Capacity and Throughput of Urban Air Mobility Vertiports with a First-Come, First-Served Vertiport Scheduling Algorithm

Nelson Guerreiro, George Hagen, Jeffrey Maddalon
NASA Langley Research Center
&
Ricky Butler
AMA, Inc.

AIAA Aviation Forum, 15-19 June 2020
Outline

• Background and Motivation

• Vertiport Scheduler

• Capacity and Throughput Analysis Design
  – Theoretical Model
  – Simulation Model
  – Queueing Model
  – Demand Scenario

• Results
Background/Motivation
Background

• NASA’s Air Traffic Management – Exploration (ATM-X) project is investigating impact of new entrants to the National Airspace System (NAS)

• The Urban Air Mobility (UAM) sub-project is investigating the services, functions, concepts, and architectures required to enable some of these new operations

• A vertiport scheduling algorithm (Vertiport Scheduler) was developed to support fast-time simulations and analysis of UAM demand scenarios

• The objectives of this study were to:
  – Understand the expected capacity of different vertiport configurations by developing a theoretical capacity model
  – Study the Vertiport Scheduler’s ability to achieve the expected capacity by comparing the capacity to the throughput within a simulated UAM demand scenario
Motivation

• Vertiports:
  – Are the designated landing and takeoff locations for UAM operations
  – Can be characterized by the number of vertipads (takeoff and landing areas) and the number of parking spaces (gates)
  – Can have many different surface layouts

• Vertiports are the sources and sinks for UAM operations and, as such, we need to account for these constraints when analyzing UAM airspace operations

• A simple model for vertiports, and an associated vertiport scheduling algorithm, was needed and developed for fast-time simulation

• This study helps us understand the performance of this simple modeling approach and scheduling algorithm
Vertiport Scheduler
Vertiport Scheduler

- Each vertiport is defined by the number of vertipads and number of parking spaces
- Each vertipad or parking space is modeled by a resource timeline
- The Vertiport Scheduler manages reservations (time blocks) on these timelines
- A typical UAM vehicle operation at a vertiport will have:
  - A vertipad reservation for the landing/arrival operation
  - A parking space reservation for the surface time at the vertiport
  - A vertipad reservation for the takeoff/departure operation
Vertiport Scheduler (2)

• The Vertiport Scheduler algorithm provides interfaces for external planners to:
  – Query the availability of the vertiport resources (next available arrival, next available departure)
  – Make reservations at the vertiport (reserve arrival, reserve departure)

• Other Vertiport Scheduler assumptions:
  – Reservations cannot overlap in time for the same resource
  – Arrivals, departures, and parking reservations are linked to a vehicle identifier (aircraft tail number)
  – Reservations have start and end times that provide a continuous presence on the vertiport resources
  – All surface time at a vertiport is allocated to a parking space timeline (e.g., taxi, passenger loading/un-loading)
  – Initial implementation uses first-come, first-served (FCFS) logic
Analysis Design and Models
Study Approach

• Develop a theoretical model for capacity

• Evaluate the throughput of different vertiport configurations using:
  – A queueing model
  – A simulation model

• Compare the throughput to the expected capacity
Definitions

**Operation(s)** – an operation is equivalent to a single reservation on a vertipad or a single reservation at a parking space.

**Capacity** – the estimated number of operations on a resource over a given time period, given some assumptions (e.g., vertipad capacity, surface capacity, vertiport capacity).

**Throughput** – the observed number of operations on a resource over a given time period (e.g., vertiport throughput).

**Vertipad Usage** – the percentage of time used by vertipad reservations, from the total available time of all vertipads at a vertiport, over a given time window.

**Space Usage** – the percentage of time used by parking reservations, from the total available time of all parking spaces at a vertiport, over a given time window.

**Space-Limited Vertiport** – a vertiport where the limiting factor for capacity is the number of parking spaces (e.g., the vertipads are able to support higher capacity).

