

Separation Assurance in Urban Air Mobility Systems Using Shared Scheduling Protocols

Surya Murthy

University of Texas at Austin

surya.murthy@utexas.edu

Co-Authors:

Ufuk Topcu, Tyler Ingebrand, Sophia Smith, Natasha Neogi, Peng Wei

AGENDA

Problem: Separation in UAM Systems

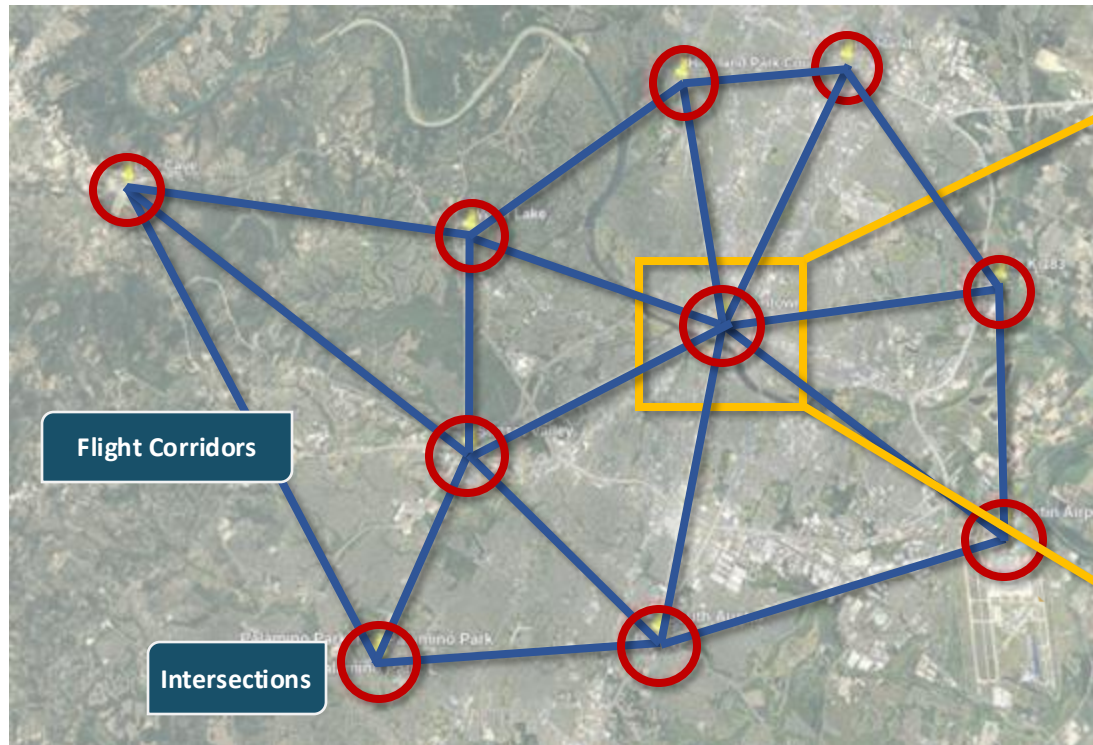
Background: Scheduling in Networking and Operating Systems

