eVTOL for Regional Travel in the State of Utah

UTAH STATE UNIVERSITY ASPIRE FOUNDATION

Introduction

In this study, we explore the possible future directions of electric aviation in the state of Utah. There currently exists many initiatives and new organizations with the goal of bringing electric aviation to the masses and integrating them into the transportation infrastructure. Nationally, initiatives such as the FAA's Electric Vertical Takeoff and Landing Integration Pilot Program create strategic partnerships with electric aviation companies such as BETA, Electra, and Joby [1]. Within the state of Utah, a $2M grant is being awarded to regional airports to facilitate infrastructure upgrades which would support aviation electrification [2]. With all that said, usage of electric aircraft in the Wasatch Front is extremely limited, and it is yet to be seen how these aircraft perform in high density altitude contexts. Additionally, FAA certification of electric vertical takeoff and landing (eVTOL) and other advanced air mobility (AAM) electric aircraft has not been completed, limiting the applications of current electric aircraft.

In this study, we take a simulation & modeling approach by utilizing the simulation tool X-Plane by Laminar Research to simulate flight paths which would likely be utilized in the aforementioned regional applications. Using X-Plane's advanced models of fixed and rotary-wing aircraft alongside accurate modeling of the battery technology and battery management system (BMS), we can accurately simulate possible future flight paths.

Beyond understanding what is possible with current technology, we would also like to look ahead and consider what becomes feasible as technology improves. One of the current primary bottlenecks for feasibility of electric aircraft is limitations in battery technology pertaining to energy density and energy capacity. Current batteries are heavy and not sufficiently dense for the majority of aircraft missions; however, promising advancements in battery chemistry could allow for denser and hence lighter batteries. These advancements would allow for expansion of applications which can be electrified. With these results, we intend

Figure 1: BETA ALIA-250 eVTOL flying over the Uinta Mountain Range in X-Plane.

to help policy makers and key players in Utah's aviation community in making informed decisions regarding incentivizing electrification in the right market segments and advancing infrastructure for aviation electrification in a way that maximizes utility.

1 Related Works

Previously, we published a feasibility study which explored various regional use cases, and determined which use cases are immediately feasible (assuming the aircraft and supporting infrastructure could be obtained). Using ADS-B data from flights in the state of Utah alongside metrics provided by Rocky Mountain Power of infrastructure available to airports, we determined instructional use cases to be a prime candidate for immediate electrification given the fixed wing single engine electric aircraft currently on the market. This conclusion was driven by the need to only electrify a single airport (a large number of instructional flights start and end at the same airport), the small payload of 1–2 passengers, and the generally short flights conducted in a flight training context. Other applications were also considered such as air-taxi, cargo transport, and recreation;

however, these were found to be largely not feasible due primarily to payload constraints, range limitations, lack of supporting infrastructure, and the energy density of present battery technologies. Mainstream adoption in these sectors will depend on demonstration of operational reliability, installation of supporting infrastructure, and economic viability [3].

In this study, we expand our previous work by further exploring both what is possible with current AAM, and what could be possible with projected advancements in technology.

2 Background

2.1 On Air-Taxis

Air-taxis introduce a short to medium range aircraft which is capable of transporting 1–5 passengers a relatively short distance. Generally, these are trips which could be taken by car, but would be orders of magnitude slower. For example, traveling from one side of Los Angeles to another during rush hour, or entering ski resorts such as Alta Resort inside of Little Cottonwood Canyon close to when the resort opens. Traffic in both of these scenarios can increase travel time from minutes to hours; however, an air-taxi would be able to circumvent the congestion and provide a direct path between two VTOL/heliport locations assuming eVTOL and other AAM aircraft are able to accommodate Utah's density altitudes.

It is important to point out that in the scenarios above, a ground transportation vehicle such as a car or bus is entirely feasible to service transportation. This means the air-taxi is providing a convenience as opposed to an opportunity which did not previously exist. Because this use case is squarely a luxury, it will be priced accordingly and will only be utilized by a small subset of the population. We can see this by looking at current air-taxi offerings.

BLADE Air Mobility is a ridesharing business currently operating in the New York City metro and in various locations in Europe. BLADE currently uses small/medium helicopters to facilitate their services. Some of their flagship routes include travel between their Manhattan lounge and the JFK airport, Manhattan to Westchester, or from Manhattan to several travel destinations including the Hamptons and Atlantic City. The service from their lounge and JFK represents a scenario which is somewhat analogous to anticipated scenarios in the SLC metro. These flights typically cost $250–$350 depending on the date and time. This high cost illustrates the intended market segment for this service alongside the evident willingness to pay for that market segment.

Some applications of air taxis which are more feasible due to more optimal environmental conditions include tourism applications in southern Utah such as flight tours/sightseeing opportunities, remote delivery via unmanned aircraft systems (UAS), and emergency support. Assuming the environmental constraints can be overcome in northern Utah, services such as travel between major cities, travel to recreation destinations such as ski resorts, and transport to airports are target use-cases for air-taxis.

2.2 Our Simulation & Modeling Environment

We use Laminar Research's X-Plane flight simulator to facilitate our simulation and modeling. This tool features very accurate modeling of aircraft enabled through blade element theory (BET). BET breaks the geometry of the aircraft into many small elements and then finds the physical forces on each element several times a second. This provides a much more nuanced and accurate simulation of flight when compared to naive implementations which use stability derivatives [4].

Beyond accurate flight dynamics, X-Plane also offers a digital twin of the BETA ALIA 250 eVTOL aircraft included with the simulation suite. This digital twin was created before the electronic hardware of the ALIA 250 was completed, so the battery modeling is not 100 % accurate to the real aircraft; however, X-Plane offers a software development kit (SDK) which enables near limitless expansion of the base simulator utilizing the Lua and C++ programming languages. This SDK can be used to extend the digital twin of the ALIA 250 enabling accurate battery dynamics, which we include in our implementation and use to drive our results. Alongside these native plug-in engines, X-Plane also supports advanced

Table 1: Battery Pack Specifications Provided by BETA

Parameter	Value
Nameplate Energy	45 kWh
Voltage	832 V max
Peak Power	400+ kW
Continuous Power	115 kW
Weight	225 kg (176 Wh/kg)

prototyping mechanisms such as hardware-in-the-loop evaluation and the ability to interface with external tools and software using common network protocols such as TCP and UDP. Other research institutions such as NASA [5] and MIT [6] use X-Plane to prototype new aircraft designs, controllers, and methodologies, further strengthening its reputation as a rigorous flight simulator. Additionally, X-Plane can be used in FAA approved flight simulators to gather hours towards pilot licenses [7].

