CS6380 Project: Large-scale UAS Traffic Management

Tom Henderson et al.

SCHOOL OF COMPUTING

THE UNIVERSITY OF UTAH

Acknowledgment

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-17-1-0077

(DDDAS-based Geospatial Intelligence)

Colleagues

David Sacharny

Tom Henderson

Robert Simmons

Taylor Welker

Xiuyi Fan
Nanyang Technological University

Amar Mitiche
INRS Montreal

Futuristic Vision

(Slide from Jared Esselman; UDOT)

SCHOOL OF COMPUTING

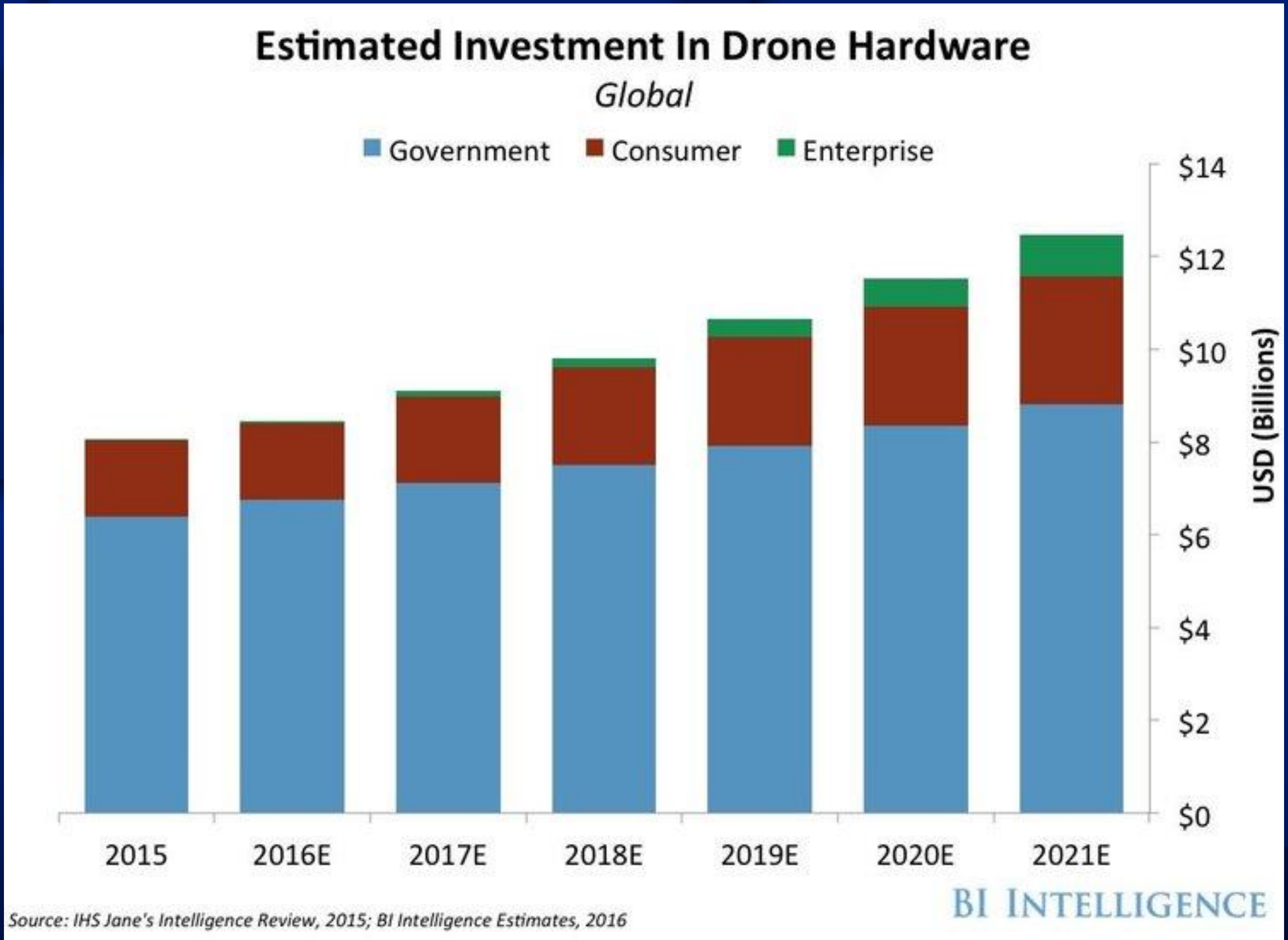
Commercial Use Cases

- 3D Mapping, Video Collection
- Delivery (Amazon, etc.)
- Inspections
- Data (Re)Transmission
- Air Taxis

➔Investment 2017: $506M

➔1000's of flights per day

Drone HW Investment ($B)

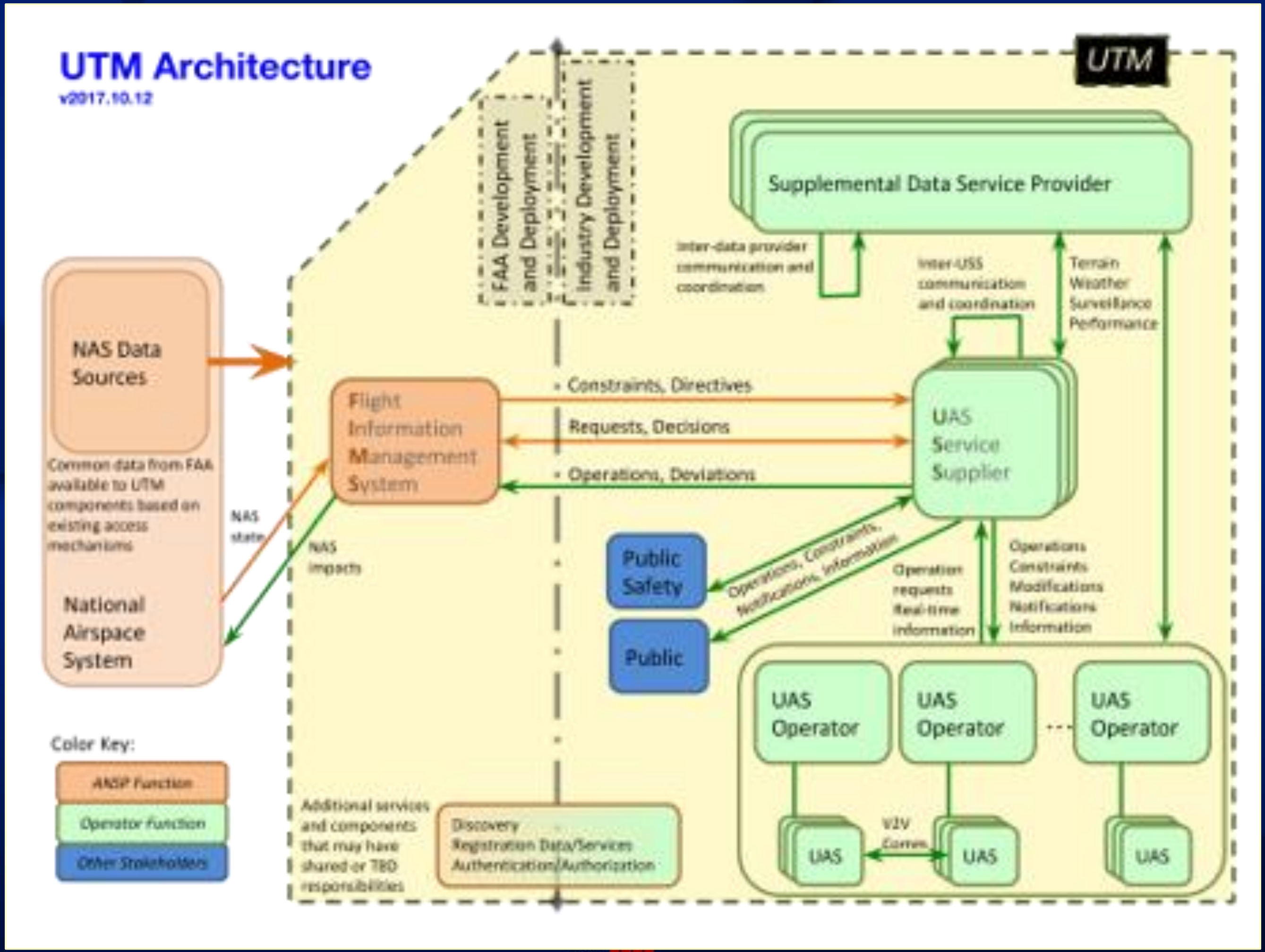


SCHOOL OF COMPUTING

Concerned Communities

- FAA/NASA
- Local & State Governments
- Service Vendors
- Users
- Public

FAA/NASA Architecture

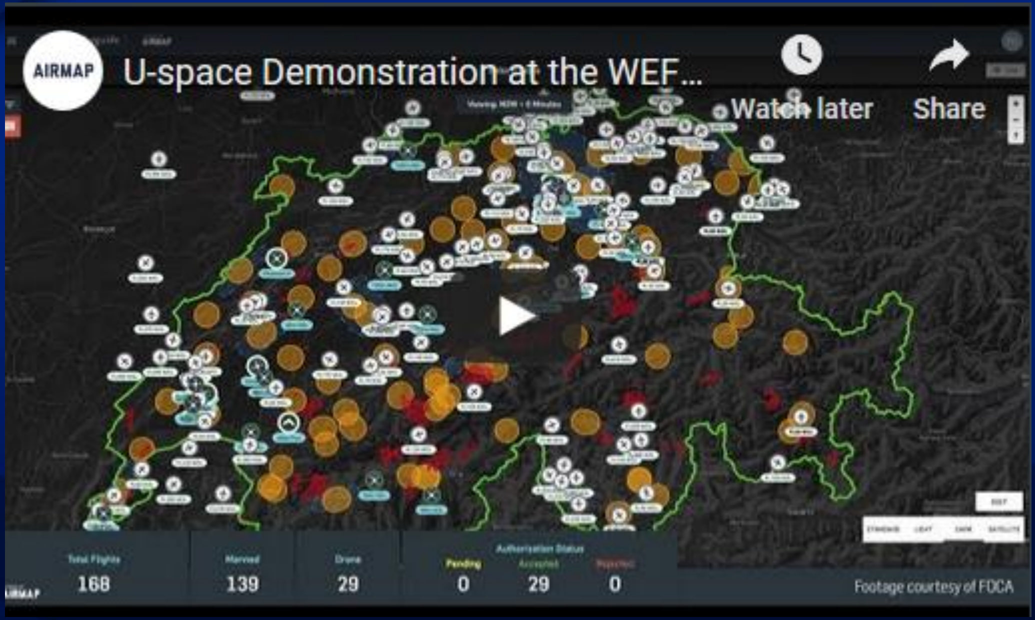


Local & State Governments

- Regulation of Flight
- Infrastructure for flight management
  - Radar
  - 5G
  - RTK GPS
  - Emergency Service
  - Data Acquisition & Analysis

Service Vendors

- Low cost
- Efficiency
- E.g.: Airmap



Users (Flying UAS's)

- Low Cost
- Privacy
- Reliability

SCHOOL OF COMPUTING

General Public

- Positive Aspects:
  - Lower cost of goods
  - Lower cost of delivery
  - Improve delivery time
- Negative Aspects:
  - Noise
  - Pollution
  - Privacy

What’s Wrong Currently

- Service Suppliers must share flight plans
- Arbitrary flight plans ➔ negative public
- Safety levels & prediction difficult
- Arbitrary paths NP-space hard

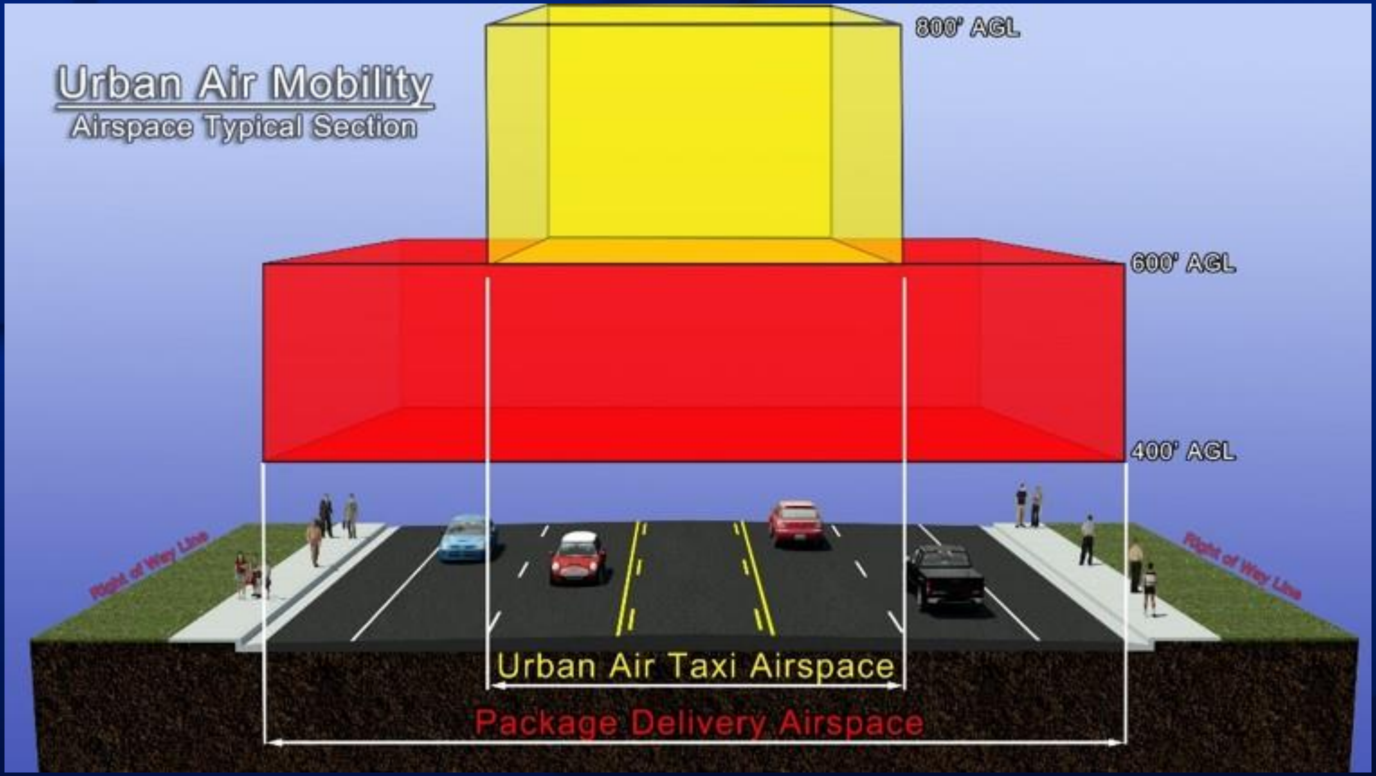
SCHOOL OF COMPUTING

Our Proposal

- Lane-based airways
  - UAS configuration space becomes 1D
  - Allows strategic deconfliction
- Airway roundabouts for intersections
  - No crossing; just merging or diverging
- Computationally tractable

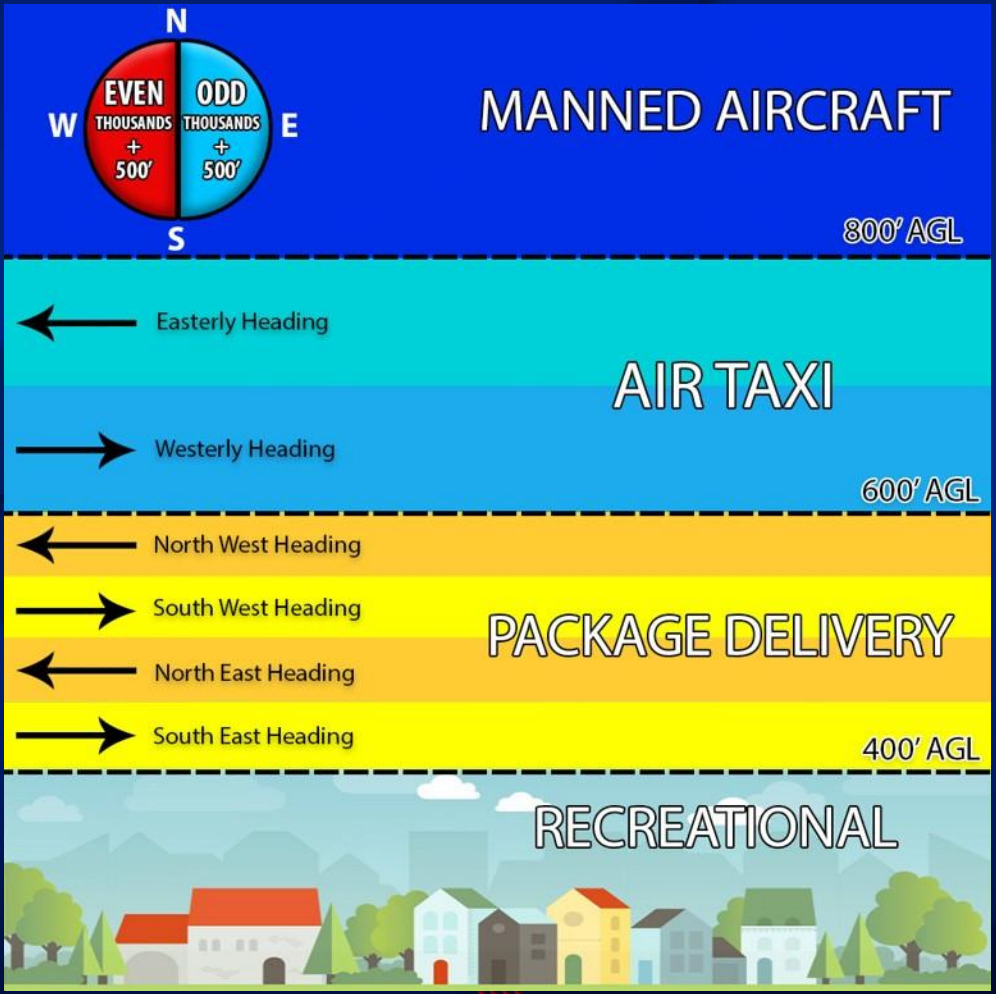
Utah Urban Air Mobility Idea

(Slide from Jared Esselman; UDOT)



UDOT UAM (cont'd)

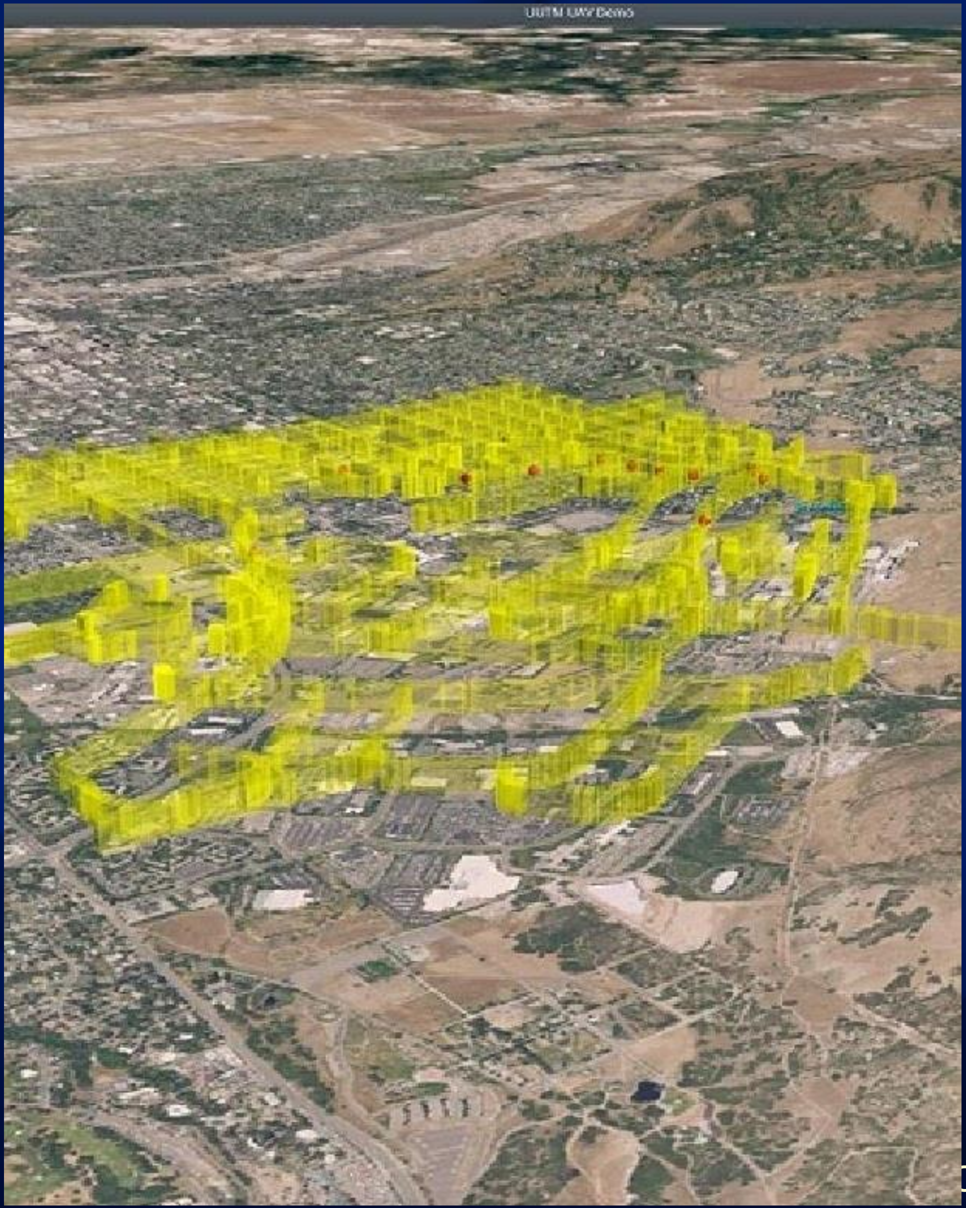
(Slide from Jared Esselman; UDOT)