Methods: Integrating Shared Scheduling Protocols in UAM Systems

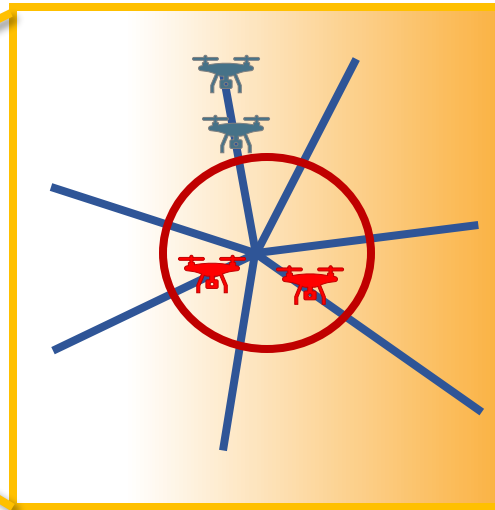
Results

Next Steps and Summary

Separation in UAM Systems



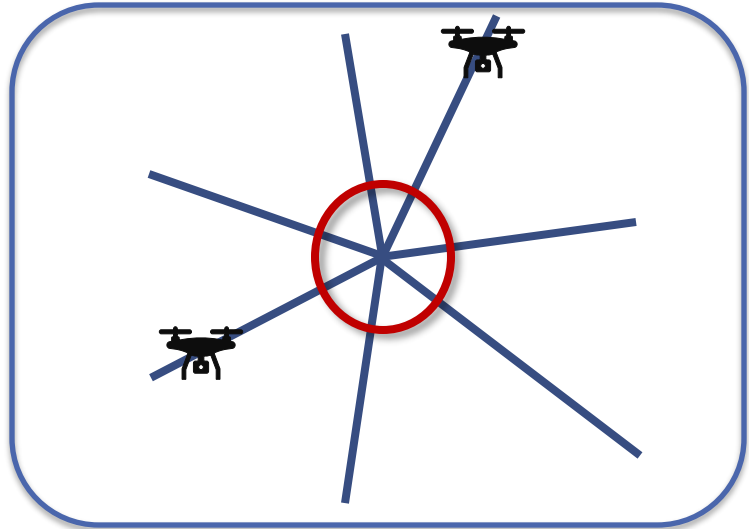
A hypothetical UAM network laid over Austin



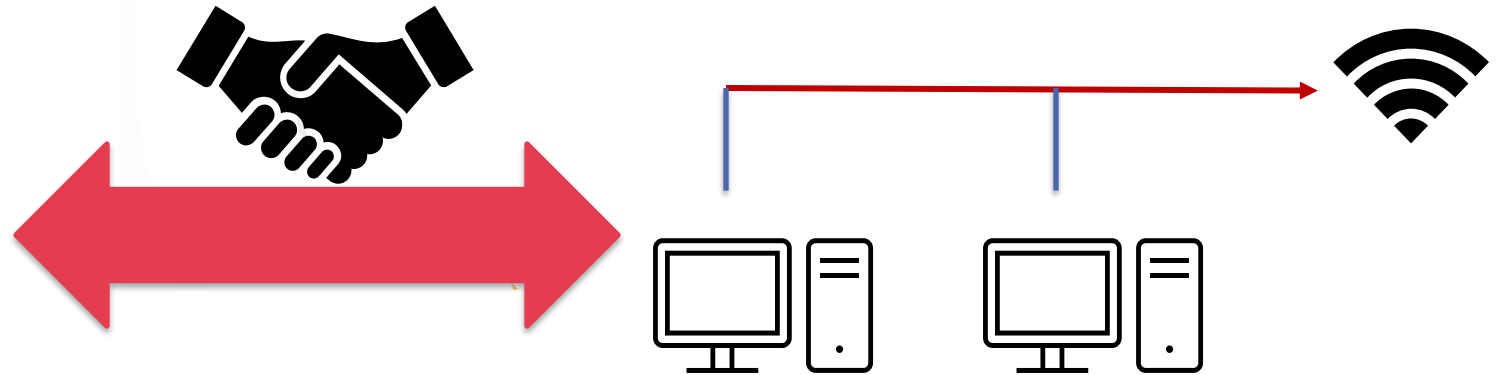
Research Question: How can we prevent Loss of Separation (LOS) using *only local observations*?