We have developed a plug-in for X-Plane which enables flight paths to be carried out autonomously using eVTOL aircraft. Using this plugin, we are able to supply a departure and arrival location or a departure location alongside a flightplan supplied from SimBrief, and from there the simulator will initialize the environment with the aircraft at the arrival location, take off using the vertical lift motors, transition to using the push motor to execute the flight plan, and then complete a landing sequence transitioning back to using the lift motors. X-Plane publishes the statistics of the aircraft including battery usage throughout the flight, enabling detailed data collection which can be used to better understand the characteristics of various intended flight paths. Additionally, X-Plane features stochastic weather modeling capturing the nuances seen in real weather patterns. This means each route can be carried out under multiple typical weather conditions allowing our results to accurately reflect flying conditions experienced in northern Utah. We can utilize tools such as SimBrief [8] to automatically generate flight plans which will then be carried out by the aircraft.

As mentioned previously, we are also able to consider advancements in battery technology by modifying the battery modeling of the aircraft digital twin. Some examples of how we can mutate the properties of the battery include battery weight and battery capacity, which are our key concerns with regards to aviation electrification. With this, we can determine the approximate thresholds which advancements in battery will have to meet to achieve feasibility for our scenarios considered.

2.3 Battery Model

As mentioned previously, the battery modeling done by X-Plane is not accurate to the actual battery cells in the ALIA-250. To that end, we opted to model the battery dynamics external to the digital twin in our plugin. BETA has published some of their battery specifications which allows us to create a more realistic model than the model supplied in X-Plane by default.

Each ALIA-250 has 5 battery packs which are wired in parallel. When battery demand is created in the simulation environment, that demand is distributed evenly across all 5 cells. Demand for cell i is represented as

$$P_i = \frac{P_{\text{total}}}{5}$$

We track the following pieces of battery state for each pack:

We model the open circuit voltage (OCV) as a function of the SOC using a lookup table which describes the lithium-ion cell voltage at different SOCs. This acts as a discrete interpolation of the voltage behavior at the cell level. The OCV for a pack is then described as

$$V_{OCV,i} = N_{\text{cells}} \cdot V_{\text{cell}}(z)$$

Table 2: Relevant State Variables for Analysis

Symbol	Description
z	State of charge (SOC)
V_{RC}	Voltage across the RC polarization network
V_T	Terminal voltage

where $N_{\text{cells}} = 198$ for the ALIA-250. Next, we model the current using

$$I_i = \min\left(\frac{P_i}{V_{OCV}}, 500\right)$$

We use this information to then model a Thevenin RC network, which provides the polarization voltage that occurs from electrochemical diffusion. This is done using a first-order ODE

$$\frac{dV_{RC}}{dt} = -\frac{V_{RC}}{R_1 C_1} + \frac{I}{C_1}$$

To simulate the change in this function, we use forward Euler with timestep Δt

$$V_{RC}[k+1] = V_{RC}[k] + \Delta t\left(-\frac{V_{RC}}{R_1 C_1} + \frac{I}{C_1}\right)$$

where we define $R_1 C_1 = \tau = 0.001 \cdot 2500 = 2.5$. The terminal voltage V_T is then derived as

$$V_T = V_{OCV} - I \cdot R_0 - V_{RC}$$

To update a pack's SOC, we use

$$z_i[k+1] = z_i[k] - \frac{I_i \cdot \Delta t}{Q_{\text{nom}}}$$

Using this formulation, we are able to pass in present demand from the aircraft and determine the dynamics of the battery.

2.4 Autonomous Aircraft Control

For this study, it was important to automate the flights in simulation. This gains us two primary advantages: first, with the amount of data we wanted, it would not be feasible to manually complete all the flights; second, human error would make the data less generalizable, meaning the results would be biased towards the performance of the operator.[1] To this end, an end-to-end autonomous flight agent has been developed which utilizes simple rules-based autonomous flight enabled by the simulated flight computer alongside a controls-based landing controller which enables autonomous descent and landing of the aircraft. The kinematics of eVTOL is unlike both a conventional aircraft and a drone since the aircraft features both configurations at once. As such, special considerations must be made implementing the transitions between flight modes alongside integrating both control surfaces to leverage advantages of both.

2.4.1 Takeoff

The takeoff sequence of the aircraft is relatively simple. We hold the roll of the plane (θ) at zero, ensuring the plane remains level while setting the lift propellers to 100 % throttle. To help stabilize the plane, we also set the throttle on the push propeller to a low value (30 %). The plane ascends in place using the top propellers to get to altitude.

2.4.2 In-flight

Once the plane is in the air and at altitude, the autopilot is automatically programmed with a flight path which has been computed in advance. The plane will try to hold an altitude which is 1200 m above ground level (AGL). If the agent detects the plane is falling out of tolerance (i.e., $\epsilon = \mid a_{\text{des}} - a_{\text{actual}} \mid > 200$ feet AGL, where a is altitude), the agent will use the autopilot's tilt-nose feature to adjust the altitude until it is within tolerance again. This portion of the flight is handled almost entirely by the X-Plane X1000 flight computer which simulates a Garmin G1000.

[1]...and we are not pilots, so energy would be wasted on poor pilot performance.

Table 3: Phases of Landing Sequence

Phase	Description
LS_IDLE	Checks if the plane is within radius of the destination to begin landing procedure
LS_ENTRY	Disable the onboard autopilot, override the controls, trim the altitude and speed
LS_DECEL	Reduce the push motor, stabilize the altitude, begin introducing the lift props
LS_HOVER_CAPTURE	Reduce horizontal velocity to zero using the top props
LS_DESCEND	Reduce the throttle of the top props to descend plane
LS_FLARE	Increase throttle of top props to soften touchdown on ground
LS_TOUCHDOWN	Plane has now landed, stop all motors

2.4.3 Landing

This is the most involved part of autonomous flight. To facilitate landing of an eVTOL, we implemented two different forms of controllers. The first consists of two variants of PID-style controllers: one operates in `LS_DECEL` and both operate in `LS_HOVER_CAPTURE`. The PID controllers function by employing different gains and control surfaces throughout the landing process. The second is an incremental non-linear dynamic inversion (INDI) based cascade controller. This controller operates in `LS_DESCEND` and `LS_FLARE` through small corrections of the actuator command; the corrections are computed from the gap between the measured angular acceleration and the desired angular acceleration. We implement both controllers because the INDI controller is able to compensate for elevation-dependent variation, while the PID controller is a fixed-gain controller where an identical command produces distinct results at different elevations and air densities.