**Vertipad-Limited Vertiport** – a vertiport where the limiting factor for capacity is the number of vertipads (e.g., the parking spaces are able to support higher capacity).
Theoretical Capacity Model

• Definitions:

\( N_s \) - number of parking spaces
\( N_p \) - number of vertipads
\( t_{\text{arr}} \) - arrival time block
\( t_{\text{dep}} \) - departure time block
\( t_{\text{surf}} \) - surface time block
\( t_{\text{win}} \) - time window

• Equations:

Surface Capacity:
\[
C_{\text{surf}} = N_s \cdot \frac{t_{\text{win}}}{t_{\text{surf}}} \rightarrow \text{surface operations}
\]

Vertipad Capacity:
\[
C_{\text{pads}} = 2 \cdot N_p \cdot \frac{t_{\text{win}}}{t_{\text{arr}} + t_{\text{dep}}} \rightarrow \text{vertipad operations}
\]

Vertiport Capacity:
\[
C_{\text{port}} = \min(2 \cdot C_{\text{surf}}, C_{\text{pads}}) \rightarrow \text{vertiport operations}
\]
• Theoretical model helps define the boundary between surface limited and vertipad limited for a vertiport
  – A vertiport is considered to be *surface capacity limited* when the following is true:

\[ N_s < N_p \cdot \frac{t_{surf}}{t_{arr} + t_{dep}} \]

• Model assumptions:
  – Vertiport arrivals and departures are balanced over any given period of time
  – One surface operation is equivalent to two vertipad operations
  – The vertipad usage is 100% for vertipad capacity formula
  – The surface usage is 100% for surface capacity
  – Maximum capacity is calculated using the minimum surface time
  – Nominal capacity is calculated using the average surface time
Queueing Model

- An individual Vertiport Scheduler instance is used
- A large queue of demand is used to fill the Vertiport Scheduler at the next available time
- Requires an assumed surface time distribution
- For each element in the queue and while the vertiport has availability within a desired time block:
  - The next available arrival is queried/identified
  - An arrival reservation is made, along with a surface reservation (initially of infinite duration)
  - A finite surface time is sampled from the distribution
  - The next available departure is queried/identified
  - A departure reservation is made (and the parking reservation’s end time is adjusted)

Queue = \{operation1, operation2, ...\}

Vertiport Scheduler Timelines
Simulation Model

• A scenario analysis tool called ATS-TIGAR was used
  – Uses a mission planning algorithm to plan UAM trips within an airspace region
    • Realistic constraints include:
      – Limited vehicle fleet with fleet management (e.g., tail tracking, passenger load/un-load times)
      – Vertiports with limited resources (parking spaces and vertipads)
      – Trips with different number of passengers
      – 4-D vertical takeoff and landing trajectories for each flight
      – Airspace constraints (disabled for this analysis)
      – Multiple operators sharing resources
  – Implements a Vertiport Scheduler for each vertiport in the scenario
• A single demand scenario with multiple vertiports was run for this study
UAM Demand Scenario

- UAM demand scenario was computed by Virginia Tech for NASA using a mode choice model
- 25,472 daily one-way multi-passenger commuter trips for the Dallas area between 102 vertiports
  - 51,632 total flights planned after including repositioning/clearing flights

Trip Time-of-Day Distribution

Trip Size Distribution
• Vertiport Locations (DF4 is the largest and busiest vertiport)

Background map source: Google Maps
UAM Demand Scenario (3)

• DF2 and DF4 have 6 vertipads
UAM Demand Scenario (4)

- DF4 has 38 parking spaces
### Study Parameters/Assumptions

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{\text{arr}} )</td>
<td>60 [s]</td>
</tr>
<tr>
<td>( t_{\text{dep}} )</td>
<td>60 [s]</td>
</tr>
<tr>
<td>( t_{\text{surf, min}} )</td>
<td>120 [s]; represents taxi in plus taxi out times</td>
</tr>
<tr>
<td>( t_{\text{surf, avg}} )</td>
<td>Dependent variable; 15 [min] used for comparison and discussion in some examples</td>
</tr>
<tr>
<td>( N_p )</td>
<td>Varies by vertiport (min: 1, max: 6)</td>
</tr>
<tr>
<td>( N_s )</td>
<td>Varies by vertiport (min: 4, max: 38)</td>
</tr>
<tr>
<td>( t_{\text{win}} )</td>
<td>15 [min]</td>
</tr>
</tbody>
</table>

- **Number of Vertiports**: 102
- **Number of Vehicles**: 1,190
- **Passenger Loading Time (per Passenger)**: 60 [s]
- **Passenger Unloading Time (per Passenger)**: 60 [s]
- **Vehicle Charge Time**: 0 [s]
- **Vehicle Model**: Generic VTOL model with cruise speed of 130 [kts]

- Two UAM operators with 75/25% split for trips and vehicle fleet
Results
Estimating Required Parking Spaces

• How many parking spaces are required to not be surface capacity limited?