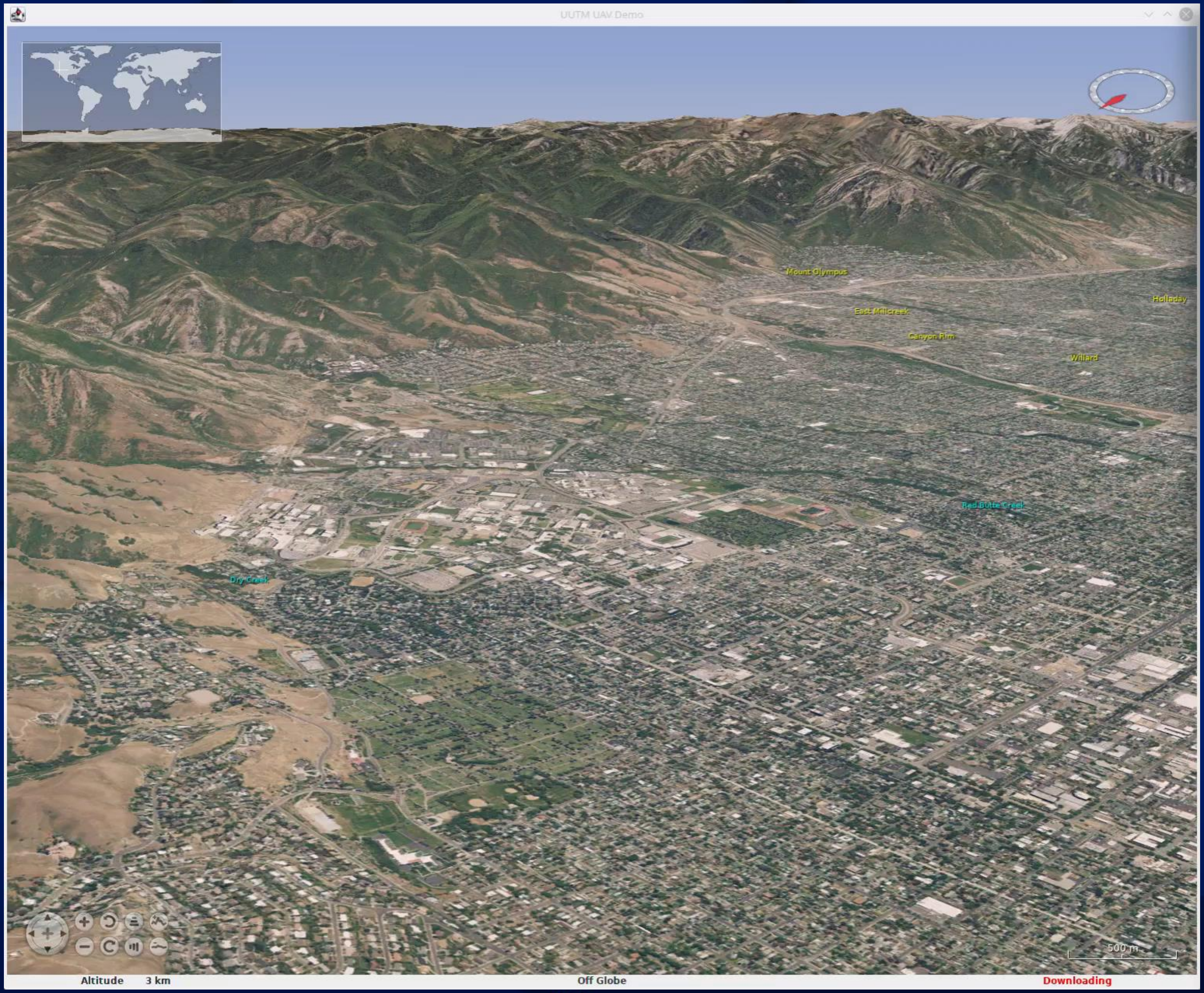
UDOT UAM (cont'd)

Proposal:

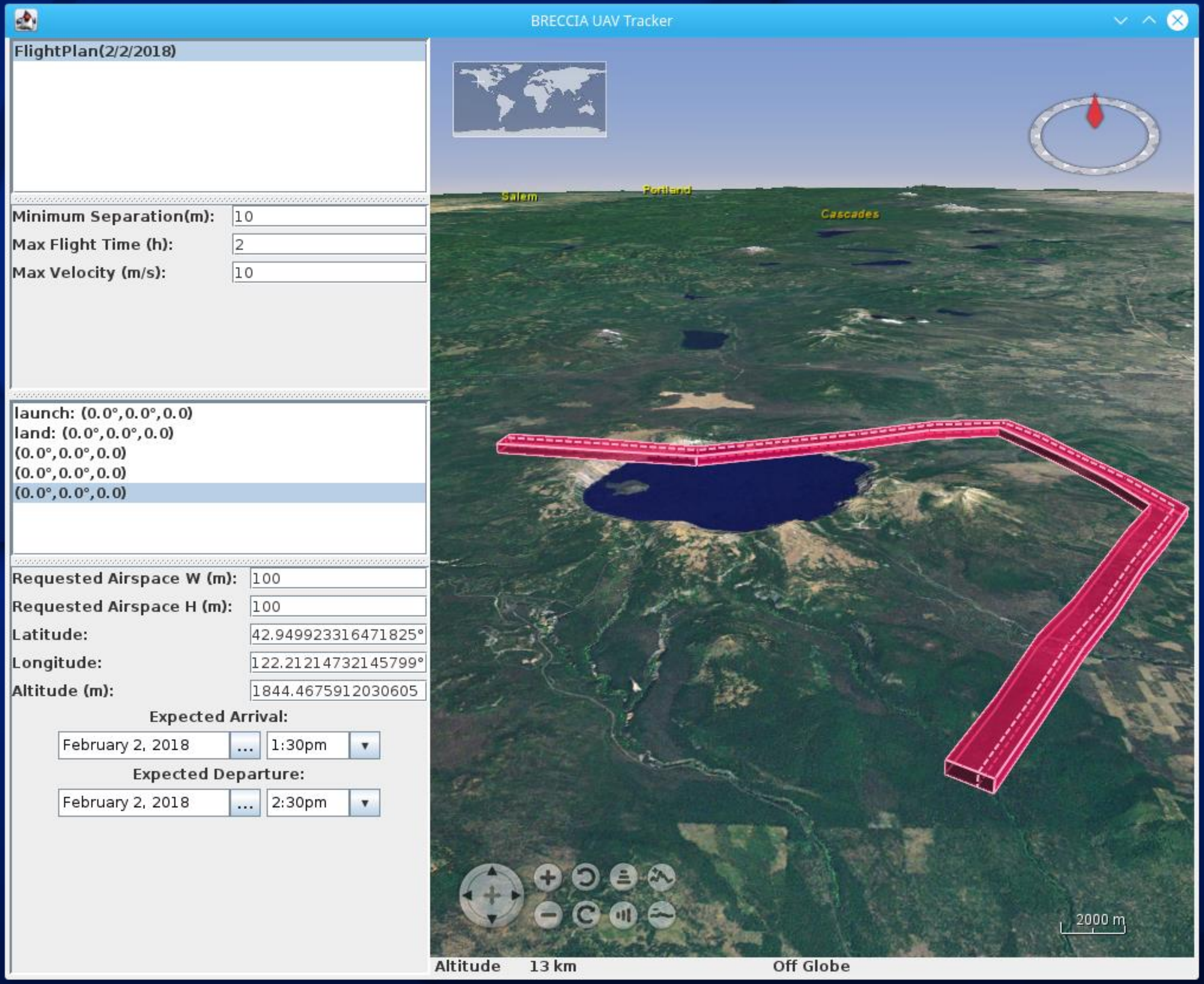
Airways above roadways.



Creating Airways



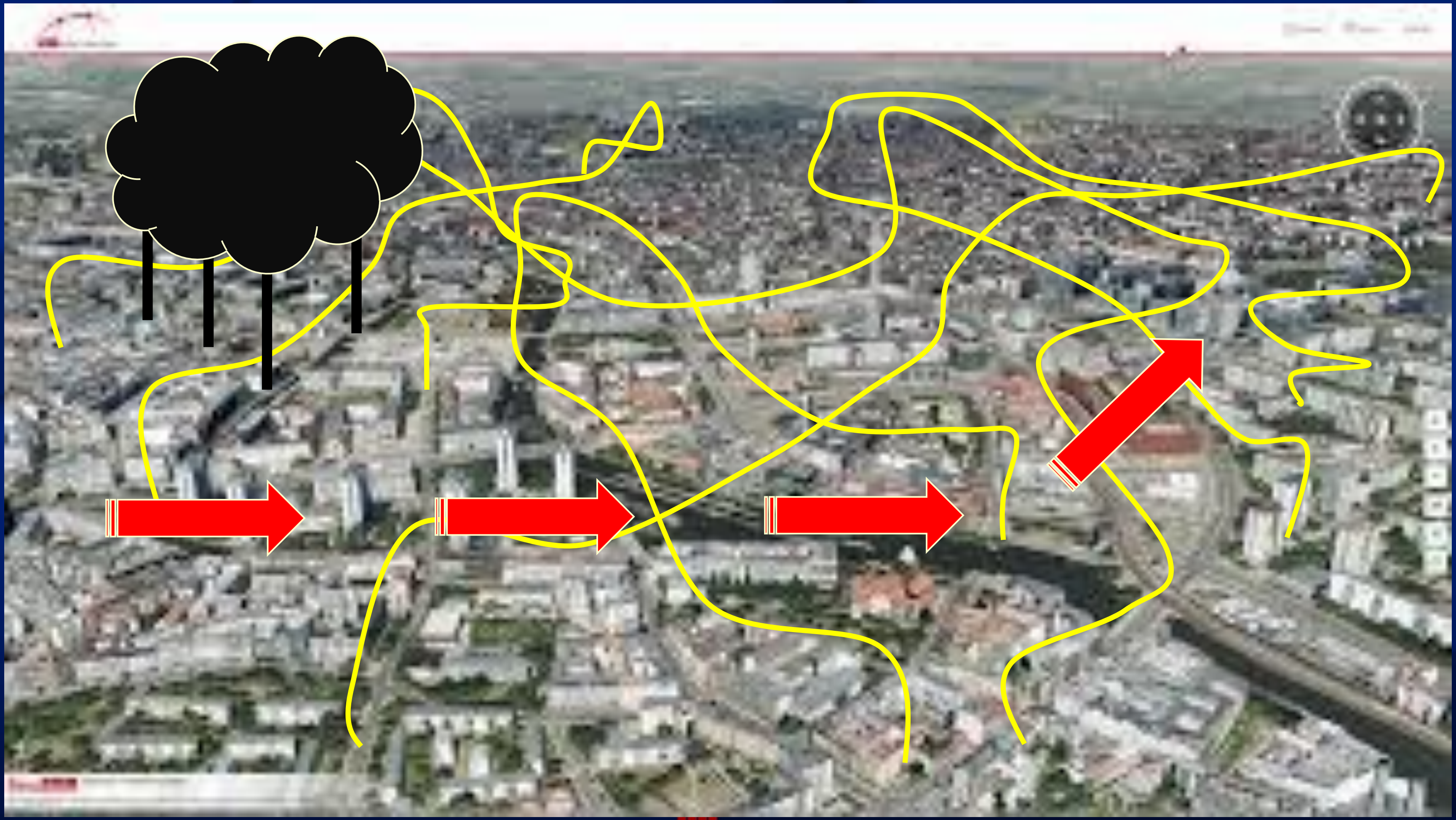
UAM: Need to Plan Flights



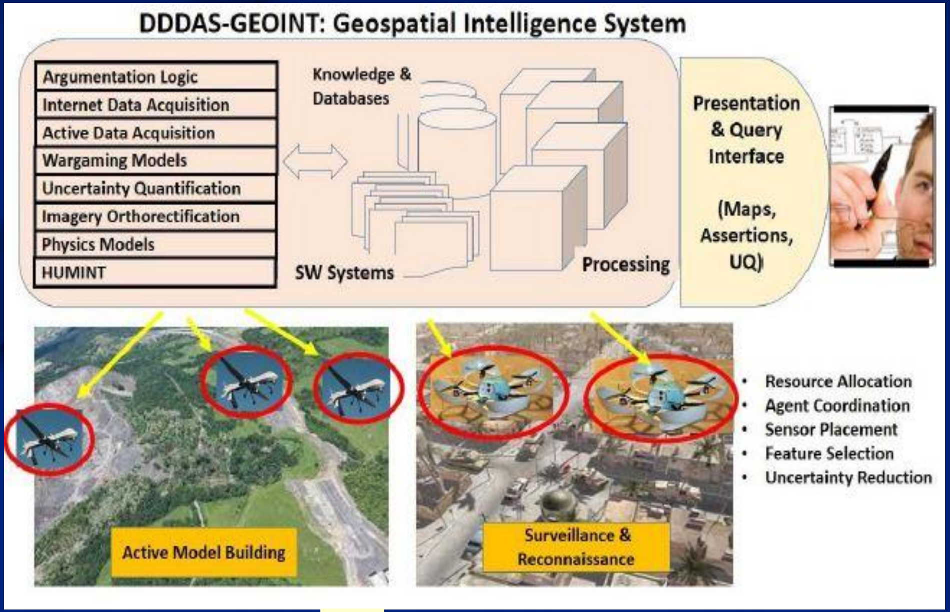
Which Path To Take? → Use lanes

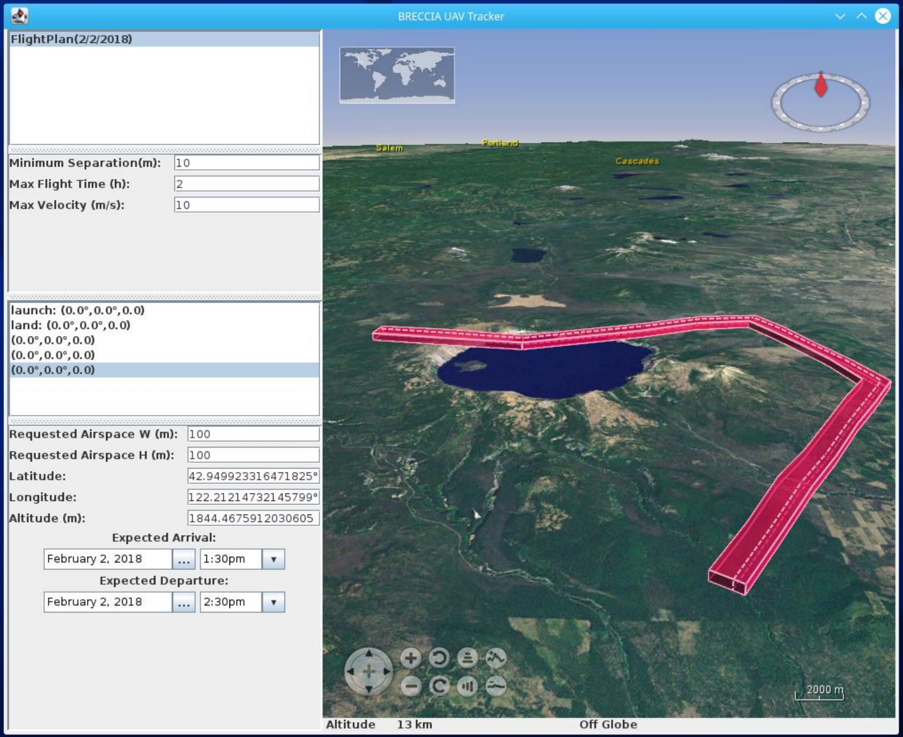
What about Wind? What about Rain?