The ALIA-250 has control surfaces which are only valuable at certain points in time. The elevator (U_θ) and the aileron (U_ϕ) are good once the plane is at sufficient airspeed, but have no control authority while the plane is hovering. Meanwhile, the top propellers with states $\delta_p, \delta_r, \delta_y$ for pitch, roll, and yaw respectively have low control authority at speed, but complete control authority while hovering. The challenge with landing is using these control surfaces at the right time to ensure a smooth landing. We implement several PID controllers and INDI controllers to facilitate landing. The top propellers must be controlled to ensure adherence to desired pitch, roll, and yaw. We use the following control scheme for controlling throttle at each propeller:

$$\mathbf{\Delta} = \begin{bmatrix} +\delta_p - \delta_r - \delta_y \\ +\delta_p + \delta_r + \delta_y \\ -\delta_p - \delta_r + \delta_y \\ -\delta_p + \delta_r - \delta_y \end{bmatrix}$$

$$k = \text{clamp}\left(\min_i \begin{cases} \frac{1-T_{\text{base}}}{\Delta_i} & \Delta_i > 0 \\ \frac{-T_{\text{base}}}{\Delta_i} & \Delta_i < 0 \end{cases}, [0,1] \right)$$

$$\begin{bmatrix} T_{FR} \\ T_{FL} \\ T_{RR} \\ T_{RL} \end{bmatrix} = \text{clamp}\left(T_{\text{base}} + k\mathbf{\Delta}, [0,1]\right)$$

To control the pitch and roll while at speed, we use identical cascading proportional-integral-derivative (PID) controllers which are able to set the state of the aircraft to a desired value. The advantage of using a cascading PID is that we can have multiple loops all observing the aircraft states at different frequencies, helping to reduce instability and oscillations in state. For our controller, we use an outer loop which only focuses on the angle and an inner loop which exercises control using the rate of change of the angle. When designing a controller for an aircraft, this approach is common as the inner loop allows for quickly approaching the correct rate of change, while the outer loop can focus on converging to the correct value without having to worry about rotational inertia. The outer loop for our controller is then a PI controller with state and control:

$$e_\phi = \phi_{\text{des}} - \phi$$

$$\sigma_\phi[k+1] = \text{clamp}\left(\sigma_\phi[k] + e_\phi \Delta t, \pm 20\right)$$

$$P_{\text{cmd}} = \text{clamp}\left(K_{P,\text{ out}} \cdot e_\phi + K_{I,\text{ out}} \cdot \sigma_\phi, \pm 20\right)$$

The inner controller is purely proportional control, and is what finalizes control to the aileron:

$$P_{\text{err}} = P_{\text{cmd}} - P$$

$$u_\phi = \text{clamp}\left(K_{P,\text{ in}} \cdot P_{\text{err}}, \pm 1\right)$$

The controller for pitch is constructed identically, but with control over pitch through the elevator, u_θ, instead of roll. When in the `LS_HOVER_CAPTURE` phase of the flight, a similar PID scheme is followed for generating the T_i throttle values of each top prop, which is then adjusted using the formulas described in the throttle-mixer equations above.

The identical cascading PID controllers are used in the `LS_DECEL` and `LS_HOVER_CAPTURE` phases of the landing. When the aircraft enters the `LS_DESCEND` and `LS_FLARE` phases, we use the INDI-based cascade. The INDI-based control cascade continues to use an outer loop with the same role of computing the angle error into a desired rate, but the function of the inner loop is different from the PID logic:

$$P_{\text{des}}[k] = K_{\text{out}} \cdot e_\phi[k]$$

The INDI inner loop measures the aircraft's current angular acceleration directly from the simulation each frame; this can also be a self-correcting mechanism. This adjustment computes the incremental correction to close the gap between desired and measured acceleration, divided by a fitted control-effectiveness constant:

$$v_{\text{des}}[k] = \omega \cdot (P_{\text{des}}[k] - P[k])$$

$$\tilde{\delta}_r[k+1] = \tilde{\delta}_r[k] + \frac{v_{\text{des}}[k] - \dot{P}[k]}{G_{\text{roll}}}$$

Essentially, the INDI control logic in `LS_DESCEND` and `LS_FLARE` measures the angular acceleration instead of using a model. The INDI logic is only employed in `LS_DESCEND` and `LS_FLARE` because the aileron and elevator are zeroed out. The prop differential, or the actuator being commanded, is the sole source of measured angular acceleration.

The INDI control logic contains accumulators of both roll differential and pitch differential. Both accumulators have no built-in awareness of whether the motors are saturated and are able to deliver commands. The pseudo-control hedging (PCH) logic prevents the accumulators from staying pinned at values larger than what are actually being delivered, by carrying the previous frame's shortfall back to slow down the accumulator's growth. We have implemented PCH as an anti-windup strategy; without it we have to wind down the excess until saturation has cleared and behavior is sensible:

$$\epsilon_r[k] = \delta_r[k] - \hat{\delta}_r[k]$$

3 Methodology

3.1 Generating Flight Paths

In general, flying in a straight line between an arrival and departure location will not represent a feasible flight path due to restrictions in airways and changes in terrain. For example, flying between Cache-Logan Airport and Brigham City Regional Airport in a straight line would require flying directly over the Wellsville Mountains, requiring an altitude of ≈ 12000 feet. The ALIA-250 has optimal cruise at altitudes between 3000 and 8000 feet. The cabin of the plane is not pressurized, making altitudes above ≈ 14000 feet not safe.

To that end, we employ the use of SimBrief [8] to facilitate the creation of flight paths. SimBrief allows us to specify the parameters and limits of the ALIA-250, and then generates flight paths which are feasible for the ALIA-250 (see Figure 2). If a trip between two locations exceeds the capabilities of the ALIA-250, SimBrief will display a disclaimer showing that path is not feasible. We still collect the results of these flights which were flagged not feasible to gain a better understanding of the battery technology that would be needed to complete that flight. We considered the routes shown in Table 4 which serve as representative candidates for regional travel in northern Utah.

