÕES):

Instituto Tecnológico de Aeronáutica - ITA

8. PALAVRAS-CHAVE SUGERIDAS PELO AUTOR:

Mobilidade Aérea Urbana, eVTOL, Modelagem Operacional.

9.PALAVRAS-CHAVE RESULTANTES DE INDEXAÇÃO:

Aeronaves de decolagem vertical; Operações de linha aéreas; Mobilidade urbana; São Paulo (cidade); Custos operacionais; Modelagem (procesos); Simulação; Transporte aéreo; Engenharia aeroespacial; Transportes.

10. APRESENTAÇÃO: **(X) Nacional ( ) Internacional**

ITA, São José dos Campos. Curso de Graduação em Engenharia Aeroespacial. Orientador: Lucas Oliveira Barbacovi; coorientador: Christopher Shneider Cerqueira. Publicado em 2025.

11. RESUMO:

Este trabalho apresenta a modelagem operacional de uma companhia aérea urbana baseada em aeronaves VTOL, com foco nos eVTOLs, no contexto da cidade de São Paulo. A proposta surge como resposta à necessidade de soluções inovadoras e sustentáveis para o transporte aéreo de curta distância em grandes centros urbanos, especialmente diante dos desafios de congestionamento e ineficiência da mobilidade terrestre. A metodologia adotada fundamenta-se em um modelo paramétrico ajustável, capaz de simular diferentes cenários operacionais com base em variáveis como número de passageiros por dia, capacidade da aeronave, janelas de operação e estrutura de custos. A análise contempla a definição de rotas urbanas otimizadas, estimativa do tamanho da frota, cálculo de métricas econômicas (como receita, custo e lucro diário) e avaliação da taxa de utilização da frota. O modelo foi implementado em linguagem Python e calibrado com dados reais de mercado, inspirando-se na estrutura operacional da empresa Revo e nas projeções da Eve Air Mobility para eVTOLs elétricos. Os resultados das simulações indicam que esse tipo de operação pode ser viável técnica e financeiramente, desde que apoiada por uma infraestrutura adequada de vertiportos, integração com sistemas de controle de tráfego aéreo urbano e políticas de incentivo à mobilidade aérea sustentável.

12. GRAU DE SIGILO:

(X) OSTENSIVO ( ) RESERVADO ( ) SECRETO