Reinforcement Learning For Optimal Policies

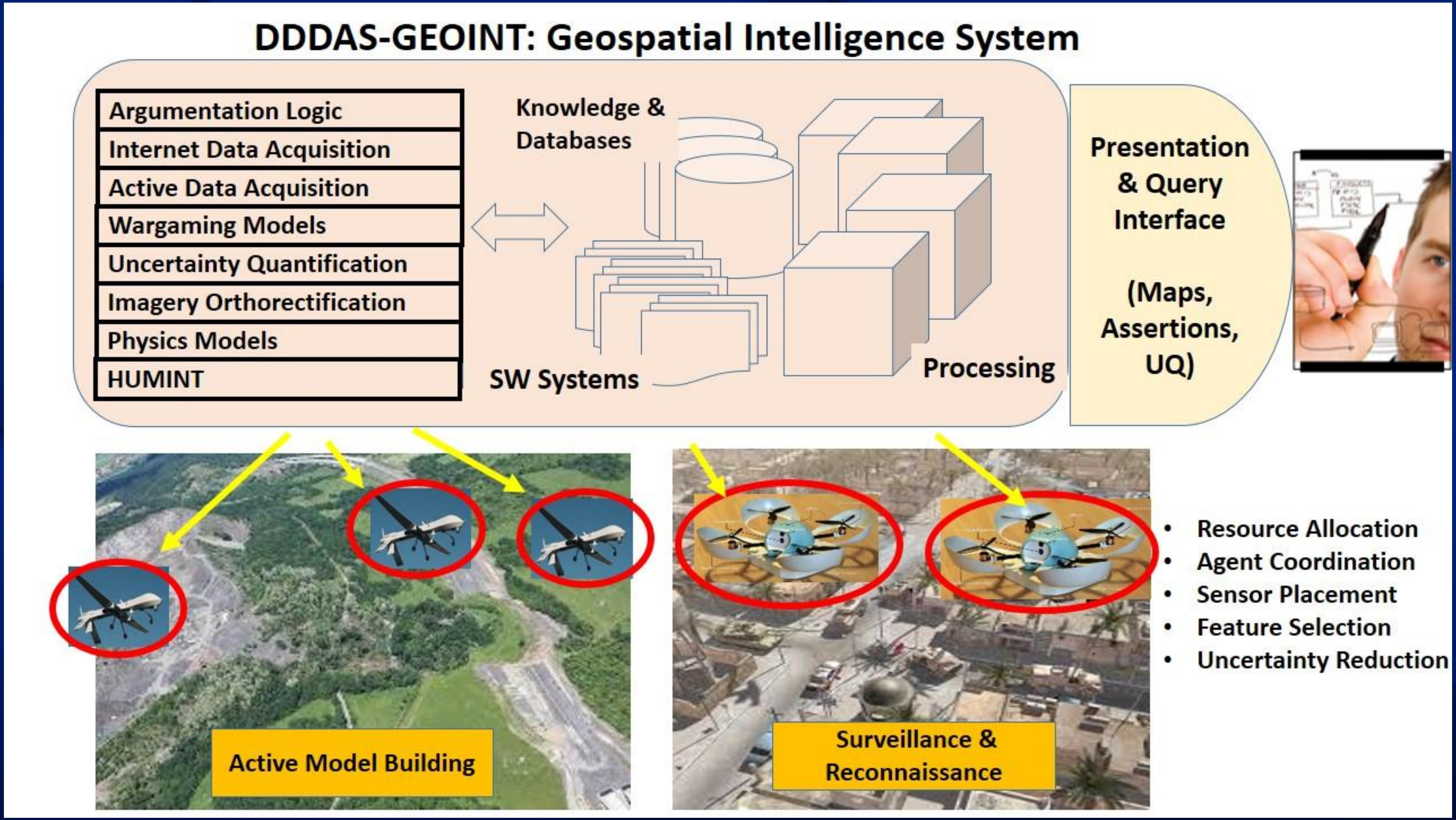


Dynamic UAV Flight Path Planning in Urban Environments

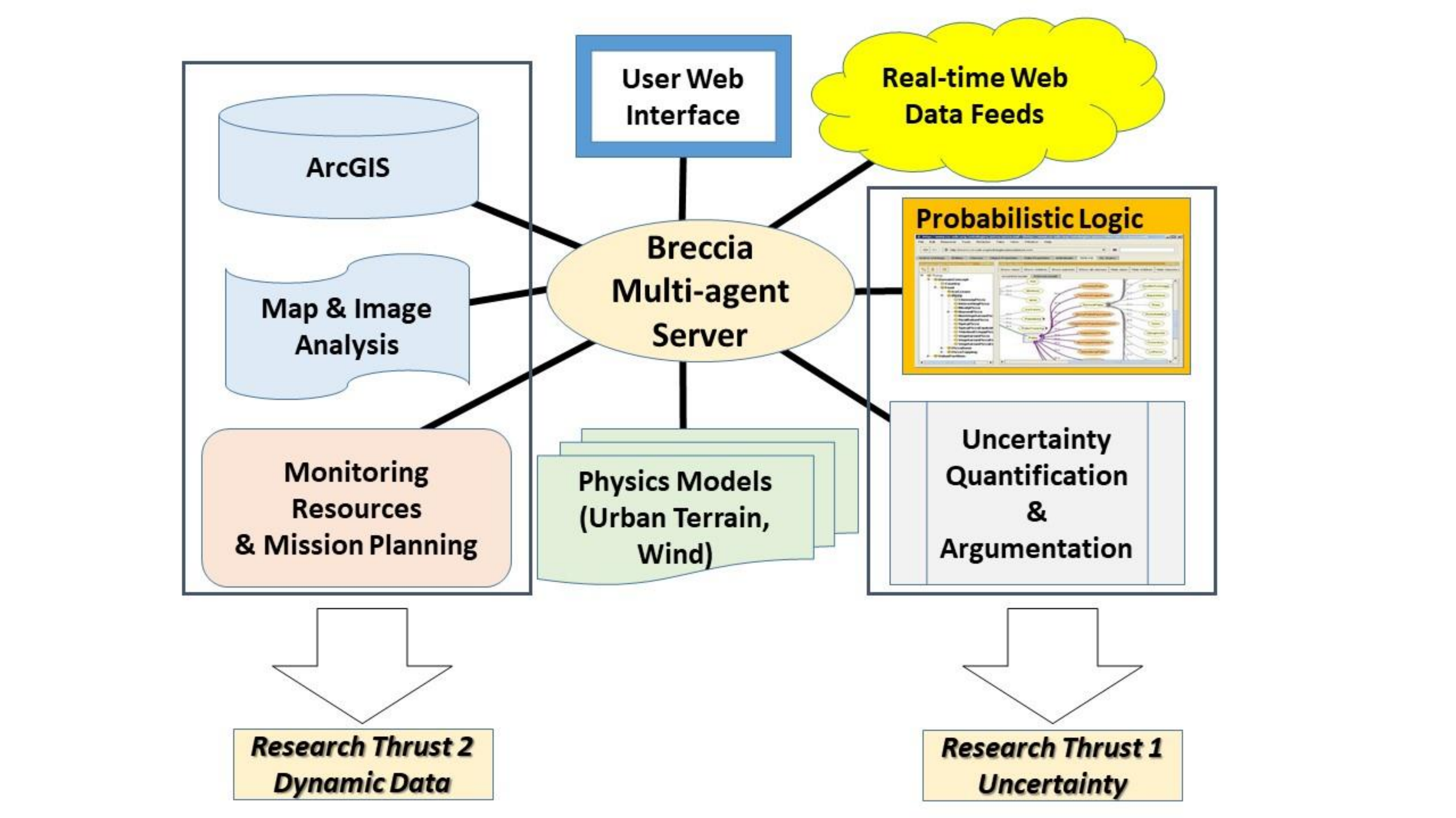




BRECCIA


DDDAS-GEOINT: Geospatial Intelligence System
Argumentation Logic
Internet Data Acquisition
Active Data Acquisition
Wargaming Models
Uncertainty Quantification
Imagery Orthorectification
Physics Models
HUMINT
Knowledge & Databases
SW Systems
Processing
Presentation & Query Interface
(Maps, Assertions, UQ)
Active Model Building
Surveillance & Reconnaissance
• Resource Allocation
• Agent Coordination
• Sensor Placement
• Feature Selection
• Uncertainty Reduction

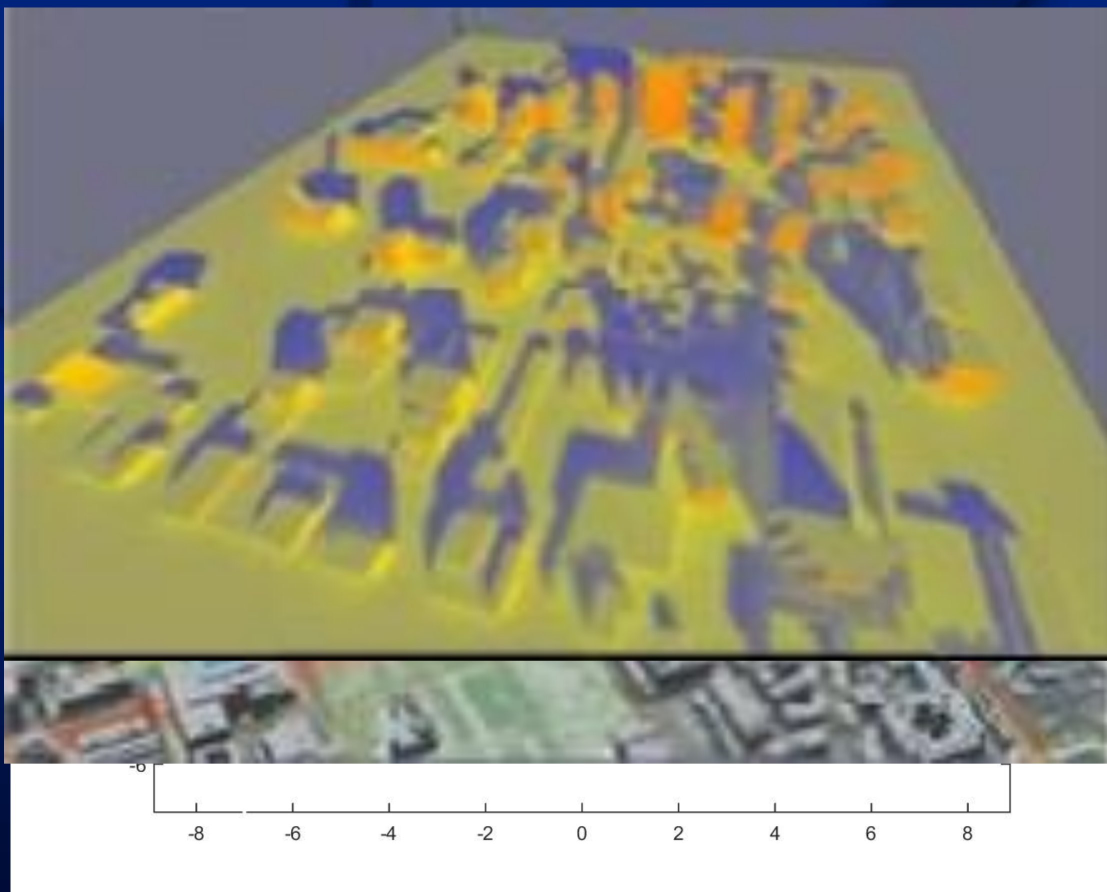
BRECCIA

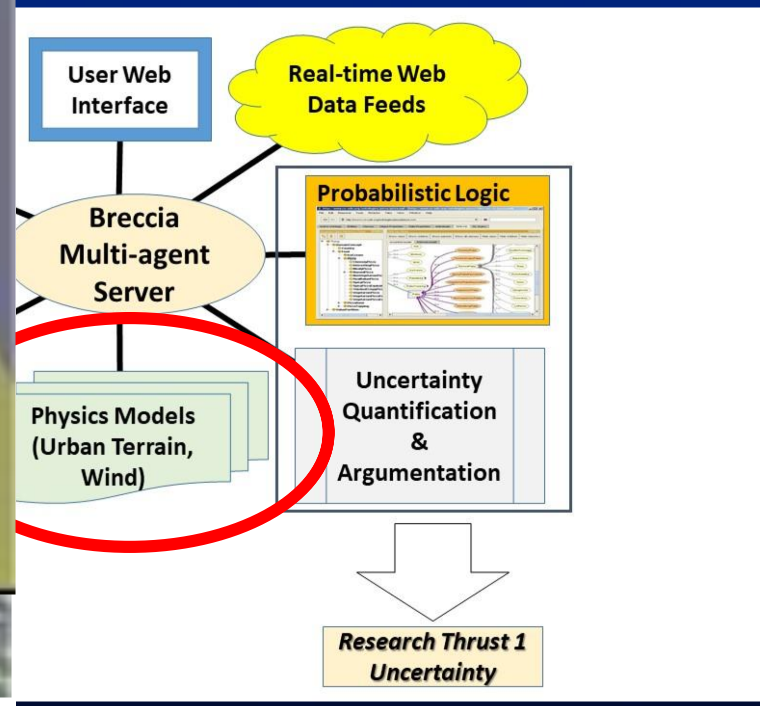


User Web Interface
Real-time Web Data Feeds
ArcGIS
Map & Image Analysis
Breccia Multi-agent Server
Probabilistic Logic
Monitoring Resources & Mission Planning
Physics Models (Urban Terrain, Wind)
Uncertainty Quantification & Argumentation
Research Thrust 2 Dynamic Data
Research Thrust 1 Uncertainty

Applications Models

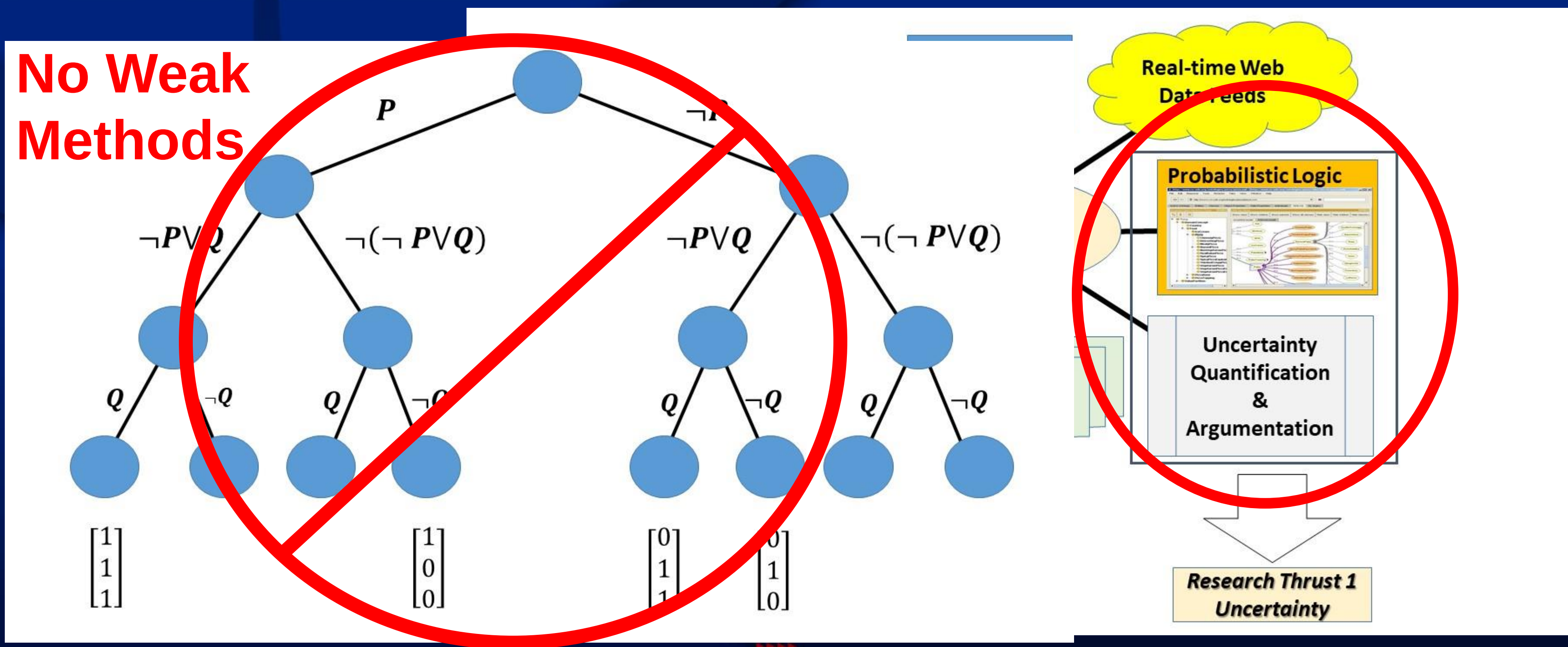
e.g., Wind/Obscurant Simulations





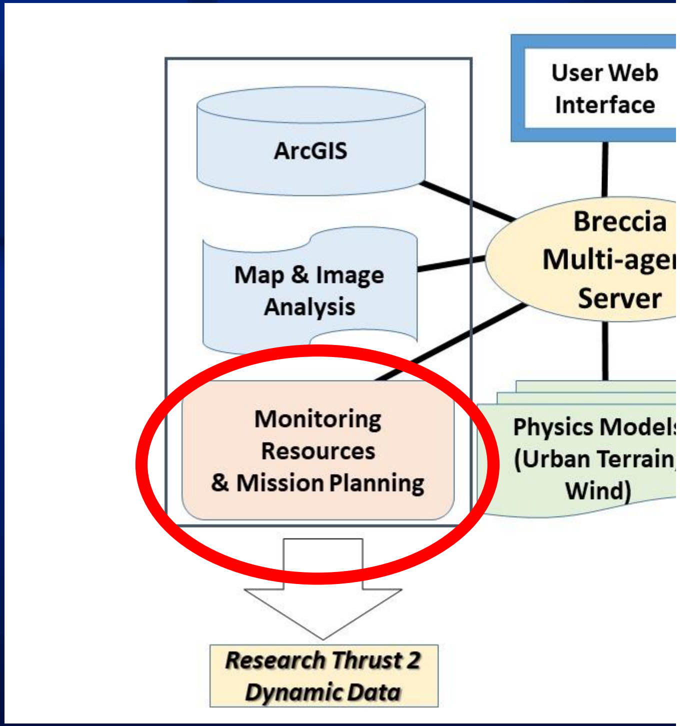
Advanced Algorithms

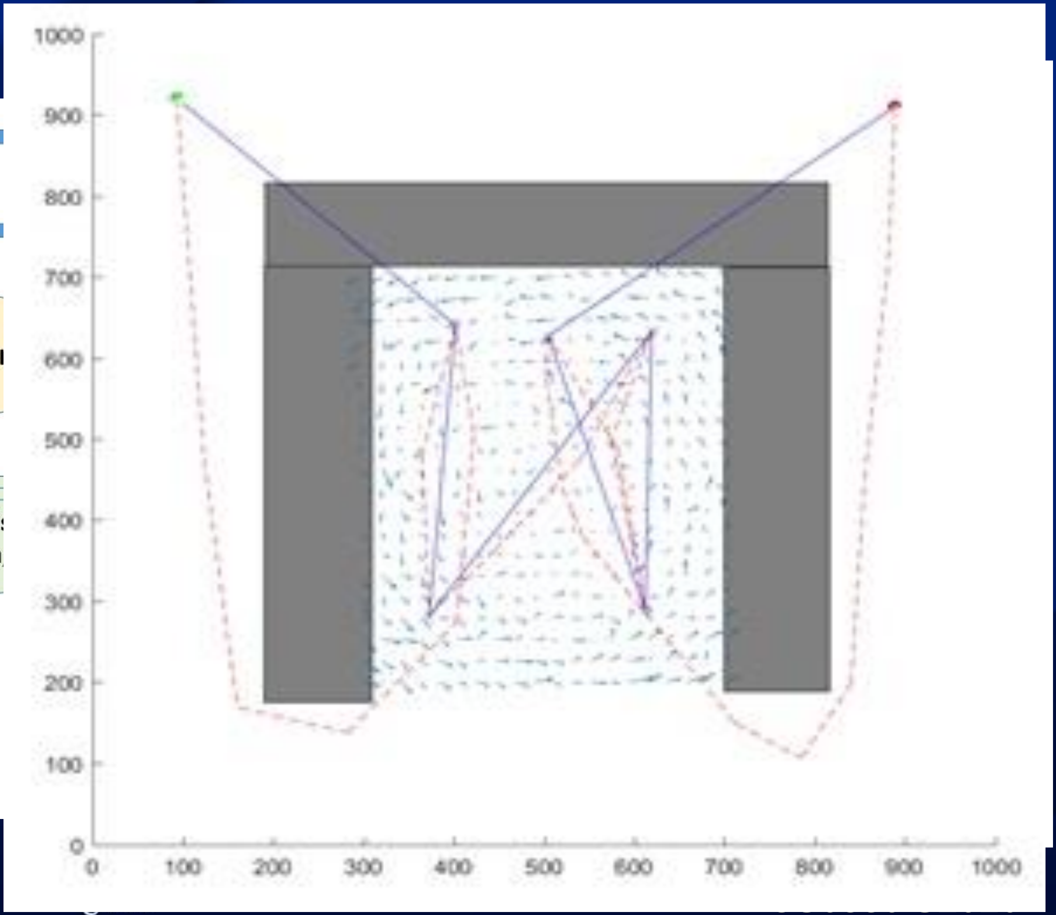
e.g., Probabilistic Logic



Application Methods

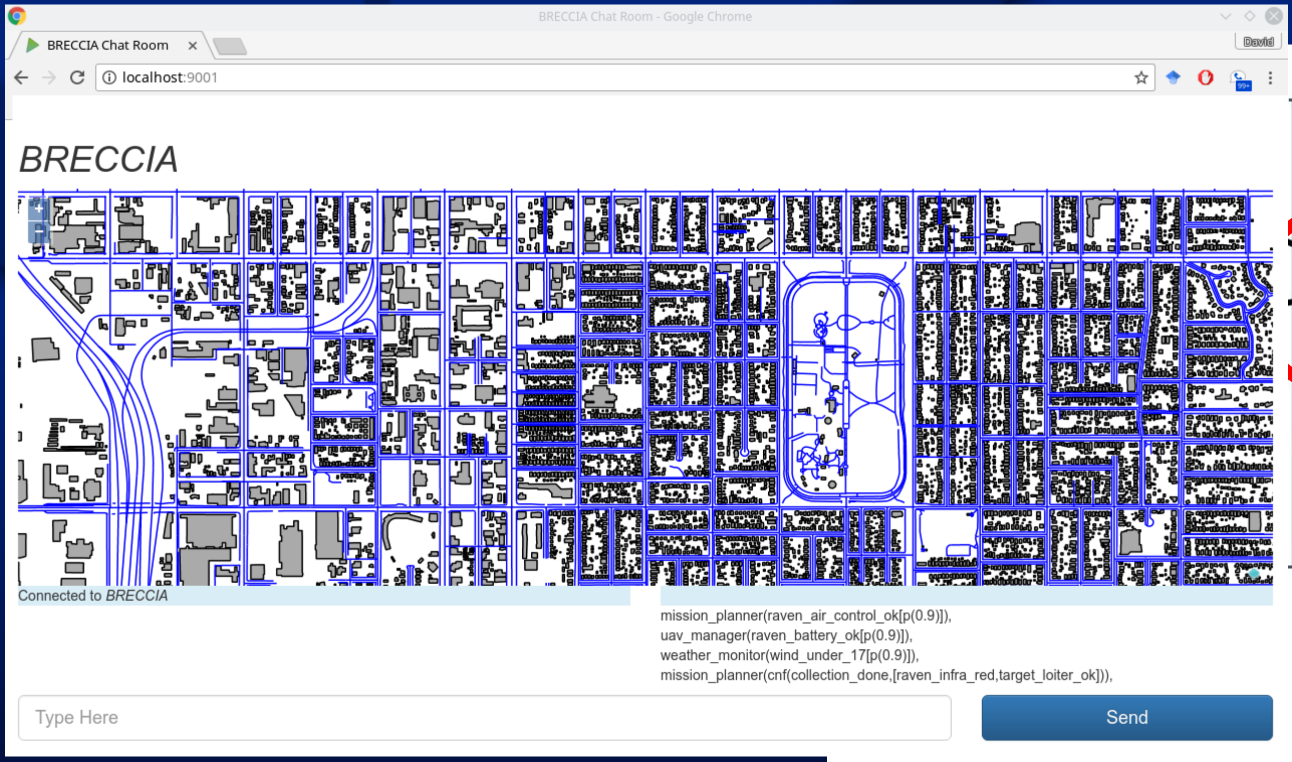
e.g., Path Planning

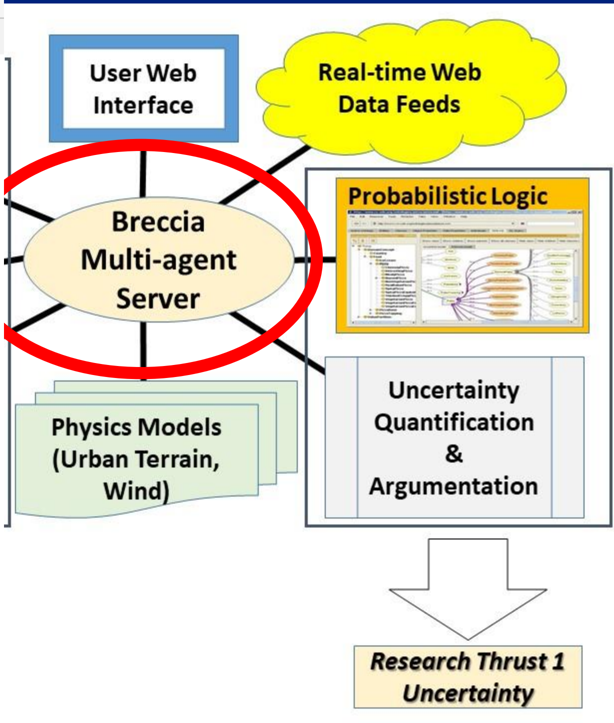




Software Infrastructure

e.g., BRECCIA Multi-agent Server





BRECCIA: Summary

- Provides **middleware** for:
  - real-time coupling of computation and knowledge
  - across heterogeneous platforms