For each of these flights, we considered permutations of the weather which are powered by the stochastic weather modeling provided in X-Plane. This enables us to speak generally about

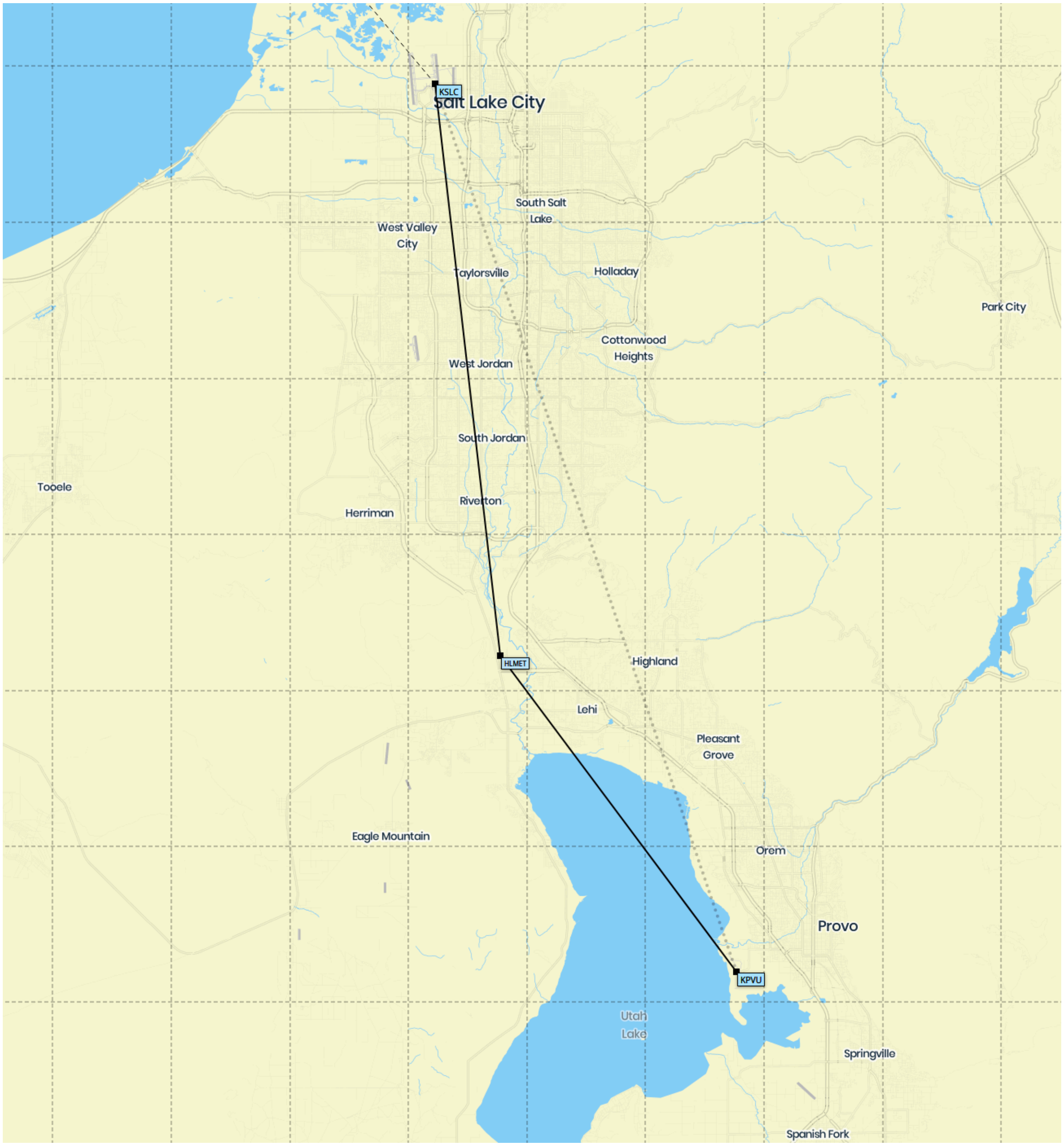

Figure 2: A flight path between Provo Regional Airport and SLC Intl. Airport generated by SimBrief.

the performance of these aircraft, and better understand how weather conditions impact the performance of the battery. We created weather presets for clear, partly cloudy, overcast, foggy, rainy, and thunderstorm weather conditions.

4 Results

Figure 3 outlines the result of all simulation runs considered. We ran a total of 157 simulation runs where 107 of them were successful, 40 crashed,

Table 4: Flight Paths Considered

#	Origin	Destination
1	KLGU (Logan-Cache)	K42U (Bear Lake County)
2	KLGU (Logan-Cache)	KSLC (Salt Lake City Intl)
3	KOGD (Ogden-Hinckley)	KSLC (Salt Lake City Intl)
4	KPVU (Provo Municipal)	KLGU (Logan-Cache)
5	KPVU (Provo Municipal)	KSLC (Salt Lake City Intl)
6	KSLC (Salt Lake City Intl)	KENV (Wendover)
7	KSLC (Salt Lake City Intl)	KLGU (Logan-Cache)
8	KSLC (Salt Lake City Intl)	KOGD (Ogden-Hinckley)
9	KPVU (Provo Municipal)	KPUC (Carbon Co. Regional)
10	KSLC (Salt Lake City Intl)	KPUC (Carbon Co. Regional)
11	KPUC (Carbon Co. Regional)	KPVU (Provo Municipal)
12	KPUC (Carbon Co. Regional)	KSLC (Salt Lake City Intl)
13	KPUC (Carbon Co. Regional)	KVEL (Vernal Regional)
14	KSLC (Salt Lake City Intl)	KVEL (Vernal Regional)
15	KSLC (Salt Lake City Intl)	KU34 (Green River Municipal)
16	KVEL (Vernal Regional)	KU34 (Green River Municipal)
17	KVEL (Vernal Regional)	KSLC (Salt Lake City Intl)
18	KPVU (Provo Municipal)	KHCR (Heber City)
19	KSLC (Salt Lake City Intl)	KHCR (Heber City)
20	KHCR (Heber City)	KSLC (Salt Lake City Intl)
21	KSLC (Salt Lake City Intl)	KBMC (Brigham City)
22	KOGD (Ogden-Hinckley)	KBMC (Brigham City)
23	KLGU (Logan-Cache)	KBMC (Brigham City)
24	KBMC (Brigham City)	KSLC (Salt Lake City Intl)
25	KBMC (Brigham City)	KOGD (Ogden-Hinckley)
26	KBMC (Brigham City)	KLGU (Logan-Cache)

6 aborted, and 4 truncated. Flights were truncated after 45 minutes of simulation time. Flights crashed as a result of either the battery dying due to the flight being longer than the range of the ALIA-250, or because the terrain exceeded the operating altitude of the aircraft. It is worth noting that the thunderstorm and rainy scenarios include high speed winds which in some scenarios caused the aircraft to lose control authority.