• This is dependent on the surface time spent at a vertiport

• Figure shows trend using two assumed average surface times (2-minute trend represents the limit for our analysis)
Surface vs. Vertipad Limited Vertiports

- Plateau regions show vertipad limited vertiport
- Sloped regions indicate surface limited vertiport

\[ t_{surf} = 2 \text{ minutes} \]

\[ t_{surf} = 15 \text{ minutes} \]
DF4 Configuration Throughput

- **DF4:**
  - 6 vertipads
  - 38 parking spaces

- **Queueing model assuming uniform** surface time 2-28 minutes (15 avg.)
  - Expect surface limited

- Maximum capacity = 90 ops./15-min.
- Nominal Capacity = 76 ops./15-min.
- Observed throughput average ~ 63.9 ops./15-min.
- Vertipad usage ~ 75.9%
- Space usage ~ 90.5% (some peaks close to 100%)
Limitations on Throughput

- Why do we not see 100% surface usage in the previous queueing model run for a surface limited vertiport?
- Partial timeline slots can be left behind on the vertipad or parking space timelines due to random surface times and arrival/departure times.
- This can be expected in a first-come, first-served scheduler, or one where little to no optimization is used.
- Note that an on-demand mode may make optimization difficult.
Observed Surface Distribution

- What if the surface distribution were not uniform?
- Observed surface distribution at DF4 (average surface time = 9.4 minutes) from the simulation model run
DF4 Throughput - Observed Surf. Time Dist.

- **DF4:**
  - 6 vertipads
  - 38 parking spaces
- **Queueing model using observed surface time distribution (9.4 min. avg.):**
  - NOW we expect vertipad limited
- Maximum capacity = 90 ops./15-min.
- Nominal Capacity = 90 ops./15-min.
- Observed throughput average ~ 70.5 ops./15-min. (~22% less than capacity)
- Vertipad usage ~ 83.8%
- Space usage ~ 60.9%
DF4 Demand in Simulation Model

- Un-impeded demand at DF4 exceeds expected capacity in morning and afternoon peaks
- Un-impeded demand includes the repositioning/clearing flights
The demand peaks are reduced by delays due to vertiport availability.

Simulation model produces the observed surface time distribution (9.4 min. avg.)
- Vertiport is vertipad limited.

Maximum capacity = 90 ops./15-min.
Nominal Capacity = 90 ops./15-min.

Observed peak throughput is 70 ops./15-min. (~22% less than capacity)
Vertipad peak usage is 83.0%
Space peak usage ~ 52% after 7 A.M.

Multiple factors contribute to reduced throughput compared to queueing model (e.g., fleet constraints, other end vertiport constraints)
All Vertiports – Simulation Model

• All vertiports had average surface times less than 15 minutes
• Peak throughputs show an increasing loss of efficiency as the number of vertipads increases
Conclusions
Conclusions

• Investigated the capacity and throughput of vertiports under different configurations using the Vertiport Scheduler algorithm
  – Developed theoretical model that provided:
    • A way to estimate the required parking spaces for a vertiport
    • A way to estimate if a vertiport configuration is surface or vertipad capacity limited
  – Analyzed the throughput of vertiport configurations using
    • Queueing model
    • Simulation model
  – Results showed:
    • Resources can be under-utilized due to FCFS or non-optimized scheduling
    • A multi-vertiport simulation model can reveal throughput dependencies not observed in the single vertiport queueing model

• Next Steps:
  – Consider the observed resource usage percentages in future scenarios and analyses
  – Explore alternative algorithms for vertiport scheduling
Thank You

Nelson Guerreiro
nelson.m.guerreiro@nasa.gov