BRECCIA: Summary

- Provides **uncertainty analysis** for
  - mission planning
  - involving combination of:
    - human statements
    - simulation results
    - sensor measurements

BRECCIA: Summary

- Agents driven by **uncertainty reduction**:
  - identification of major uncertainty sources
  - uncertainty quantification
  - propose measures for uncertainty reduction
- Next Version: URBAN (urban UAS flight planning)

URBAN: Uncertainty Reduction-Based Agent Network

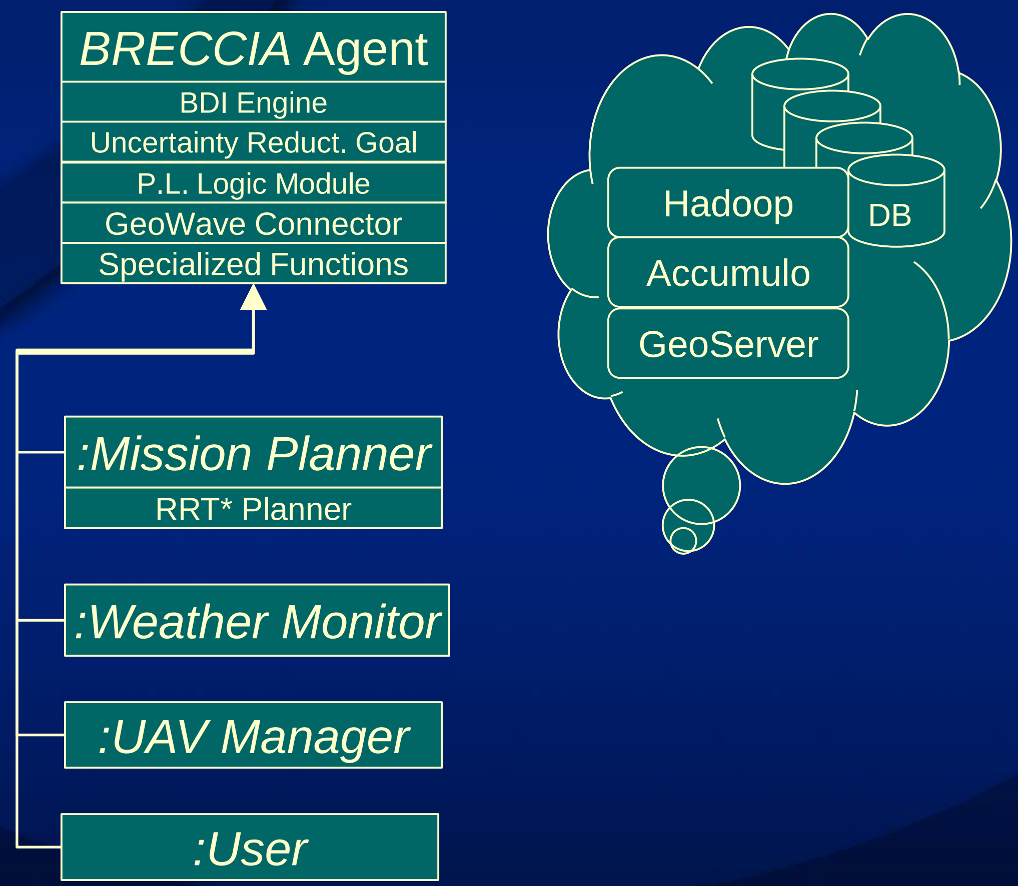
- A mult-agent system specifically designed for geospatial-temporal analysis across massive distributed datasets.
- Leverages the GeoWave project developed at the National Geospatial-Intelligence Agency (NGA) (http://locationtech.github.io/geowave/) and the open source frameworks Apache Hadoop (for distributed processing) and Accumulo (for key/value database storage).
- Conceptually, a layer atop GeoWave that provides probabilistic logical reasoning over space and time.

URBAN (cont'd)

- Dissemination of knowledge in the form of probabilistic sentences and maps published to GeoServer (http://geoserver.org/)
- Addresses tasking, processing, exploitation, and dissemination of data (TPED) with an ***agile sensor network*** and the unifying concept of ***uncertainty reduction***.

URBAN Implementation

- The BRECCIA Agent represents the core abstraction for all **agents** in the system.
- Agents are **distributed** across specialized machines such as UAVs, mobile laptops, or high performance computers.
- The inherited components of each BRECCIA agent enable an overall system that is ***dynamic and data-driven***.



Example Instantiations of the BRECCIA Agent

URBAN Implementation

BRECCIA Agent
BDI Engine
Uncertainty Reduct. Goal
P.L. Logic Module
GeoWave Connector
Specialized Functions

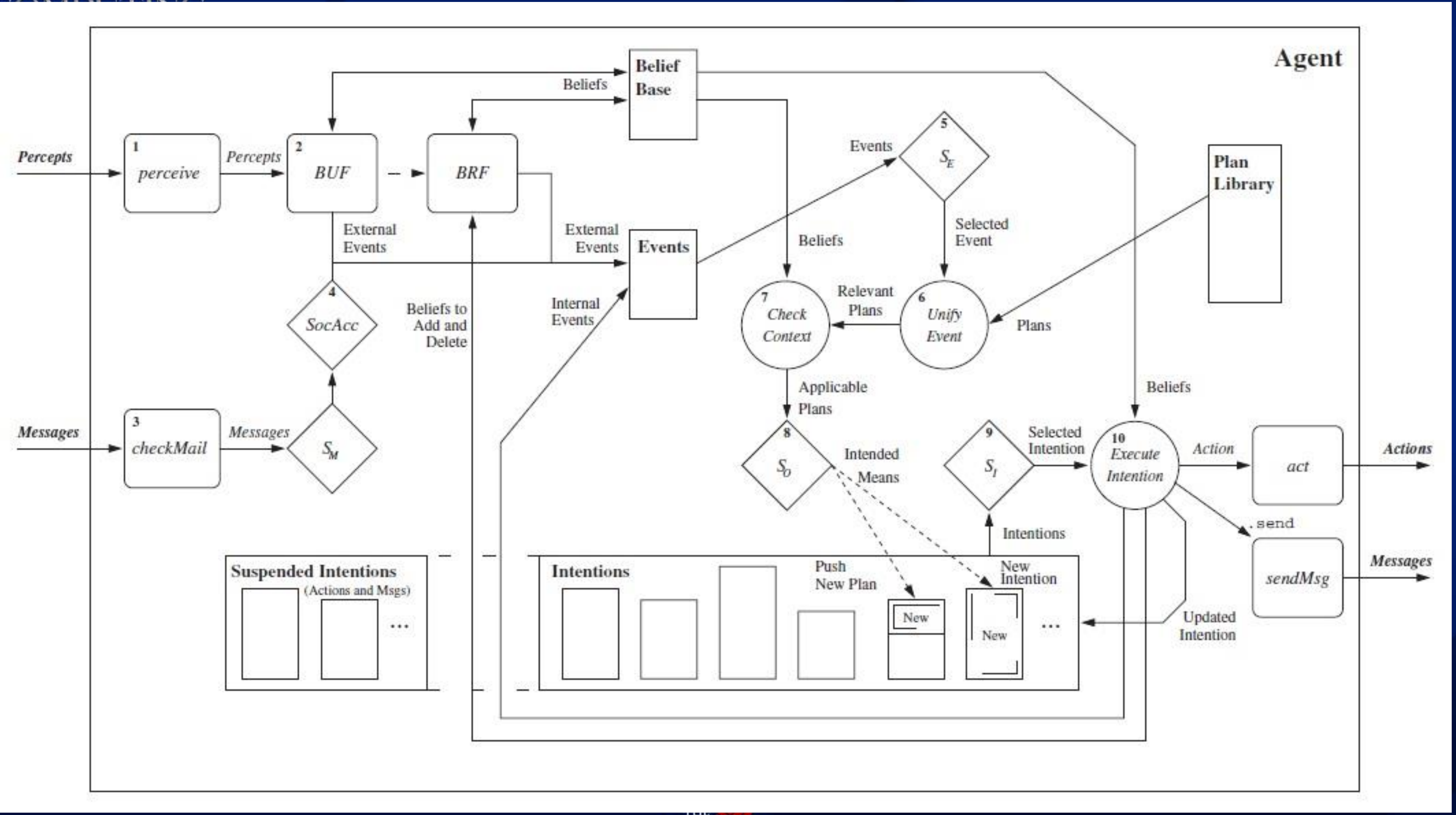
- The Belief-Desire-Intention (BDI) engine serves a dual purpose
  - As a software architecture it **facilitates the discussion and design of agents**
  - As a software cognitive model it **enables goal-driven behavior**

- Jason (http://Jason.sourceforge.net/wp/) provides the language interpreter and BDI engine to BRECCIA agents.

URBAN Implementation

BRECCIA Agent
- BDI Engine
- Uncertainty Reduct. Goal
- P.L. Logic Module
- GeoWave Connector
- Specialized Functions

The Jason Reasoning Cycle. From Programming Multi-Agent Systems in AgentSpeak Using Jason (pg. 68), by Rafael H. Bordini et al., 2007, England: John Wiley and Sons Ltd.



THE UNIVERSITY OF UTAH

URBAN Implementation

- How does Jason enable data-driven behavior?
  - Plans are executed due to events which may be achievement requests or a change in belief.
  - Example: Consider the case where a UAV is executing a path and periodically querying the geospatial database for path obstruction. To cause the agent to re-plan in the event of an obstruction, the code is as follows:

+path_obstructed(PathName) -> !replan(PathName)

React to the belief that path is obstructed… …by replanning

  - The language defined by Jason is inherently data-driven.

URBAN Implementation

BRECCIA Agent
BDI Engine
Uncertainty Reduct. Goal
P.L. Logic Module
GeoWave Connector
Specialized Functions

- GeoWave Connector
  - GeoWave enables agents to simultaneously access a distributed geospatial-temporal database.
  - Agents publish geospatial knowledge, written to the database, via GeoServer. This enables remote sharing of this type of knowledge.
  - In Jason, internal actions coded into the GeoWave connector provide direct access to the databases.
  - Example from weather agent:

```
+!share_storm_info(Location, Agent) ->
        geowaveConnector::get_wms_url(Location, WmsUrl) ;
        .send(Agent, tell, storm_info(WmsUrl).
```

  - Data-driven response from UAV agent:

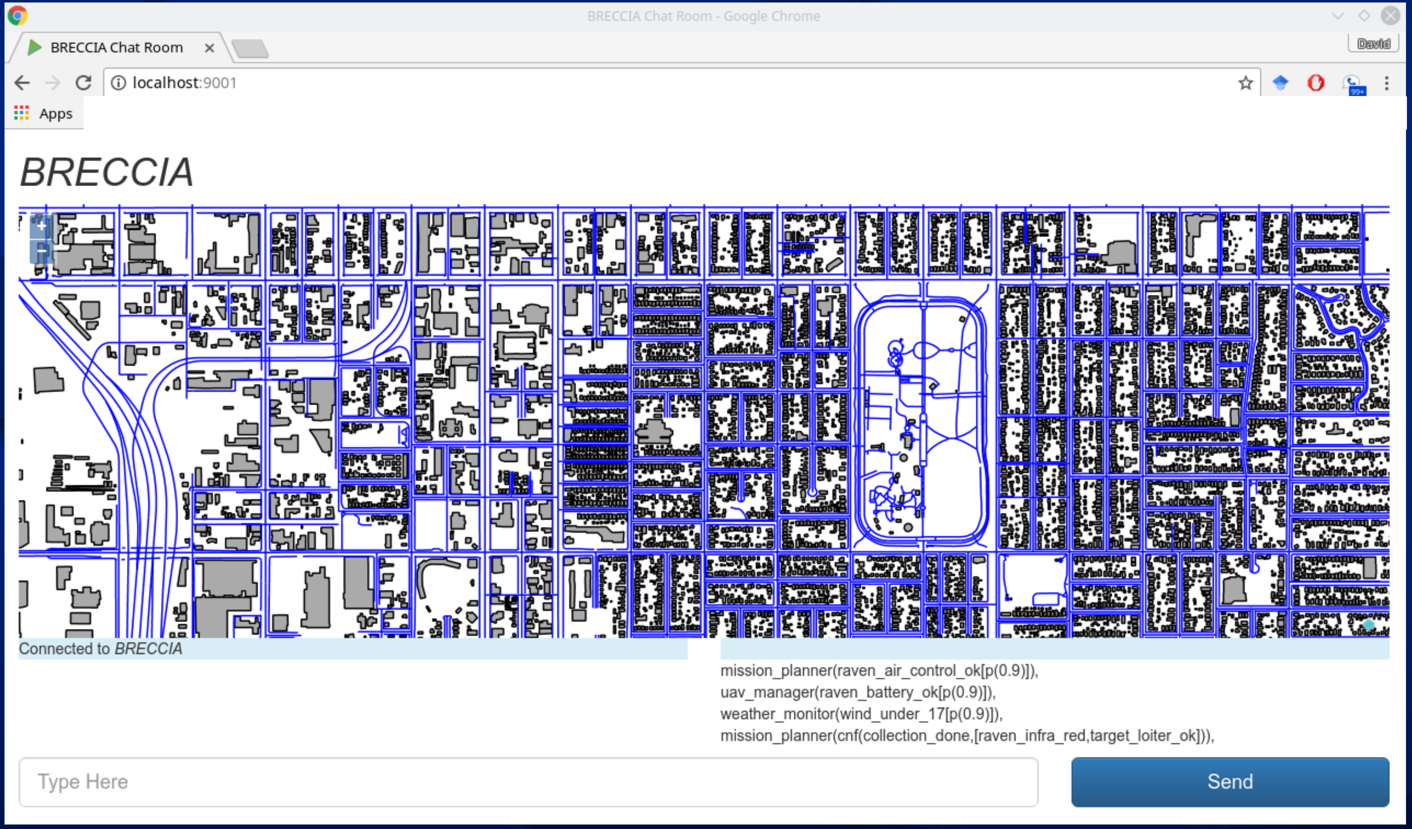
```
+storm_info(WmsUrl) -> !check_path_obstruction(WmsUrl)
```

URBAN Implementation

BRECCIA Agent
BDI Engine
Uncertainty Reduct. Goal
P.L. Logic Module
GeoWave Connector
Specialized Functions

SCHOOL OF COMPUTING

- Distributed approach enables web-based user access:



Prototype BRECCIA Client interface and Chat Window Featuring Map of Salt Lake City from Local GeoServer Instance

URBAN Implementation

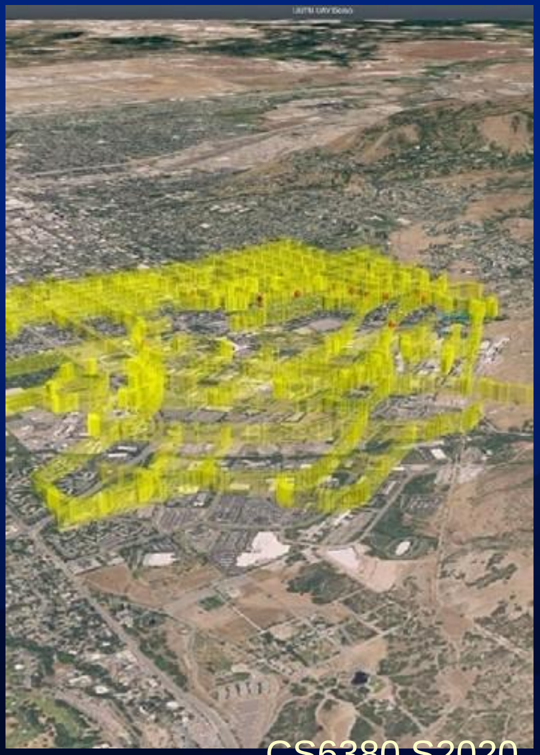
SCHOOL OF COMPUTING

- Specialized Functions
  - Current implementations of specialized functions include
    - Connecting to MATLAB instances (Agents who know how to use MATLAB)
    - RRT* path planner (Agents who know how to plan over space with vehicle constraints)
    - Wind Simulator (Agent that runs a wind vortex simulator)
  - Ongoing work of specialized functions
    - GDELT database query (Agents that can query the massive GDELT global event database (http://www.gdeltproject.org/)
    - OpenWeather API Agent (Agents that can query distributed weather information)
    - UAV simulator (Agents that can run real-time UAV simulators)
    - UAV controller (Agents that can control quadcopters in real-time)