Looking at Figure 4 gives insights into the relationship between route distance and SOC. We can see the relationship between the two data points is not exactly linear due to the need to adjust the altitude to maintain a cruise AGL along-

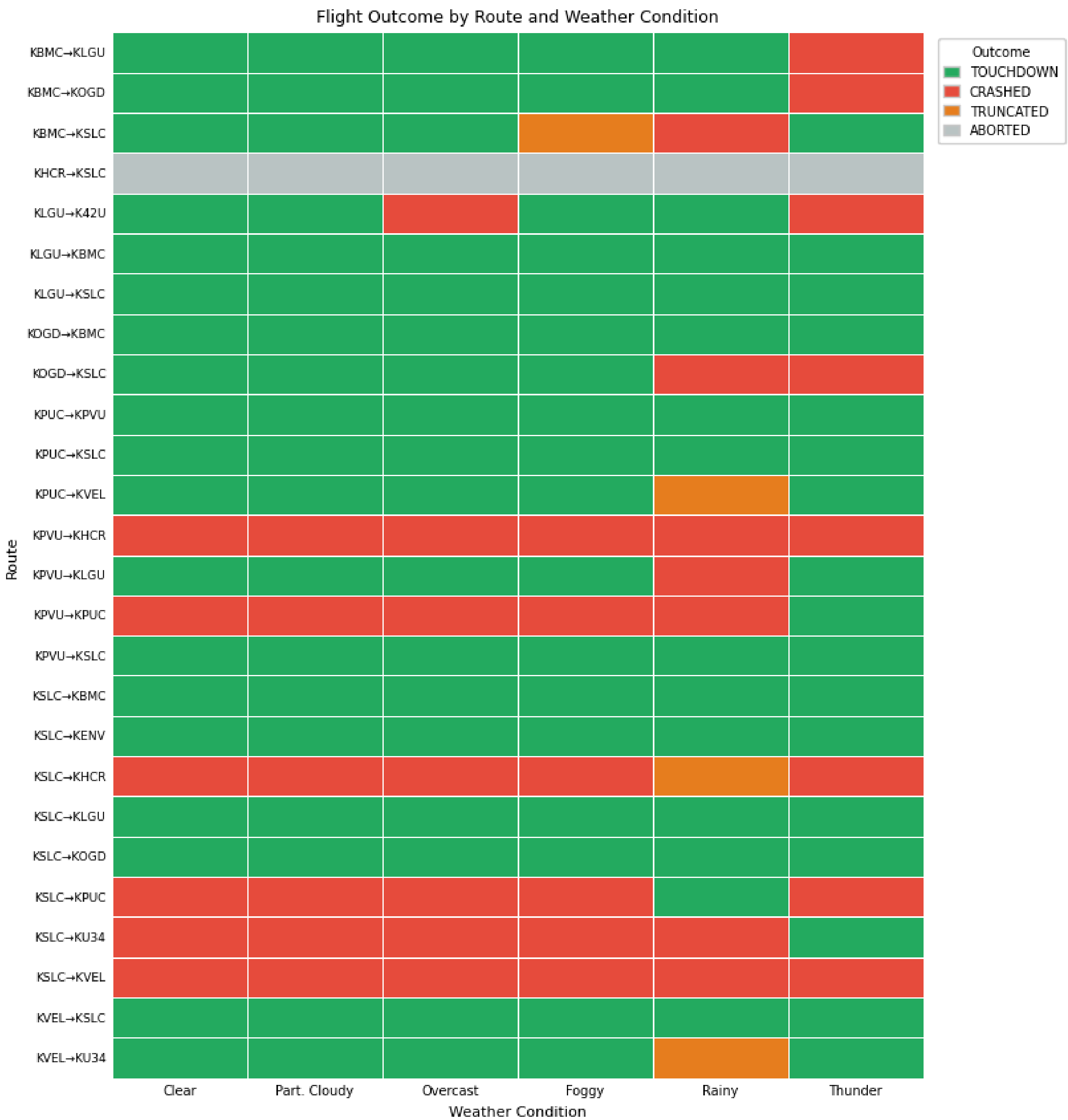

Figure 3: Result of all simulation runs considered. A flight being aborted, crashed, or truncated indicates the flight was not feasible.

side some locations requiring a more energy intensive takeoff and/or landing sequence. At around 100 km, the uncertainty around the route maintaining a safe battery margin increases significantly.

Battery utilization varied widely throughout simulation permutations. Provo to Salt Lake airports roughly follow a normal distribution, while Brigham City to Salt Lake was left skewed. Despite Brigham City being further from Salt Lake City than Provo, the distributions still have signif-

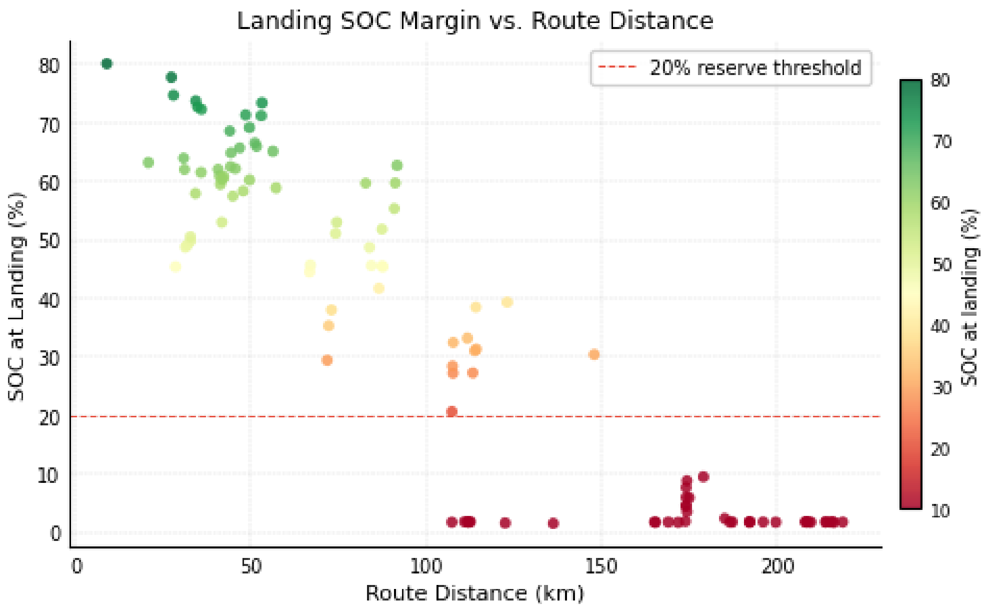

Figure 4: Flight distance versus SOC upon landing. A 20 % reserve is notated as a rough approximation of the required reserve power. Only flights which were completed successfully are considered.