Linking UAM Scheduling to Existing Solutions

UAM



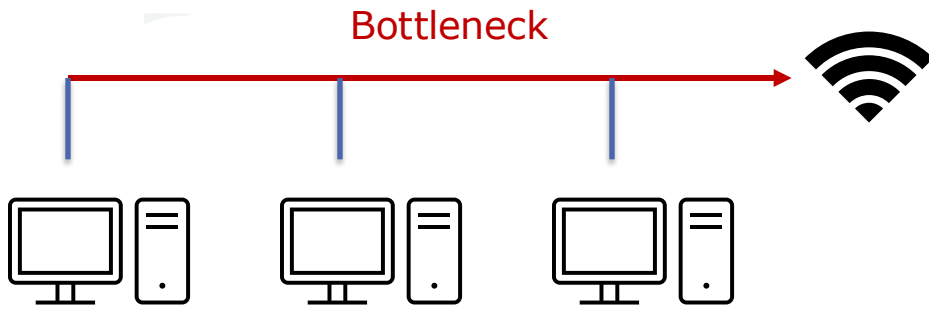
Networking and Operating Systems



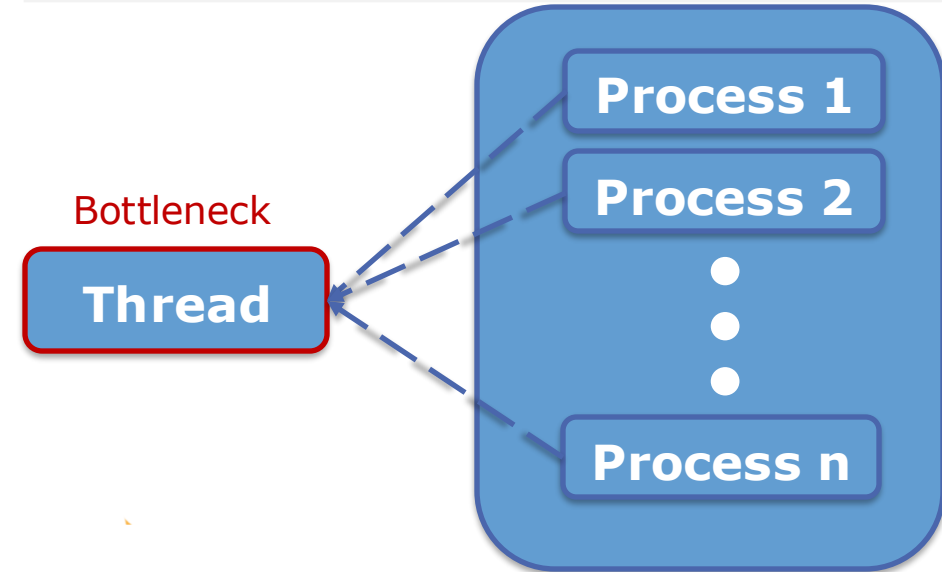
Background

Processing in Shared Networks

Ethernet Networks



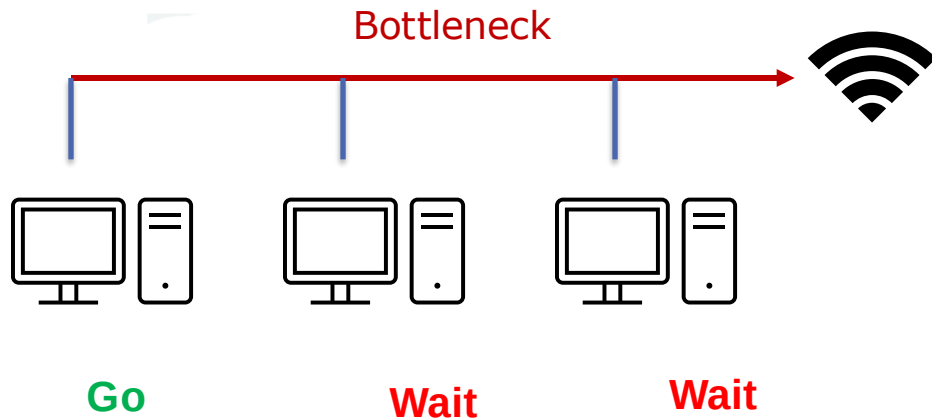
Operating Systems



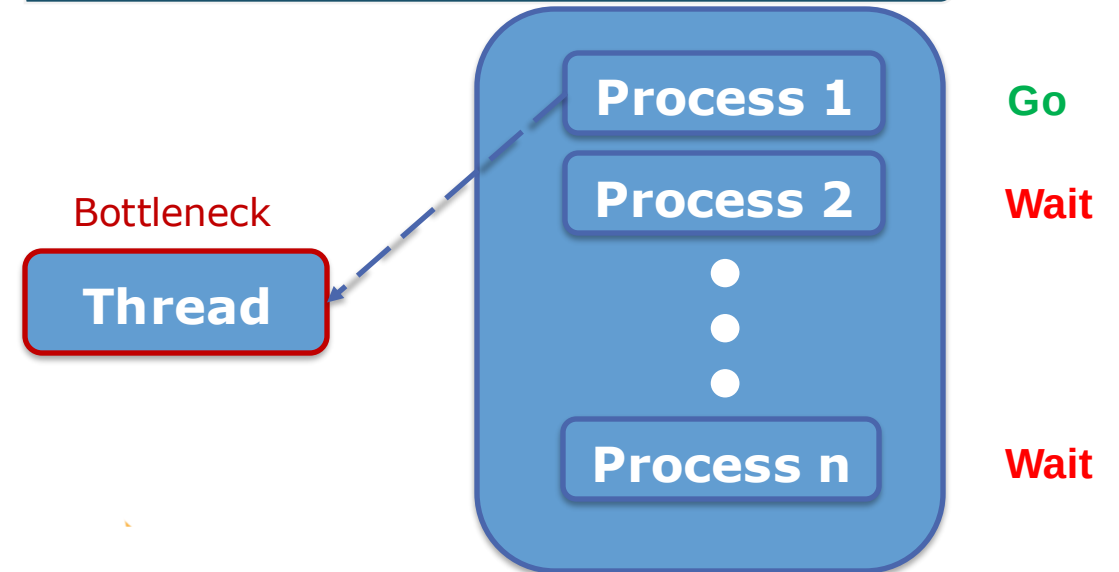
Problem: How can different processes coordinate access to bottlenecks?