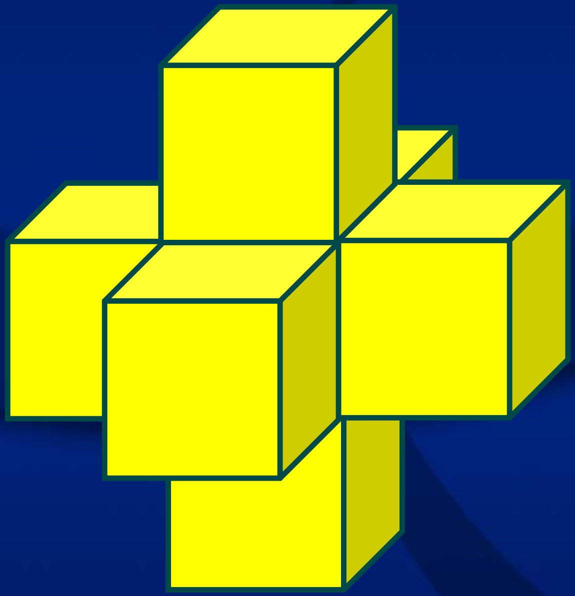
Back to Airways

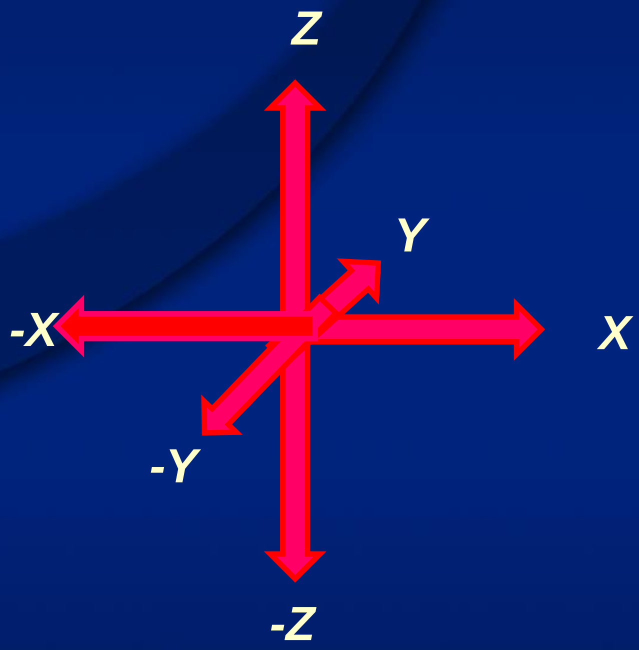
Airway Corridors

E.g., over Salt Lake City



Airspace Volumes





(a) Airspace Volumes

(b) Action Directions

Problem: Strategic Deconfliction

- Planning collision free flight paths
- Typically PSPACE-hard

➔Reduce configuration space to 1D

➔ tractable

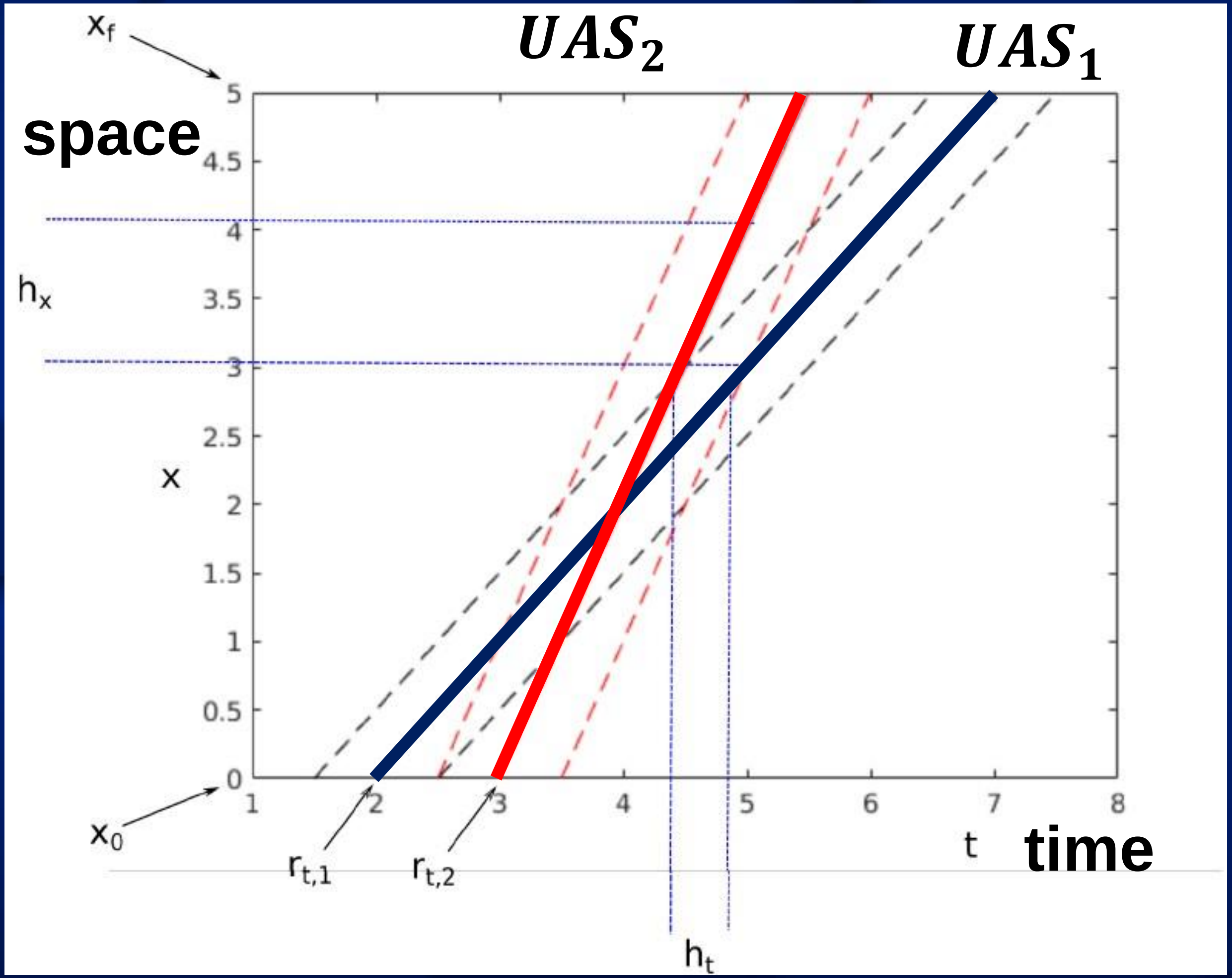
➔ allows capacity analysis

Related Problems

- (Commercial) Air Traffic Flow management
- Job scheduling problem (schedule sections)
- Multi-robot motion planning
- Traffic Assignment Problem
- Optimization Problems

Air Corridors: 1D Problem

UAS_2
Enters at t=3; exits At t=5.3



UAS_1
Enters at t=2; exits At t=7

SCHOOL OF COMPUTING

Roundabouts

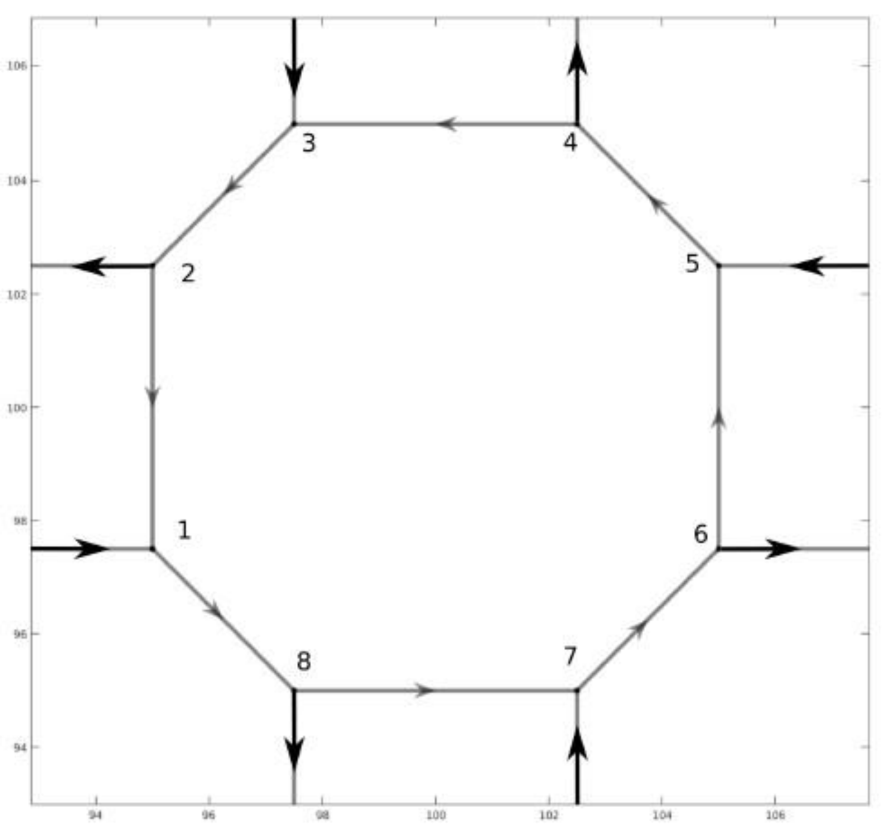


Fig. 5. Airway Roundabout

A Roundabout

A Set of Airways With Roundabouts

THE UNIVERSITY OF UTAH

SCHOOL OF COMPUTING

Issues of Interest

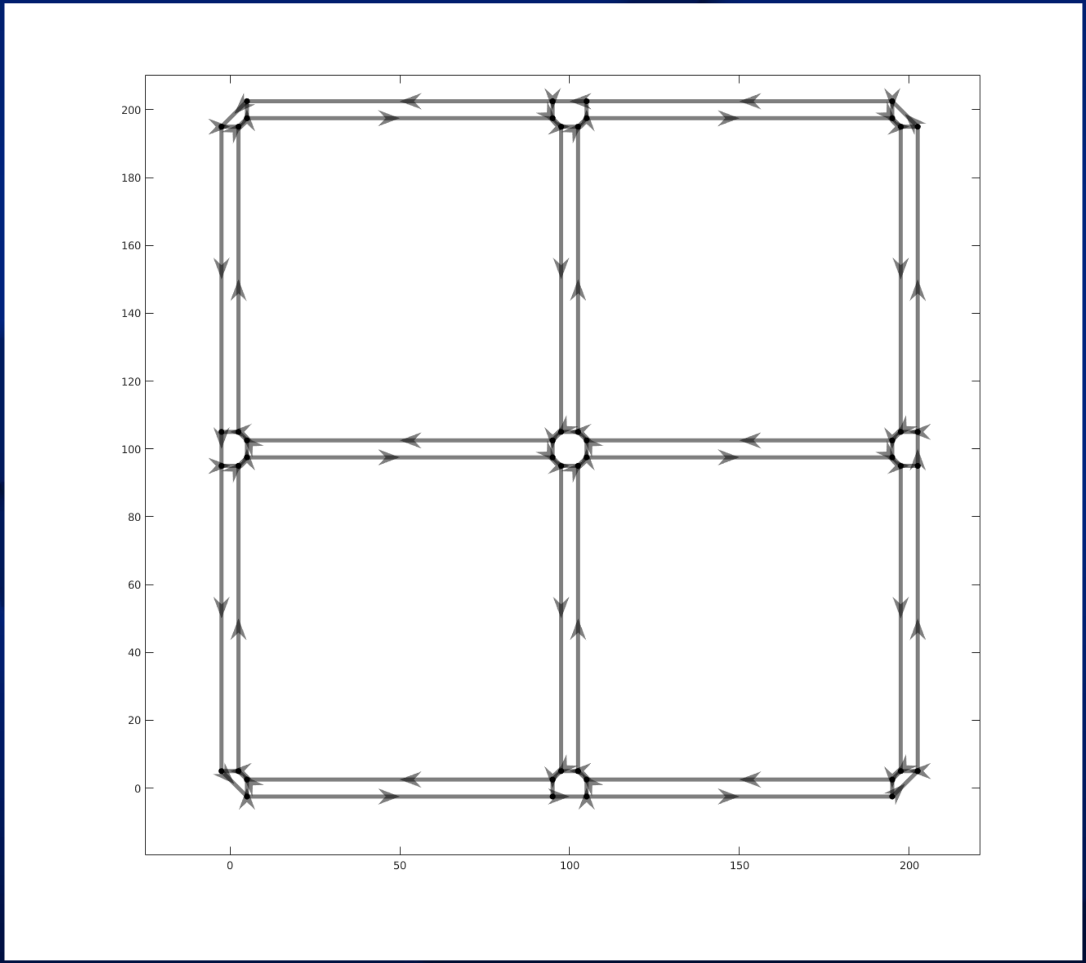
- Relationship between:
  - Airspace structure and capacity
  - Demand and reliability
- Experiments (simulation)

THE UNIVERSITY OF UTAH

Simulation Experiments

- Given network graph
- Demand: uniform distribution of vertex pairs
- Ground speed: 5 m/s
- Space Headway: 25m
- Yields 500 time slots to schedule
- Can't violate headway requirement

Air Traffic Network

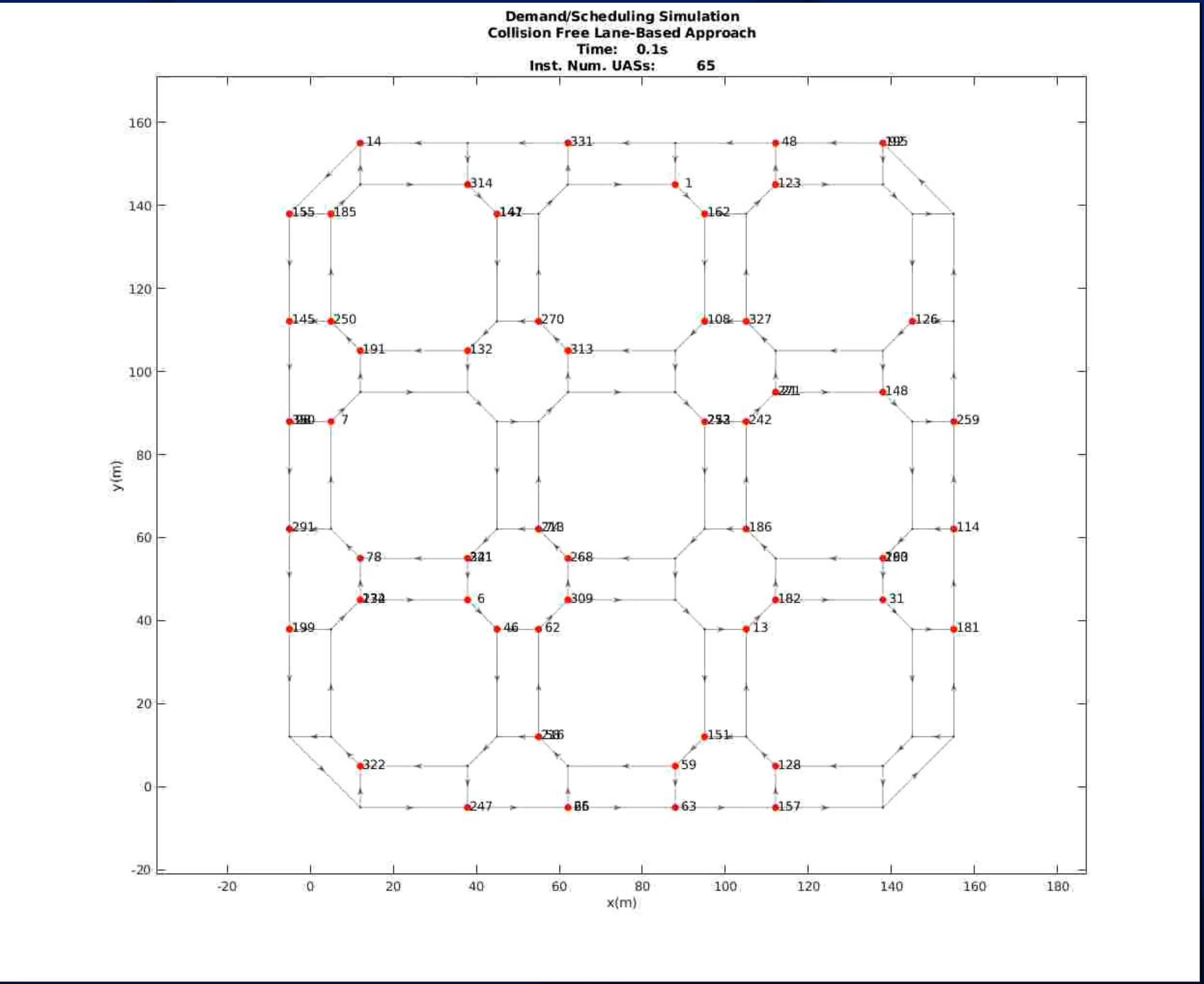


SCHOOL OF COMPUTING

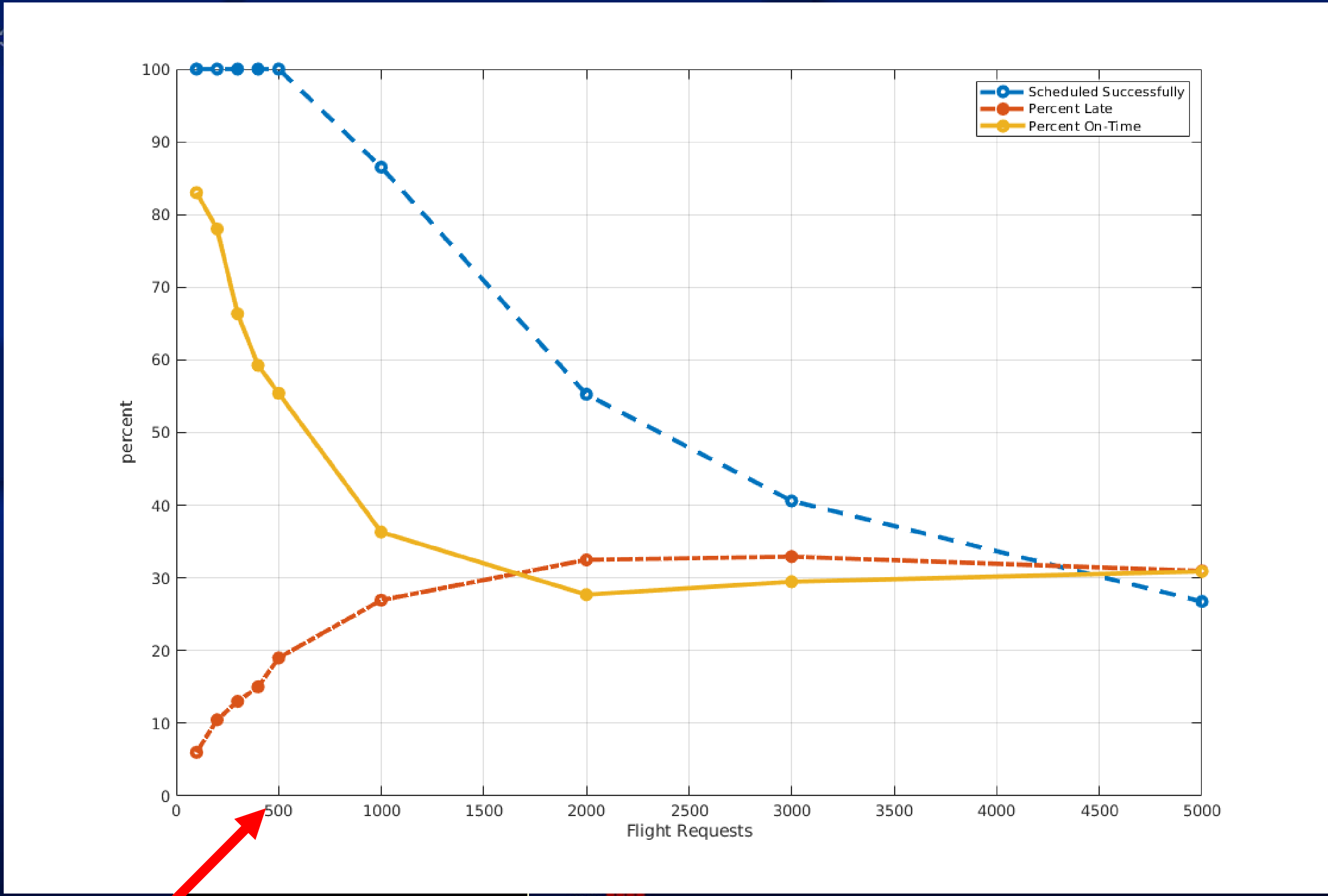
Measures

- Reliability: variability in scheduled release versus desired
- Lateness: amount of time after desired release time
- Earliness: amount of time before desired release time