icant overlap. Figure 6 reinforces this idea by showing that the route between Salt Lake City and Provo is more energy efficient compared to the Provo and Brigham City routes.

A noteworthy anomaly in Figure 6 is the route between Logan and Brigham City. This route is very inefficient due to the Wellsville mountains separating the cities. Completing this route requires either going over the Wellsville mountains or going around them. Both options require significant energy consumption and exemplify the challenges imposed on electric aircraft specifically for regional travel in Utah.

Power consumption can be better understood when segmenting consumption between the top motors which provide lift authority and the rear motor which provides push authority. As shown in Figure 7, the most energy intensive part of the flight is the takeoff sequence which draws ≈ 1.2 MW. Once the plane is in flight, it only needs to use the rear motor which on average consumes around ≈ 250 kW. Minimizing the time spent taking off and landing is then important to decrease energy usage and increase range.

5 Discussion

5.1 Topographical Constraints

The topographical constraints imposed on the aircraft's flight paths are not obvious from the battery analysis. As mentioned previously, the ALIA-250 does not have a pressurized cabin. This limits the altitude to ≈ 14000 feet to maintain sufficient oxygen in the cabin. Because of this, the aircraft's ability to fly over the mountains is extremely limited. The resulting effect is that many paths increase in length significantly due to the need to fly around the mountains, as shown in Figure 8. Planning flights for an eVTOL is a delicate balance between the constraints imposed on the aircraft with regards to altitude and range.

5.2 Flight Paths Considered

X-Plane has a limitation in that flight paths must be between two airports or heliports registered in its database. As a result, flights to prospective locations such as ski resorts and other popular venues were not considered. The flights considered in this study represent representative candidates and are not necessarily the flight paths

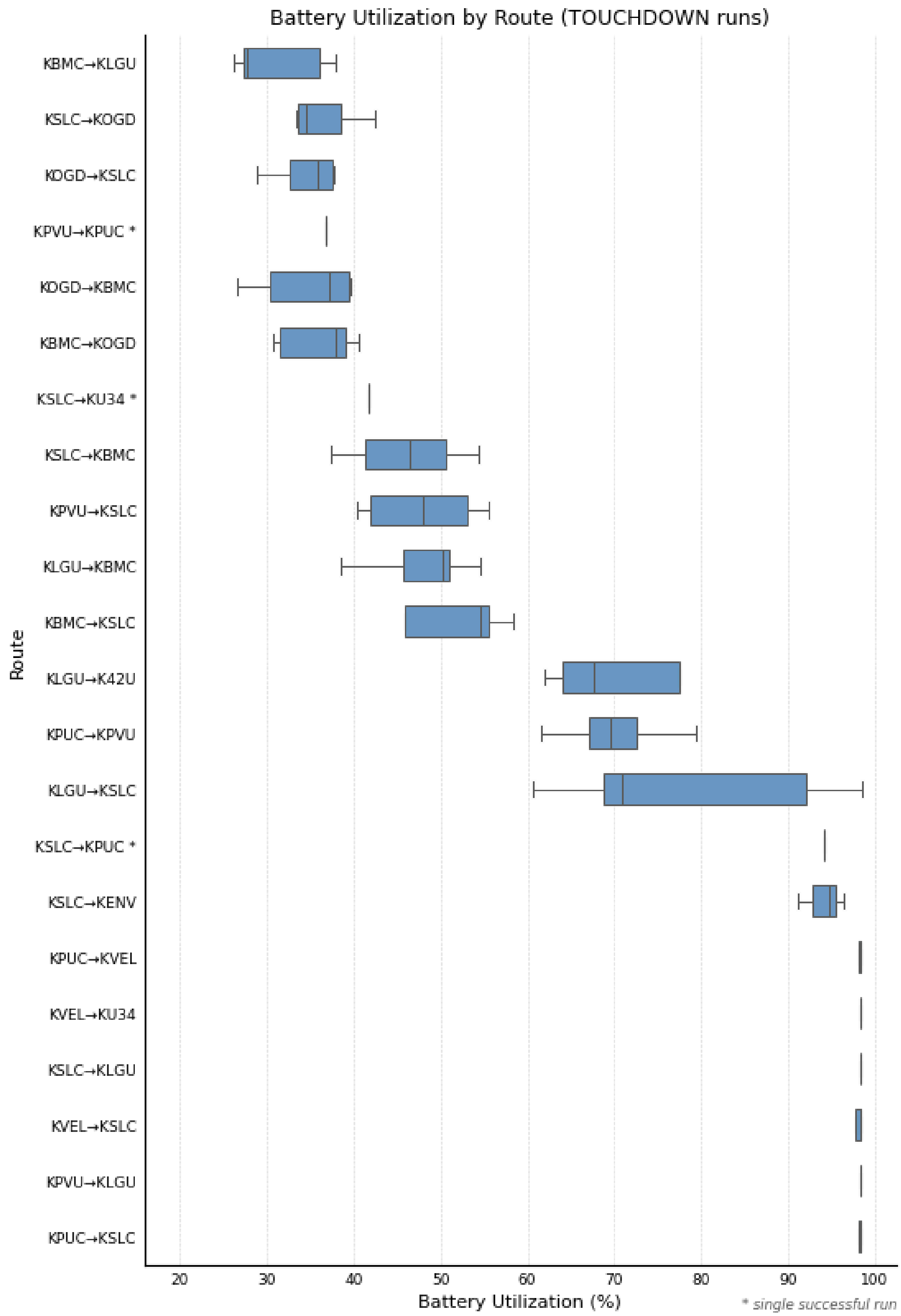

Figure 5: Battery utilization across all simulations per route. An asterisk by the route label indicates only a single data point had a successful run.

which are expected to be popular with electric aircraft.