Simulation Example

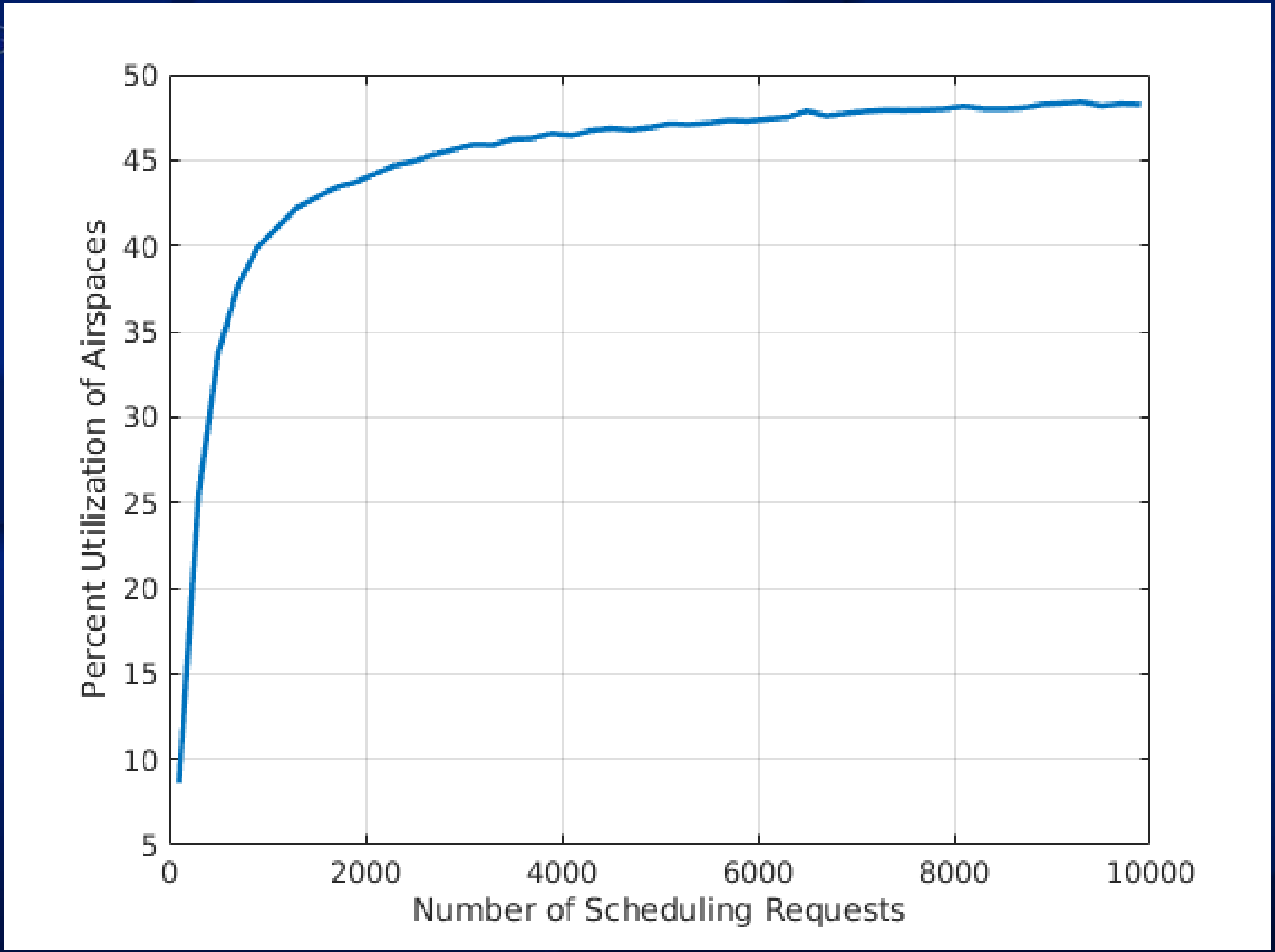


Demand vs Successful Flights

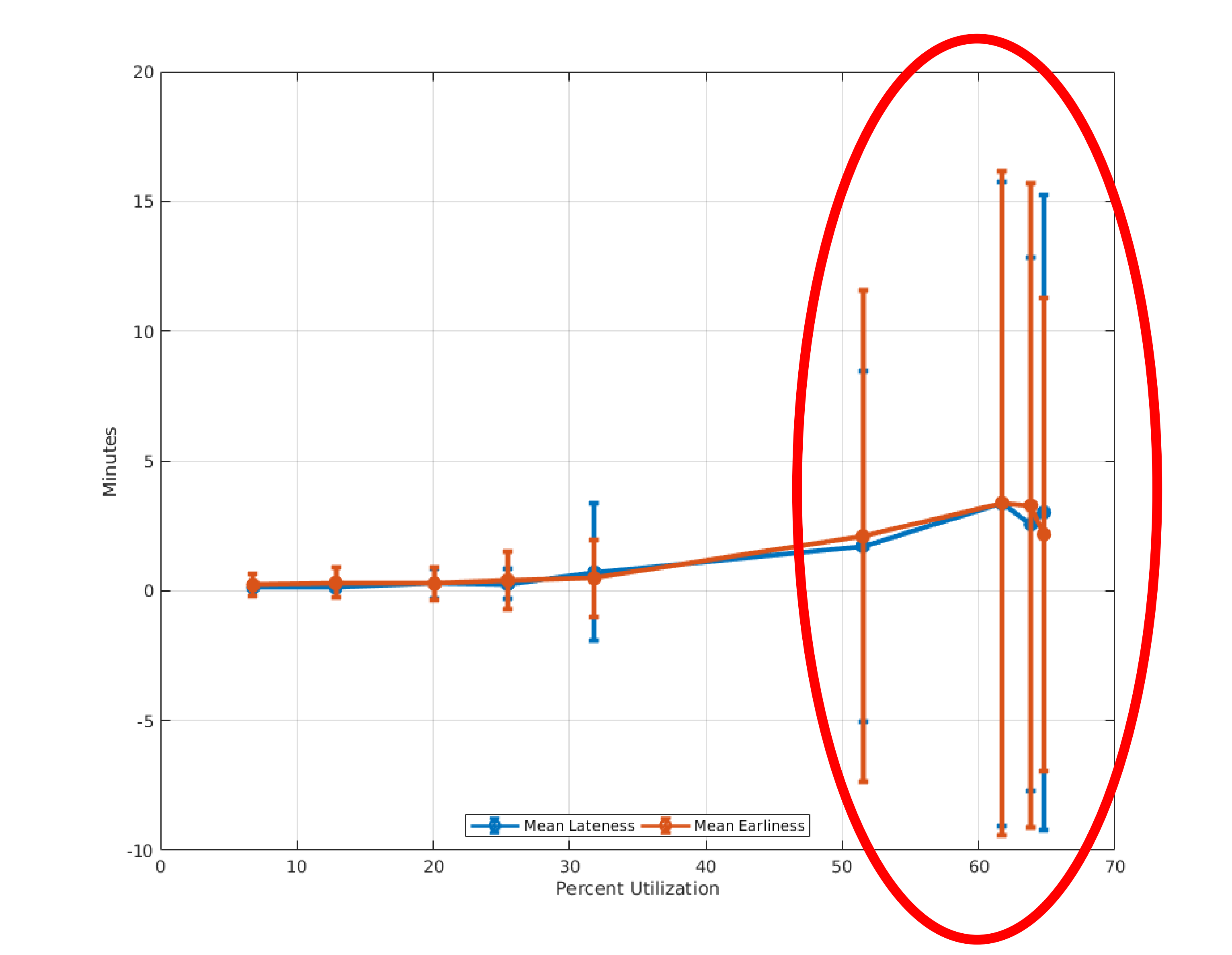


Critical point

Utilization vs Requests



Mean Lateness vs Earliness



Measures of Effectiveness

The results in previous slides used:

- To set flight parameters
- To design lanes (fast vs slow, to reduce congestion, etc.)
- To assure low variance in lateness
- etc.

Reinforcement Learning for Optimal Policies

Bellman Equations:

$$U(s) = R(s) + \gamma max_{a \in A(s)} \sum P(s' \mid s, a) U(s')$$

Optimal Policy:

$$\pi^*(s) = argmax_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s')$$

SCHOOL OF COMPUTING

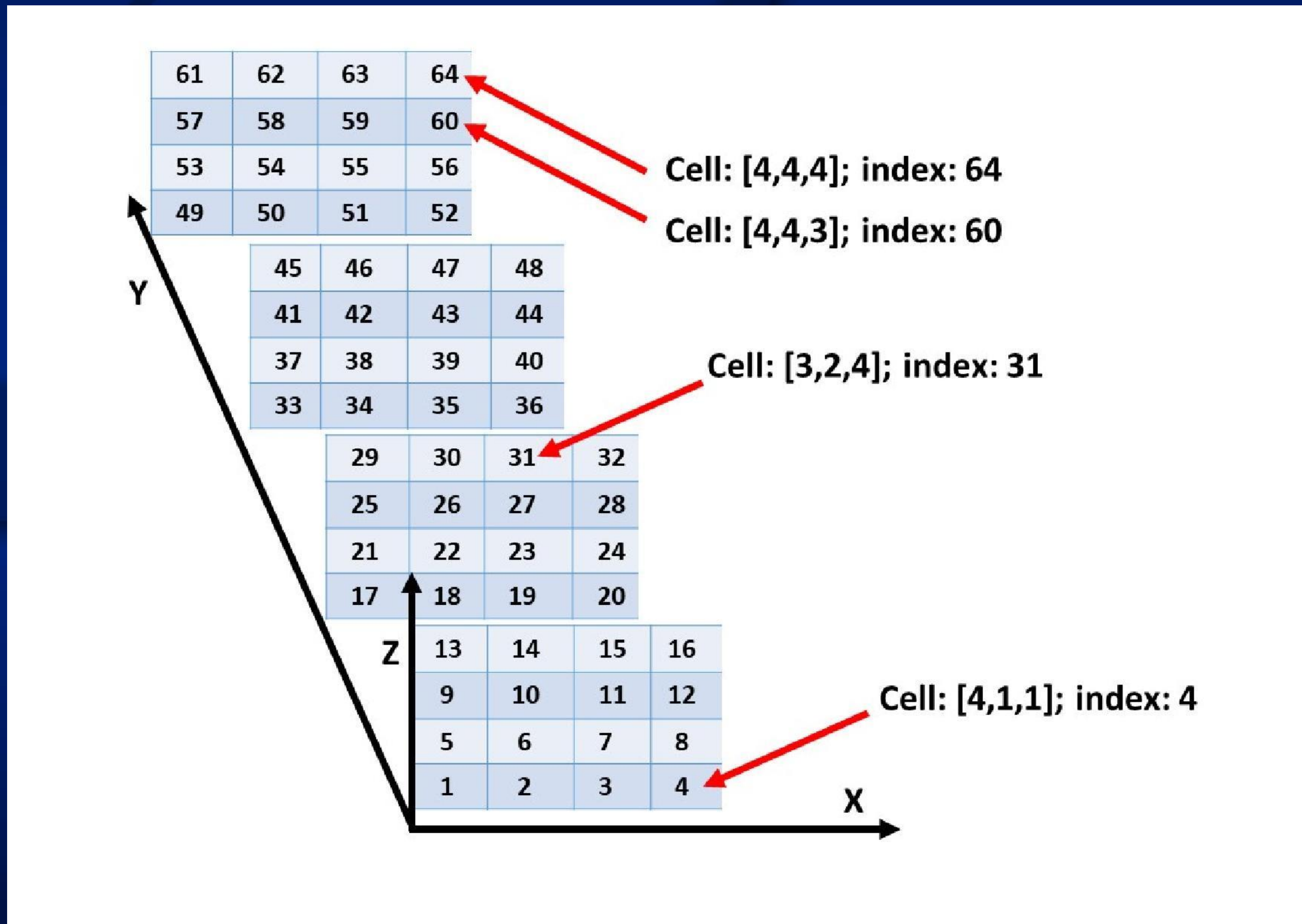
Reinforcement Learning

- States
- Rewards
- Actions
- Transition Probabilities

THE UNIVERSITY OF UTAH

Learning Optimal Action Policy

4x4 Grid



Bellman Equations

$$U(s) = R(s) + \gamma max_{a \epsilon A(s)} \sum P(s'|s,a)U(s')$$

where:
U(s) is the utility of state s
a is an action
A(s) is the set of actions in state s
R(s) is the reward for state s
γ is a horizon coefficient

State Representation

state space: $S = Z^3 \text{x} R^3 \text{x} R^+ \text{x} R$

* 3 integer grid coordinates
* 3 wind vector values (x,y,z)
* 1 precipitation value
* 1 temperature value

State Representation: Reduced

state space: $S = G^3 \text{x} W \text{x} P \text{x} T$

* G = {1,2,3,4}: grid indexes
* W = {0,1}: no wind; wind
* P = {0,1}: no rain; rain
* T = {0,1,2}: cold, normal, hot (temp)

Actions

A = {X, -X, Y, -Y, Z, -Z}

* move in one of the coordinate directions

Probabilistic State Transition

Action	X	-X	Y	-Y	Z	-Z
1	0.60	0.00	0.10	0.10	0.05	0.15
2	0.00	0.60	0.10	0.10	0.05	0.15
3	0.10	0.10	0.60	0.00	0.05	0.15
4	0.10	0.10	0.00	0.60	0.05	0.15
5	0.15	0.15	0.15	0.15	0.40	0.00
6	0.05	0.05	0.05	0.05	0.40	0.80

Table 2. Probabilities Used for Transitions for Actions given Normal Temperature, No Wind and No Precipitation.

SCHOOL OF COMPUTING

Reward Function

$$R(s) = \begin{cases} -0.04 & s \neq \text{goal,excluded state} \\ -1 & \text{excluded state} \\ +1 & \text{goal state} \end{cases}$$

THE UNIVERSITY OF UTAH

SCHOOL OF COMPUTING

Value Iteration Algorithm

function VALUE-ITERATION(mdp, ϵ) **returns** a utility function
  inputs: mdp, an MDP with states S, actions $A(s)$, transition model $P(s' \mid s, a)$,
    rewards $R(s)$, discount γ
    ϵ, the maximum error allowed in the utility of any state
  local variables: U, U', vectors of utilities for states in S, initially zero
    δ, the maximum change in the utility of any state in an iteration

  repeat
    $U \leftarrow U'$; $\delta \leftarrow 0$
    for each state s **in** S **do**
      $U'[s] \leftarrow R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a)\, U[s']$
      if $|U'[s] - U[s]| > \delta$ **then** $\delta \leftarrow |U'[s] - U[s]|$
  until $\delta < \epsilon(1-\gamma)/\gamma$
  return U

Figure 17.4 The value iteration algorithm for calculating utilities of states. The termination condition is from Equation (17.8).

From: Russell & Norvig

SCHOOL OF COMPUTING

Experiments

- Start Location: 1,1,1 (index 1)
- Goal Location: 4,4,4 (index 64)
- Blocked Cell: 4,4,3 (index 60)

→ Can't exit 4x4x4

→ Preference for horizontal motion

State Utilities and Path

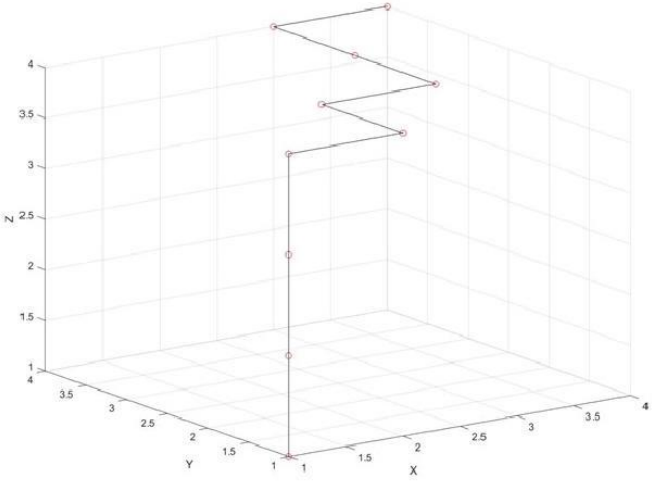
.60	.71	.83	1.0
.50	.55	.49	-1
.40	.42	.37	22
.29	.30	.27	[illegible]

.53	.627	.72	.83
.45	.51	.55	.49
.36	.39	.41	.37
.26	.28	.29	.27

.466	.54	.627	.71
.391	.44	.51	.55
.31	.35	.39	.42
.22	.25	.28	.30

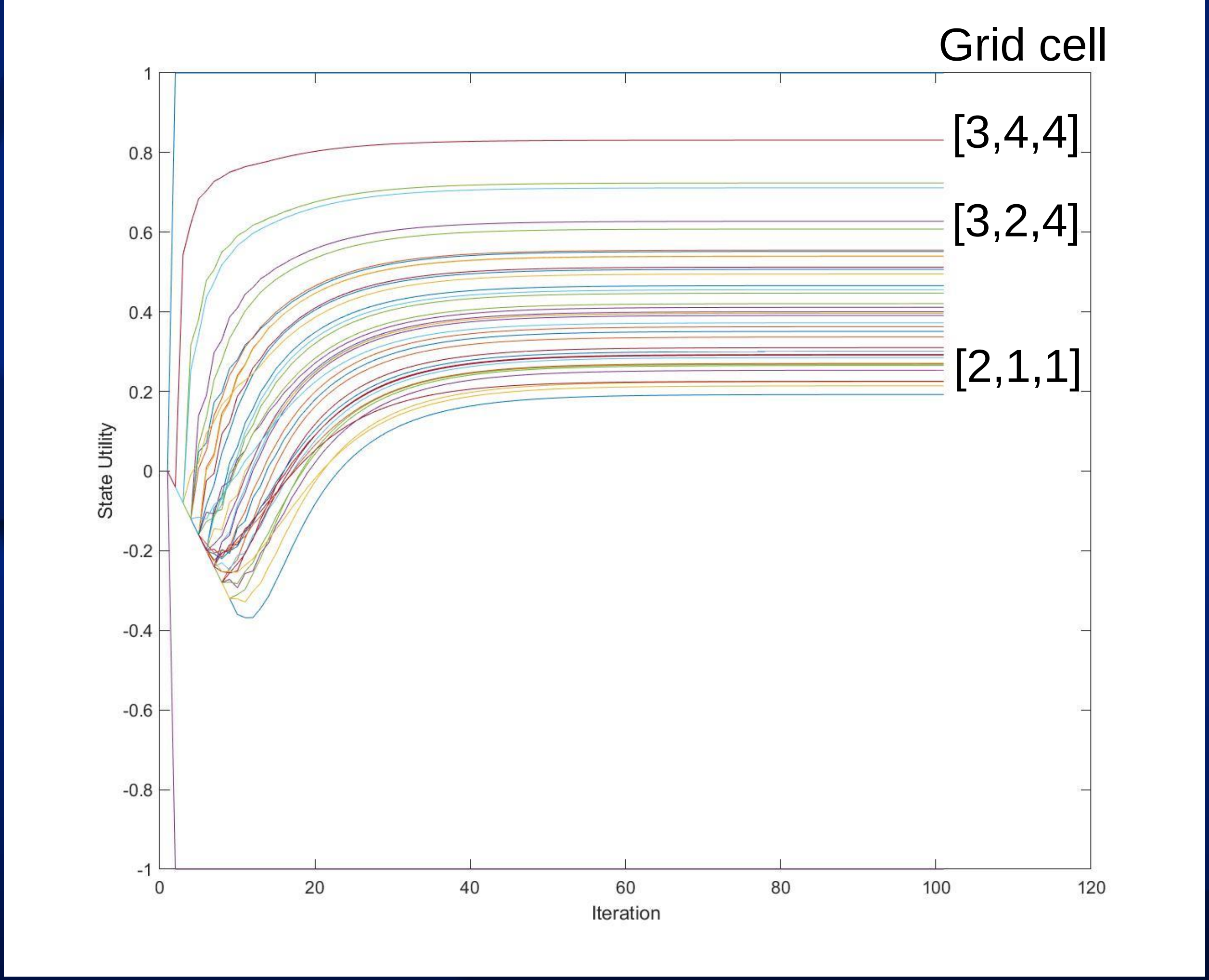
.396	.466	.53	.60
.33	.391	.45	.50
.26	.31	.36	.40
.19	.22	.26	.29

(a) State Utility Trace



(b) Flight Path Trace

Convergence for Utilities



Optimal Policies

State	Policy	State	Policy	State	Policy
1	5	23	5	45	1
2	5	24	5	46	1
3	5	25	5	47	3
4	5	26	5	48	3
5	5	27	5	49	5
6	5	28	5	50	5
7	5	29	3	51	5
8	5	30	3	52	2
9	5	31	3	53	5
10	5	32	3	54	5
11	5	33	5	55	5
12	5	34	5	56	2
13	3	35	5	57	5
14	1	36	5	58	5
15	3	37	5	59	2
16	3	38	5	60	-
17	5	39	5	61	1
18	5	40	5	62	1
19	5	41	5	63	1
20	5	42	5	64	-
21	5	43	5		
22	5	44	4		

Table 3. Optimal Policies for the states.

Z: UP

Y: BACK

X: RIGHT

Optimal Policies

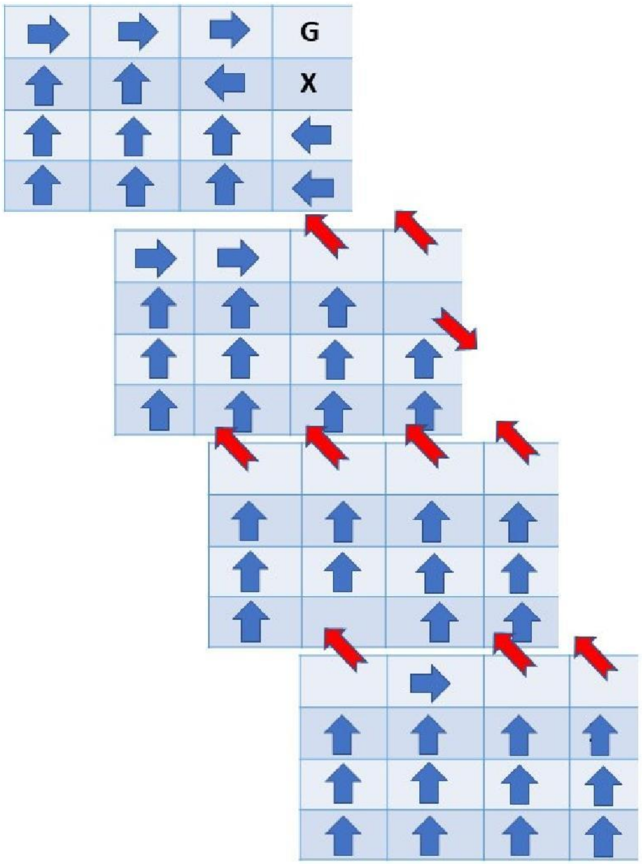
1	1	1	G
5	5	2	X
5	5	5	2
5	5	5	2

1	1	3	3
5	5	5	-3
5	5	5	5
5	5	5	5

3	3	3	3
5	5	5	5
5	5	5	5
5	5	5	5

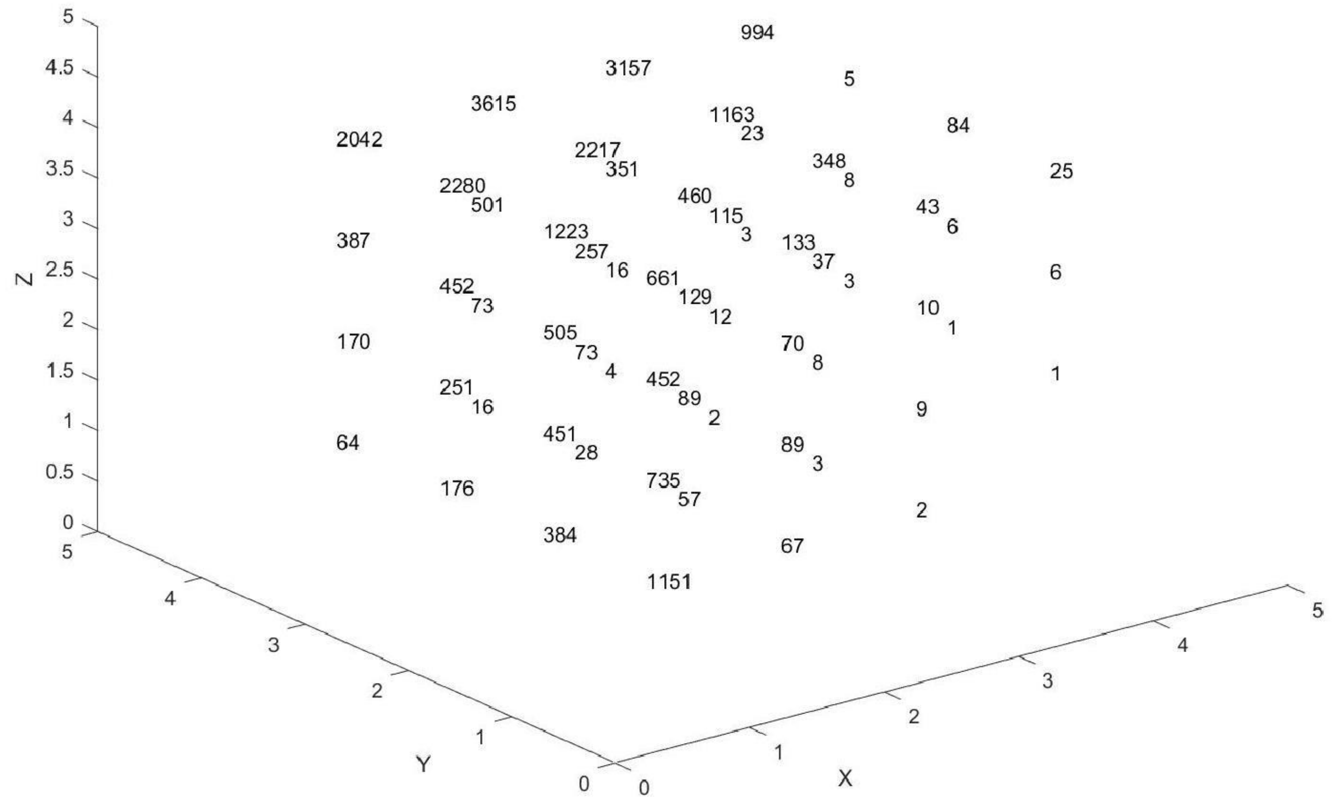
3	1	3	3
5	5	5	5
5	5	5	5
5	5	5	5

(a) Action Numbers

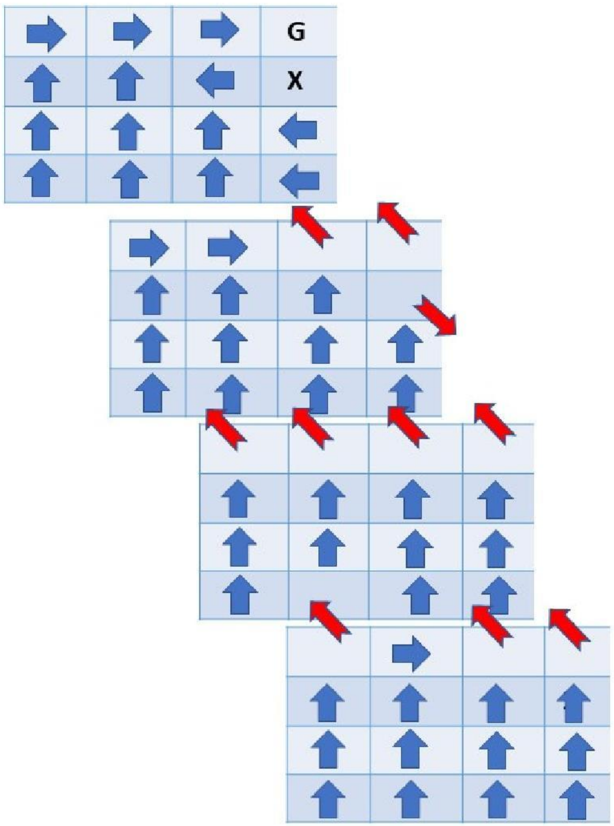


(b) Directions

Cell Travel Density

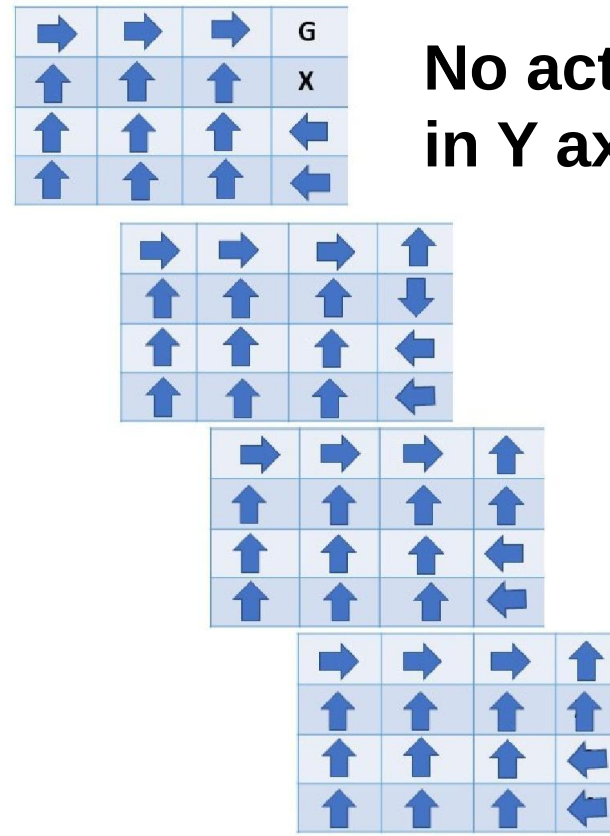


Policies with Wind in Y



(b) Directions

No Wind

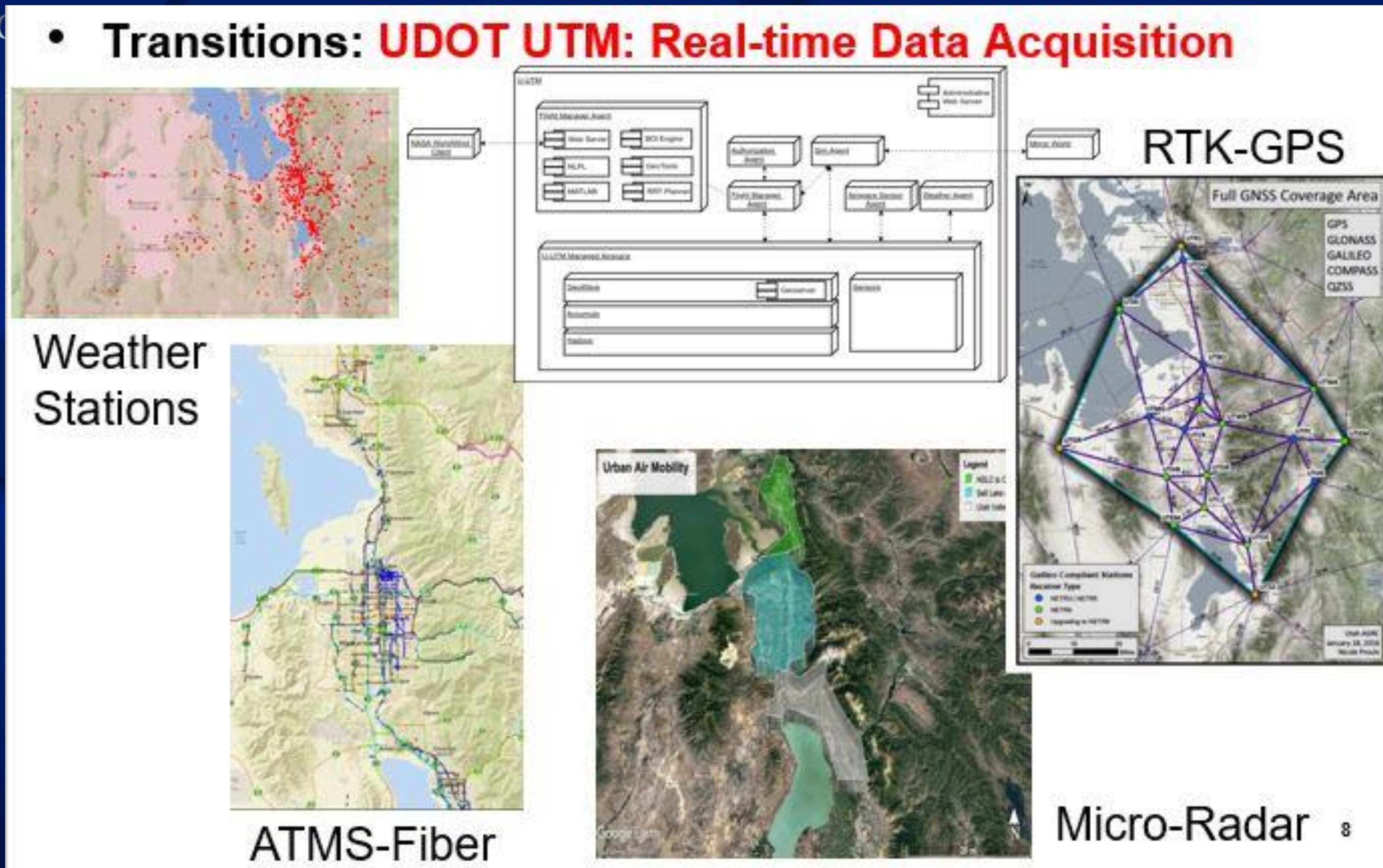


(b) Directions

Strong Wind in Y Direction

No action in Y axis!

Current Work: Get Data!

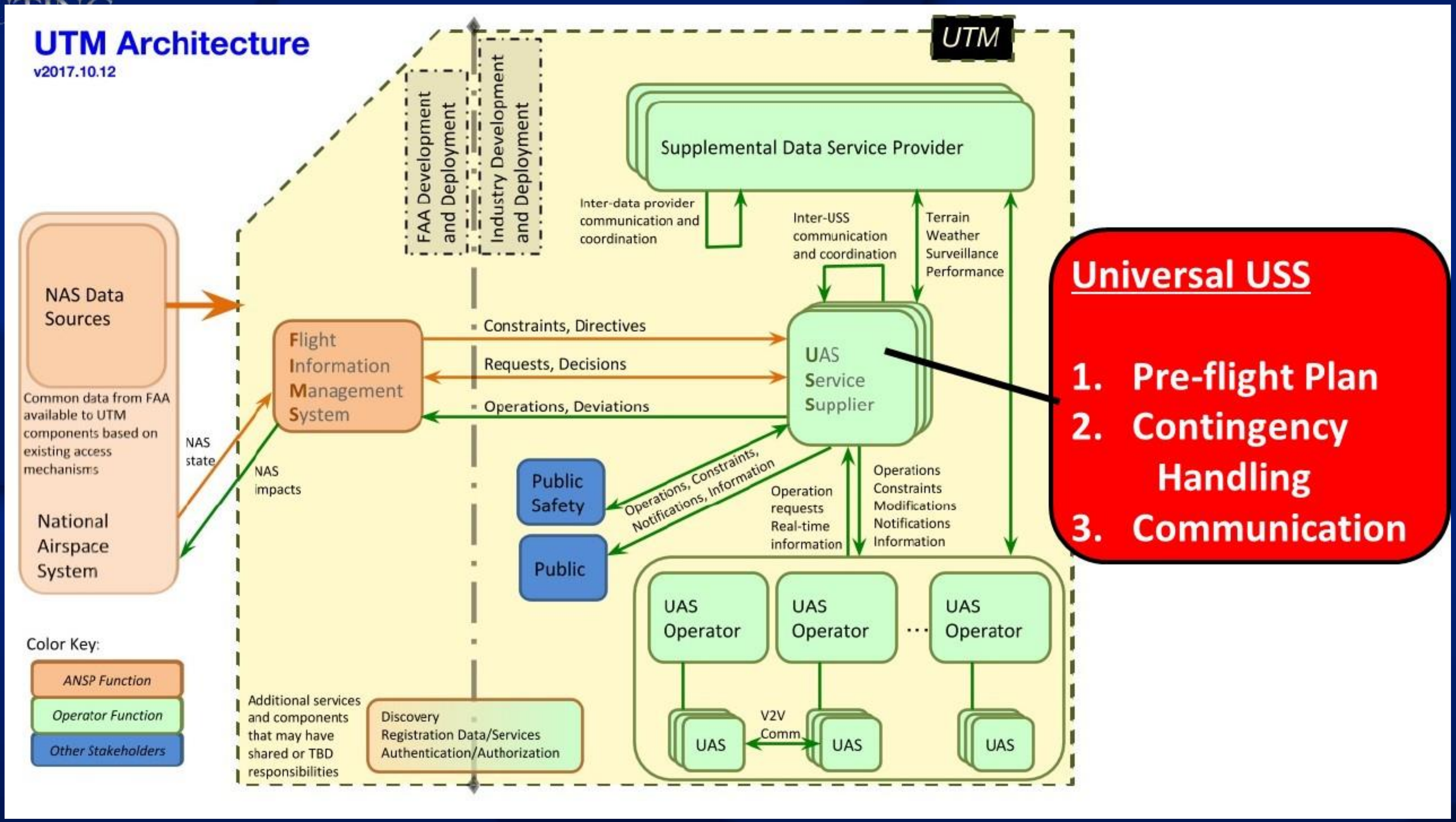


Current Work: Testing!
Deseret UAS

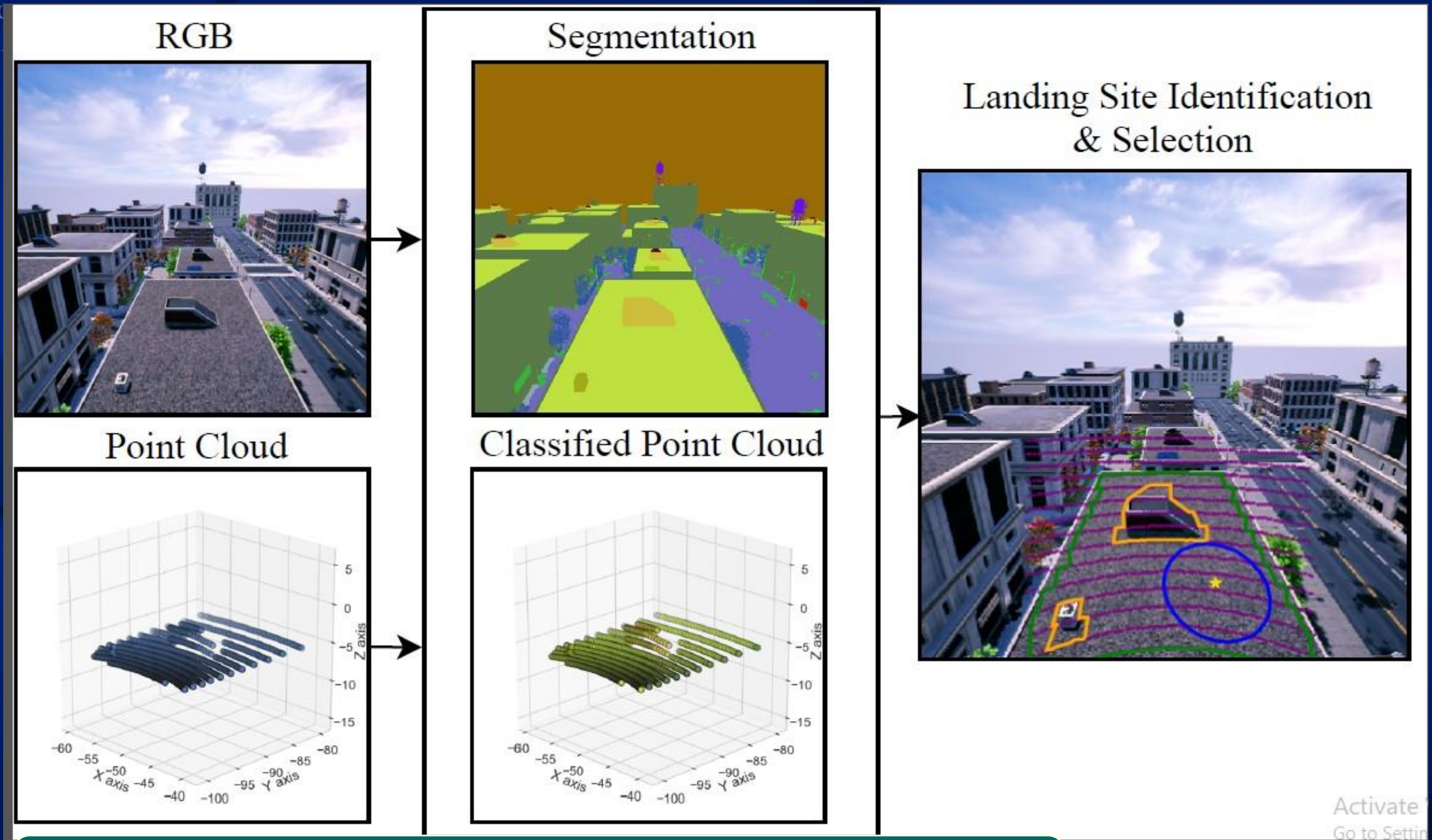
Conclusions

- Developed effective and efficient optimal policy method
- Converted core BRECCIA system to work for UAS Traffic Management
  - allows communicating, autonomous agents
  - Cloud computing