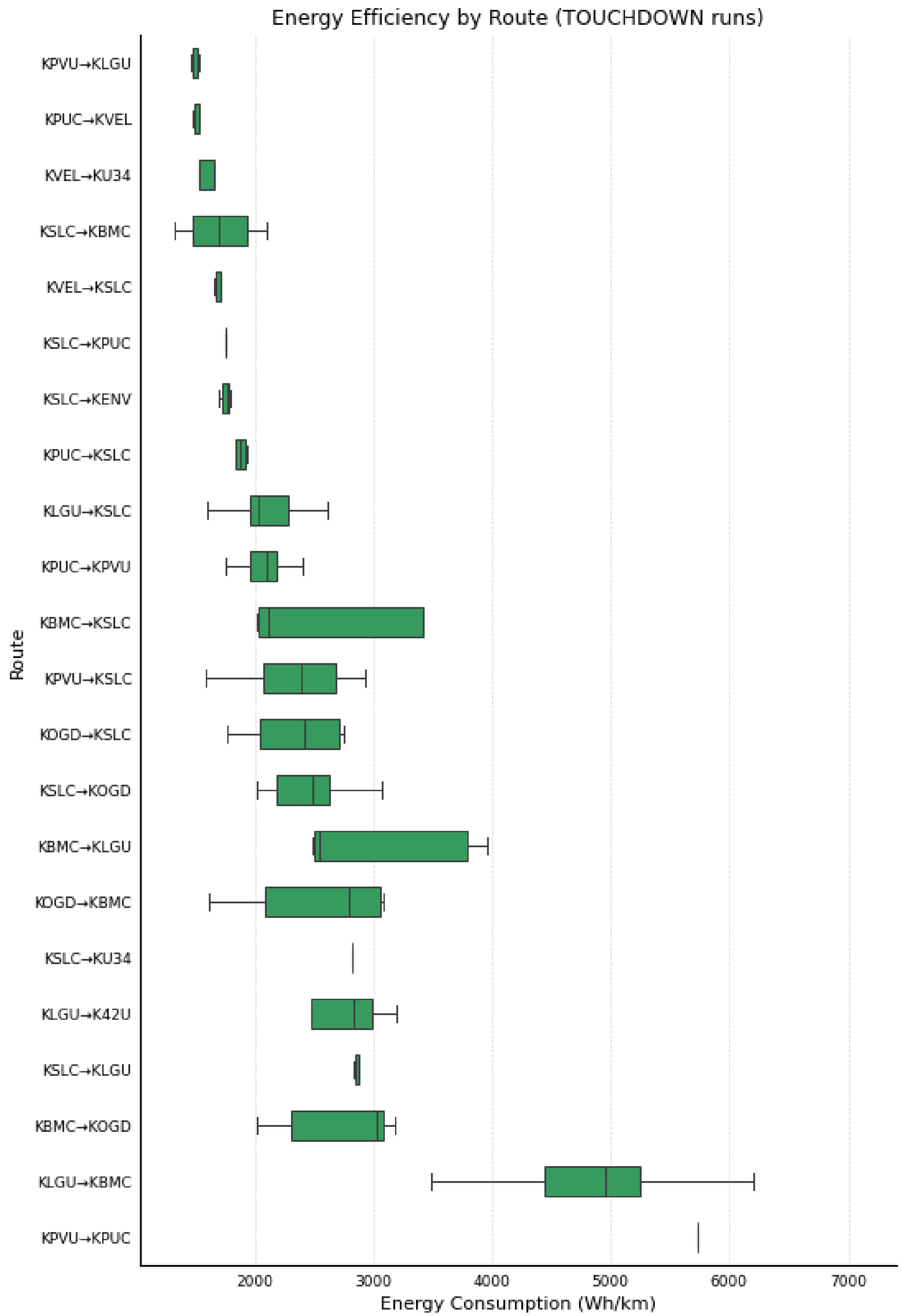

Figure 6: Energy consumption.

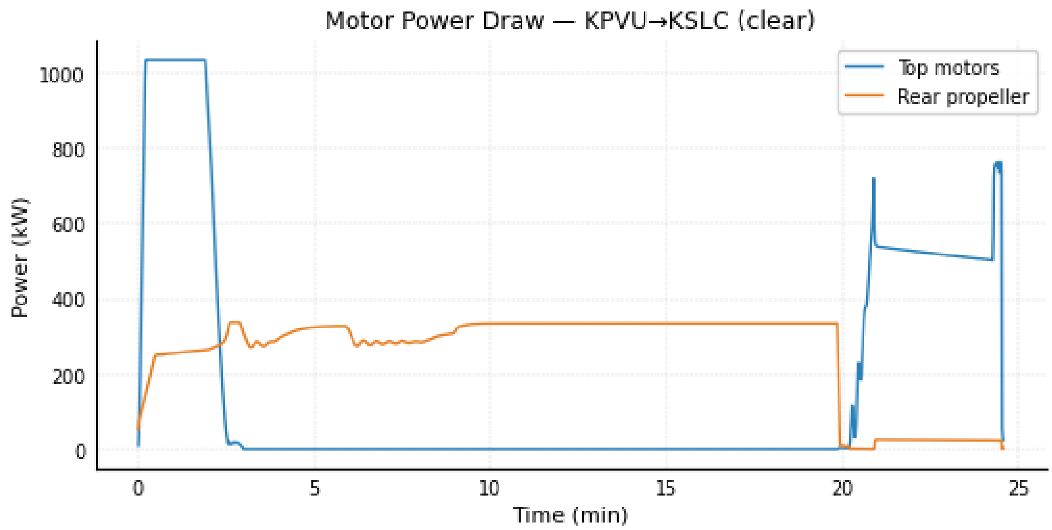

Figure 7: Power draw throughout flight split between the top lift motors and the rear push motor.

5.3 Energy Constraints

A large constraint which emerges from the results is the range of the ALIA-250 imposed by its battery. The ALIA-250 uses state of the art batteries which represent some of the highest energy density we can safely obtain with current technology. Despite this, the constraints are still very limiting in terms of feasible flight paths. Since we cannot increase the altitude of the aircraft to fly over terrain, the better option is to increase the density of energy in the battery.