Future Work

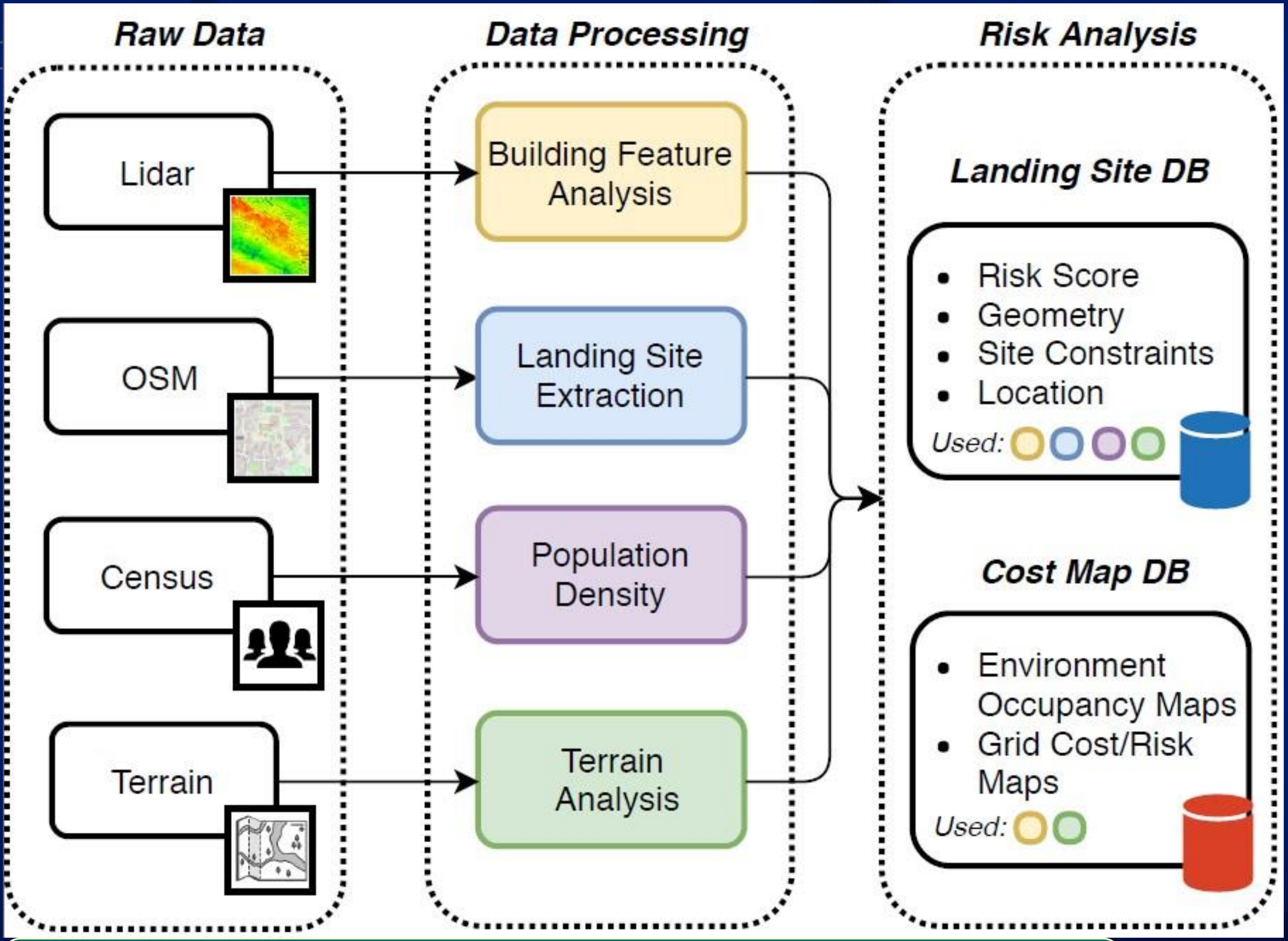

UTM Architecture
v2017.10.12
UTM
FAA Development and Deployment
Industry Development and Deployment
Supplemental Data Service Provider
Inter-data provider communication and coordination
Inter-USS communication and coordination
Terrain Weather Surveillance Performance
NAS Data Sources
Common data from FAA available to UTM components based on existing access mechanisms
NAS state
NAS impacts
National Airspace System
Flight Information Management System
Constraints, Directives
Requests, Decisions
Operations, Deviations
UAS Service Supplier
Public Safety
Public
Operations, Constraints, Notifications, Information
Operation requests Real-time information
Operations Constraints Modifications Notifications Information
UAS Operator
UAS Operator
UAS Operator
UAS
V2V Comm
UAS
UAS
Color Key:
ANSP Function
Operator Function
Other Stakeholders
Additional services and components that may have shared or TBD responsibilities
Discovery Registration Data/Services Authentication/Authorization
Universal USS
1. Pre-flight Plan
2. Contingency Handling
3. Communication

Contingency Handling



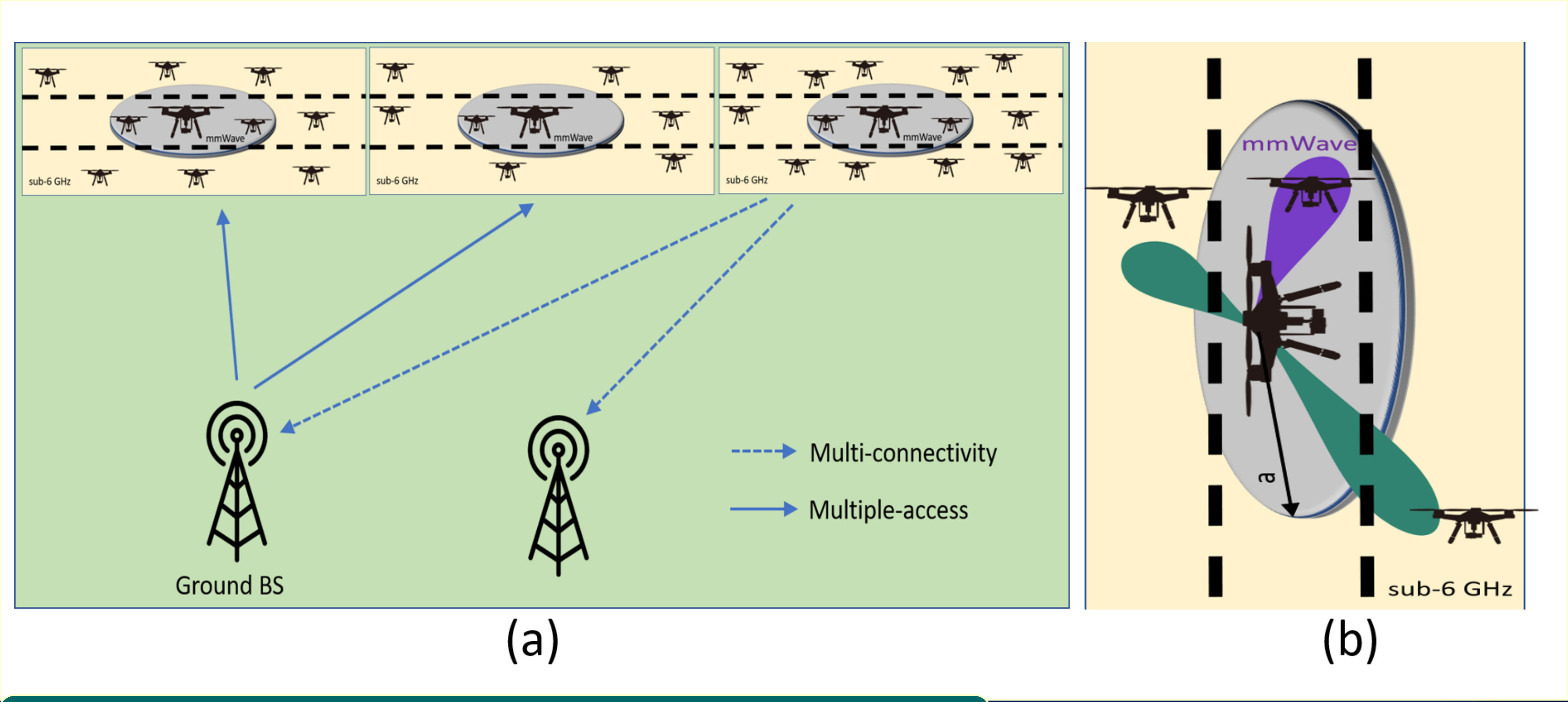
Colleague: Ella Atkins, Univ of Michigan

Contingency Handling



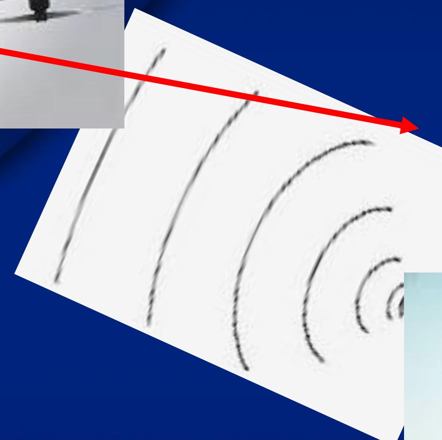
Colleague: Ella Atkins, Univ of Michigan

5G Communications



Colleague: Tadilo Bogale, NC A&T

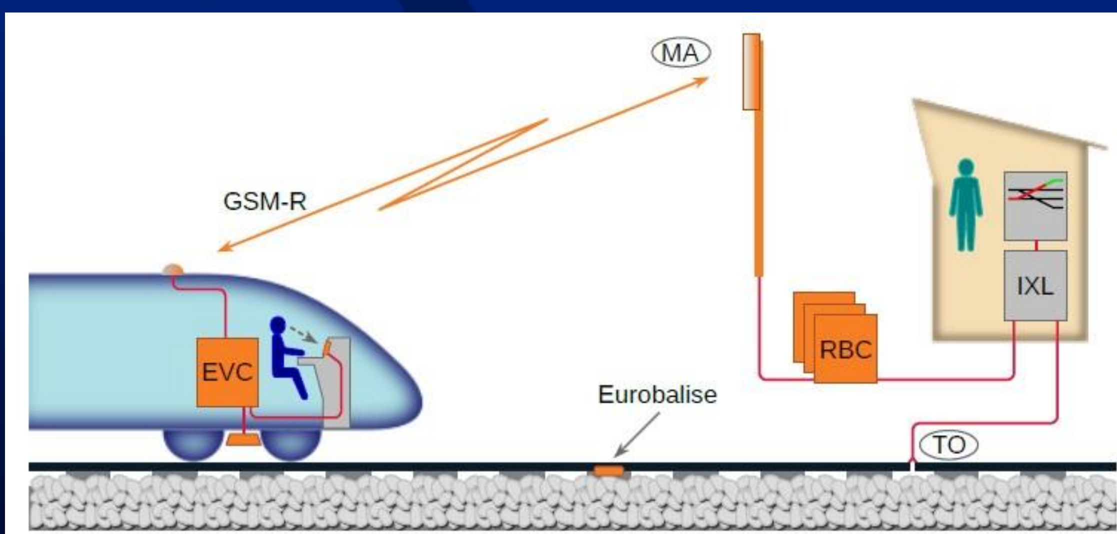
If we knew about trains, maybe …. we could do drones ….



Corridor Occupancy

Movement Authority

Radar



Verification of the American Air Traffic Management System

In Real-Time Maude?

- Guarantee safety?
- Measure reliability & performance?
- Measure and improve capacity?

➔ Qualitatively analyze safety using timed model checking?

➔ Quantitatively analyze capacity & energy?

Need to Read & Understand:

Verification of the European Rail Traffic Management System in Real-Time Maude

Ulrich Berger[a], Phillip James[a], Andrew Lawrence[b], Markus Roggenbach[a], Monika Seisenberger[a]

[a] *Swansea University, Swansea, UK*
[b] *Siemens Rail Automation UK, Chippenham, UK.*

Abstract

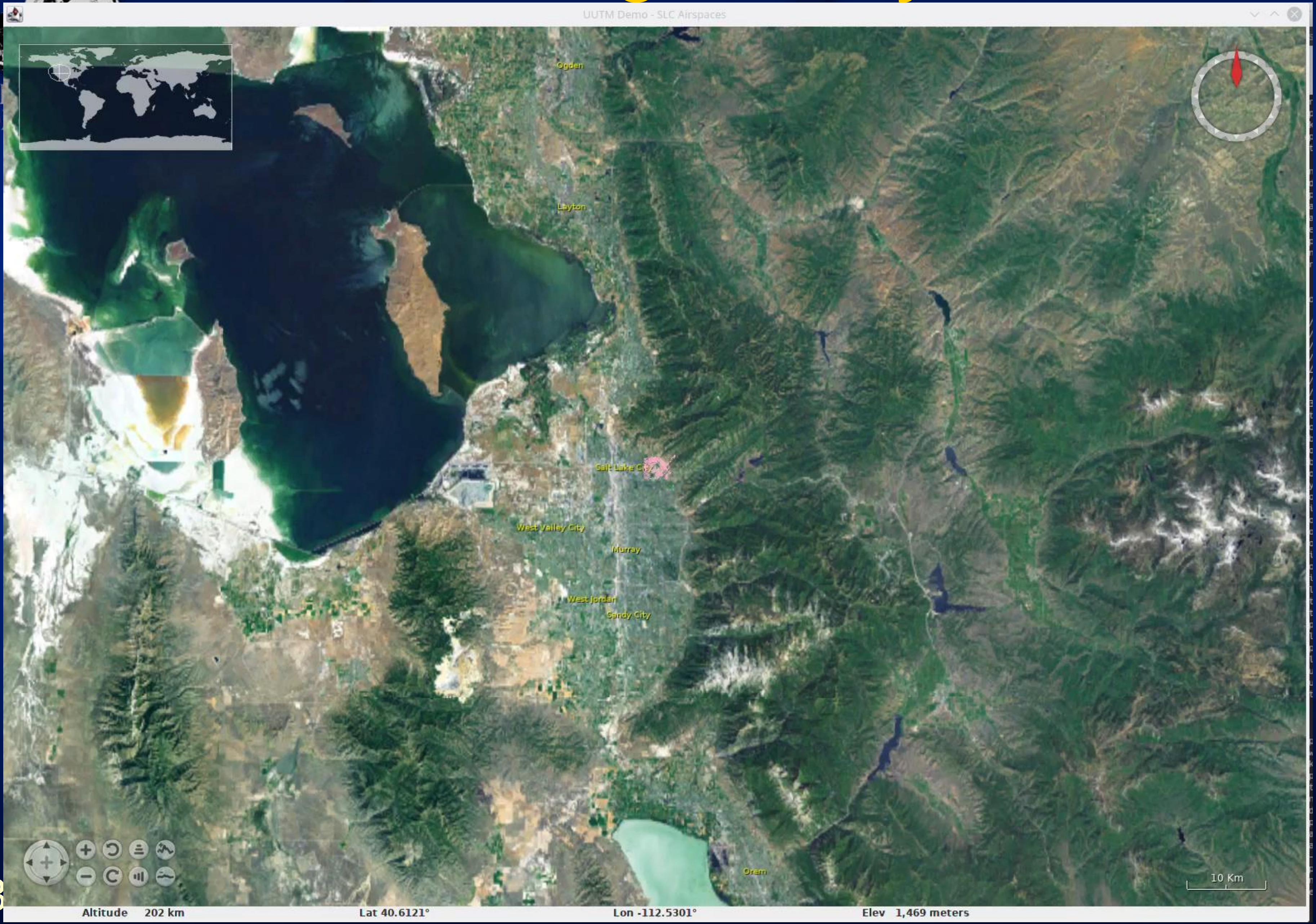
The European Rail Traffic Management System (ERTMS) is a state-of-the-art train control system designed as a standard for railways across Europe. It generalises traditional discrete interlocking systems to a world in which trains hold on-board equipment for signalling, and trains and interlockings communicate via radio block processors. The ERTMS aims at improving performance and capacity of rail traffic systems without compromising their safety.

Large-scale Simulation

- http://www.cs.utah.edu/~cem/uav/

Questions?

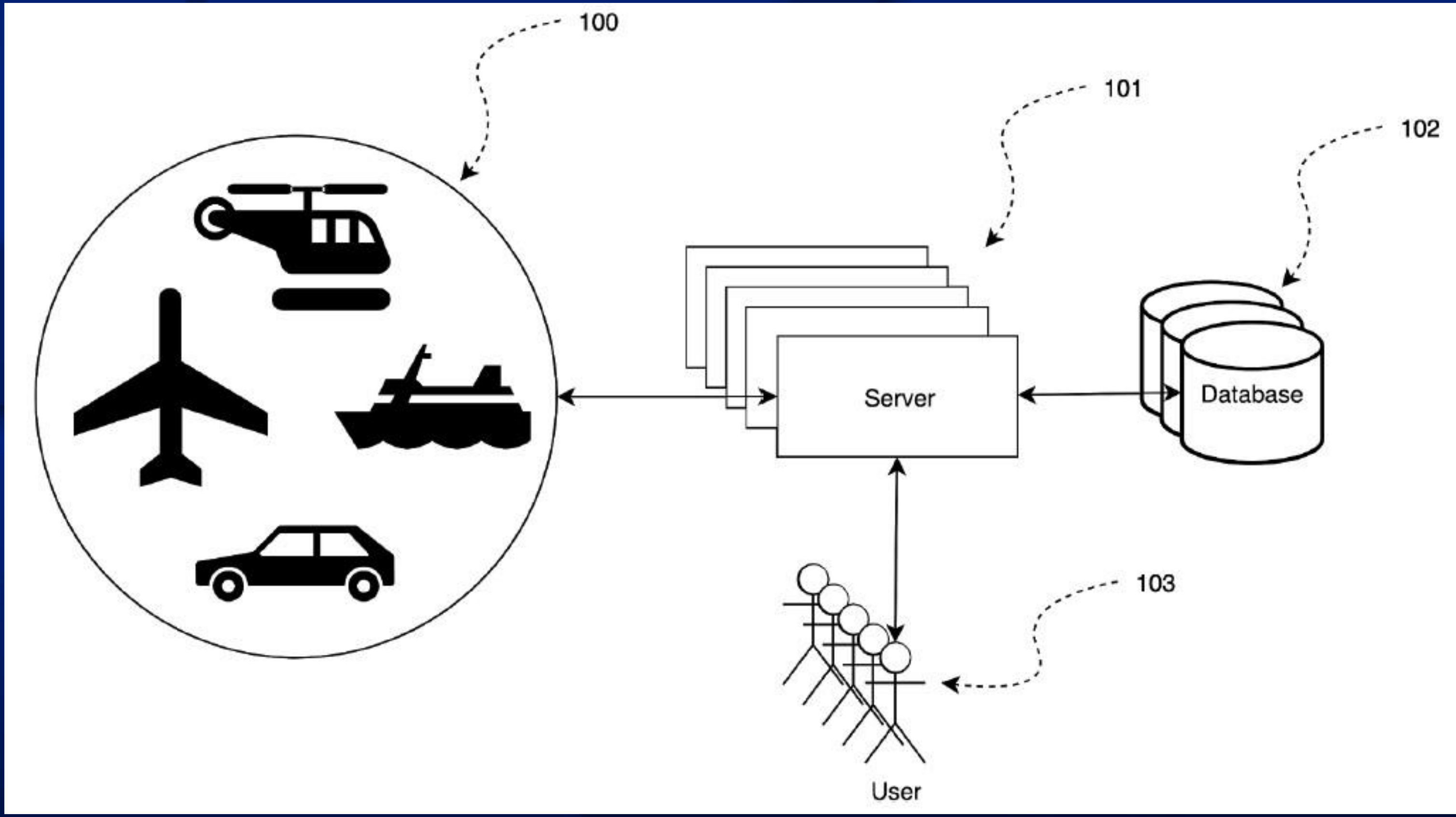
Creating Airways



SCHOOL OF COMPUTING

UTAH UAV Fleet

THE UNIVERSITY OF UTAH


100
101
102
Server
Database
103
User

Require: $r_d, r_e, r_l, path, v_g$

$r_d \leftarrow$ desired release time

$r_e \leftarrow$ earliest release time

$r_l \leftarrow$ latest release time

$path \leftarrow$ requested segment ids

$v_g \leftarrow$ speed

$seats \leftarrow$ available time slots

$l_s \leftarrow 0$ {The segment length}

for each $segment$ in $path$ **do**

  $seats_{segment} \leftarrow$ seats on segment at $t \in [r_e, r_l] + \frac{l_s}{v_g}$

  $seats \leftarrow seats_{segment} \mid seats$ {Binary OR}

  $l_s \leftarrow$ segment length

end for

$r_t \leftarrow$ open seat closest to r_d

return r_t