6 Conclusion

In this paper we evaluate the performance of the BETA ALIA-250 eVTOL, using this aircraft as a representative candidate for advanced air mobility (AAM) aircraft at large. To facilitate this study, we create an automation pipeline which allows for end-to-end autonomous flight. Using this pipeline, we are able to simulate many regional travel scenarios providing insights into the possible use-cases for AAMs in Utah.

We find that the ALIA-250 is capable of completing many regional flights, but the limitations are clear. The ALIA thrived in flat terrain and was regularly able to complete flights of this type with reserve battery left over. Flying over mountains and other terrain proved challenging due to the control constraints of the aircraft alongside the limited range imposed by current battery technology. While the technology is promising, overall more research and development of battery technologies is pivotal to improving feasibility of these aircraft in the Mountain West region.

7 References

1. U.S. FAA, “The future of aviation is here: Trump’s transportation secretary Sean P. Duffy and FAA unveil eight selections for pilot program testing next-gen aircraft in America’s skies.” https://www.faa.gov/newsroom/future-aviation-here-trumps-transportation-secretary-sean-p-duffy-and-faa-unveil-eight. Accessed: 2026-03-09.
2. “Airport revitalization grant — UDOT.” https://connect.udot.utah.gov/aboutus/operations/aeronautics/aeronautics-grants-2-0/airport-revitalization-grant/,

   2025.
3. C. Justice, K. Goodrick, and M. Harper, “Considering feasibility of electric aircraft at a regional scale,” in *2025 IEEE 28th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 66–71, IEEE, 2025.
4. L. R. Ribeiro and N. M. F. Oliveira, “UAV autopilot controllers test platform using MATLAB/Simulink and X-Plane,” in *2010 IEEE Frontiers in Education Conference (FIE)*, pp. S2H-1, IEEE, 2010.

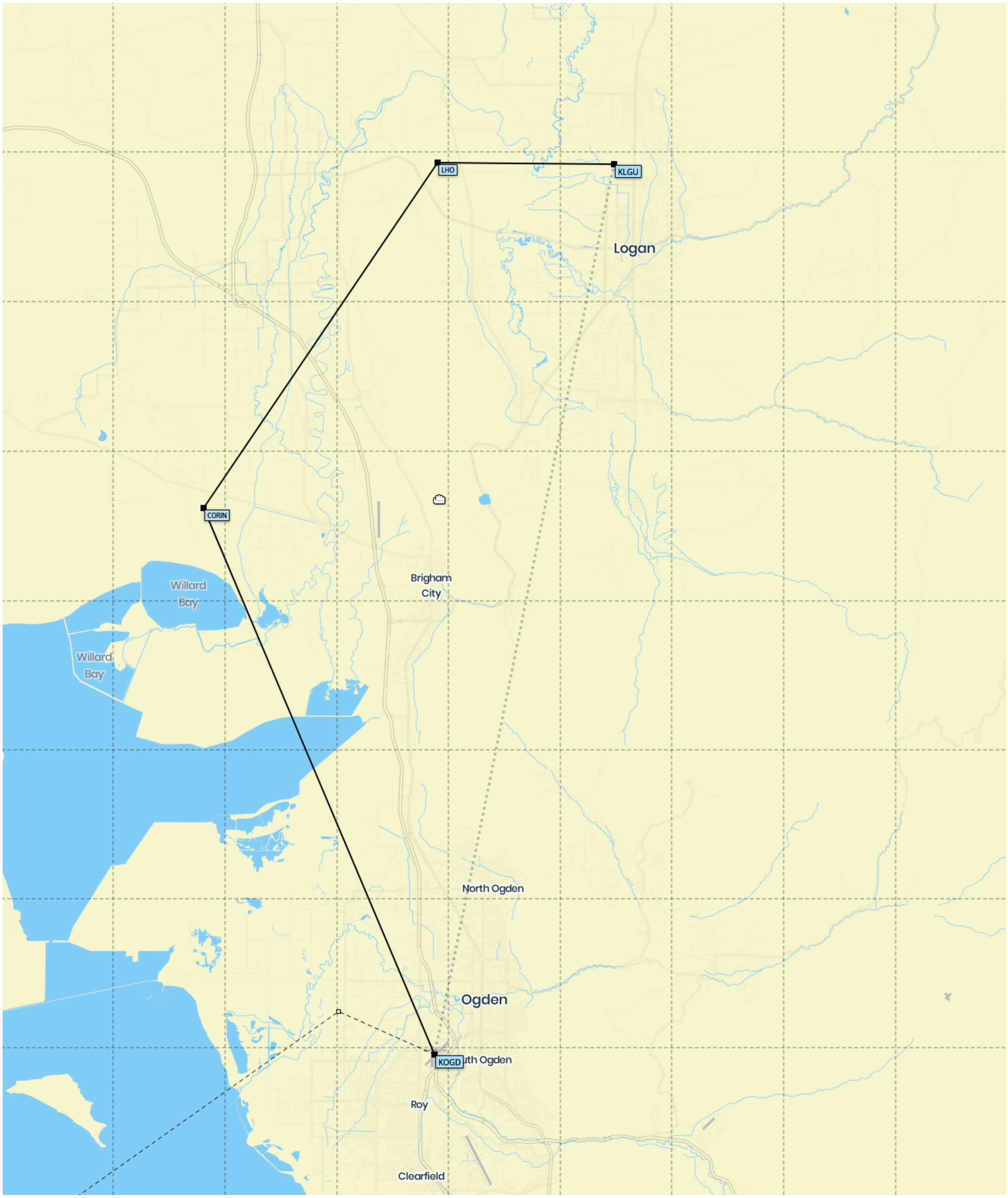

Figure 8: Flight path between KLGU and KOGD. The path required adds 30 % additional distance over the straight line path.

5. J. S. Litt, T. S. Sowers, H. Buescher, and R. Jansen, “Implementation approach for an electrified aircraft concept vehicle in a research flight simulator,” in *AIAA SCITECH 2022 Forum*, p. 2306, 2022.
6. A. Kaviyarasu, A. Saravanakumar, and M. L. Venkatesh, “Hardware in loop simulation of a way point navigation using MATLAB/Simulink and X-Plane simulator,” in *2019 International Conference on Intelligent Sustainable Systems (ICISS)*, pp. 332–335, IEEE, 2019.
7. X-Plane, “Certified.” https://www.x-plane.com/pro/certified/.
8. SimBrief. https://www.simbrief.com/home/.