Solution: Shared Scheduling Protocols

Ethernet Networks



Operating Systems



Requirements: Processes only need to access the shared resource for a **finite amount of time β** and the protocol assigns turns of at least β consecutive time steps

✓ **Guarantee: If every process takes advantage of the turn assigned by the protocol, then no process will be in repeated conflict.**

Protocols from Ethernet and Processor Communities

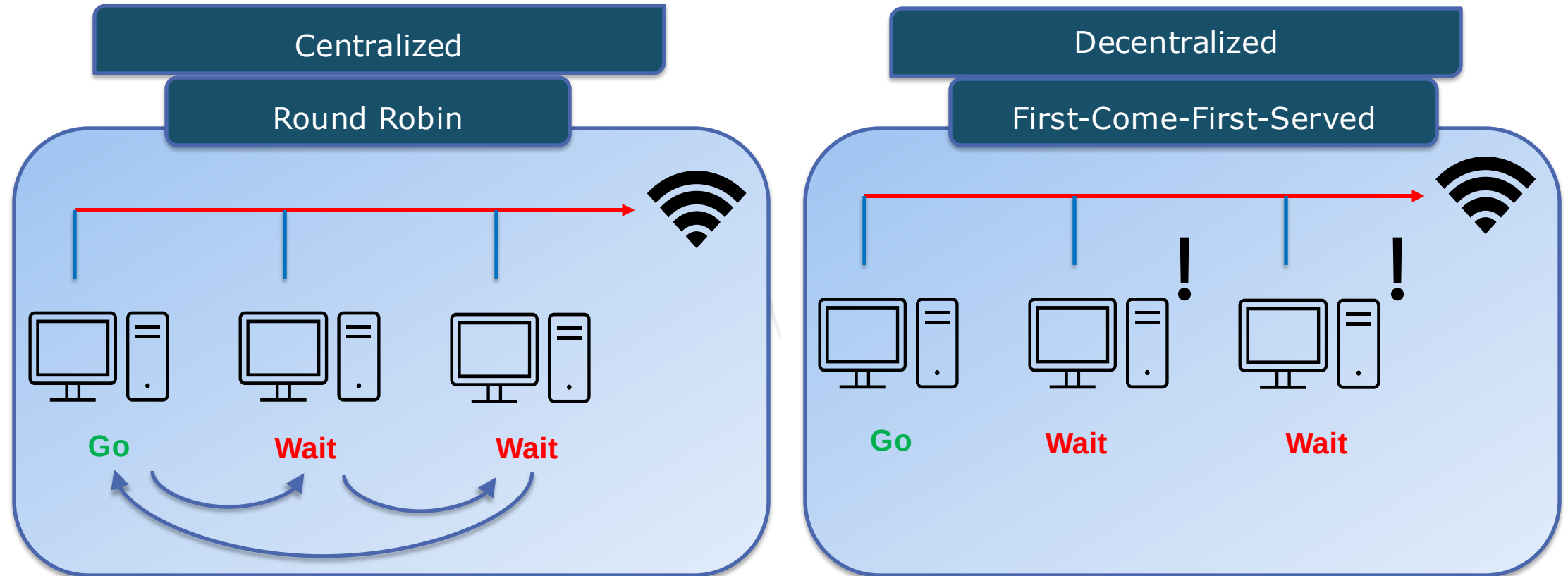
Centralized



Decentralized



Centralized and Decentralized Protocols

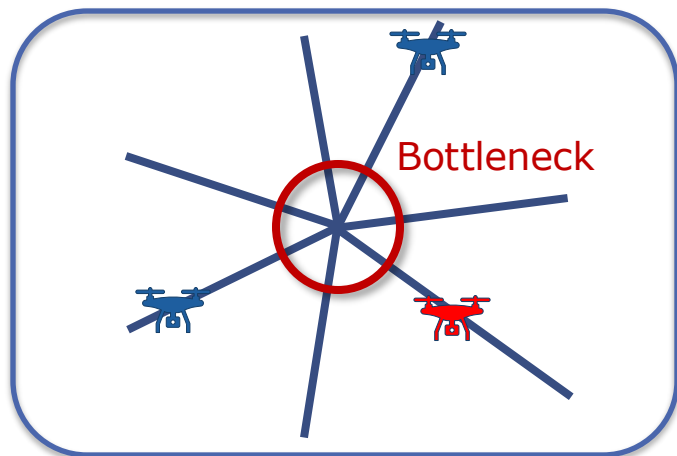


Centralized protocols use a central authority to assign turns, while decentralized protocols use local observations to assign turns for each process.

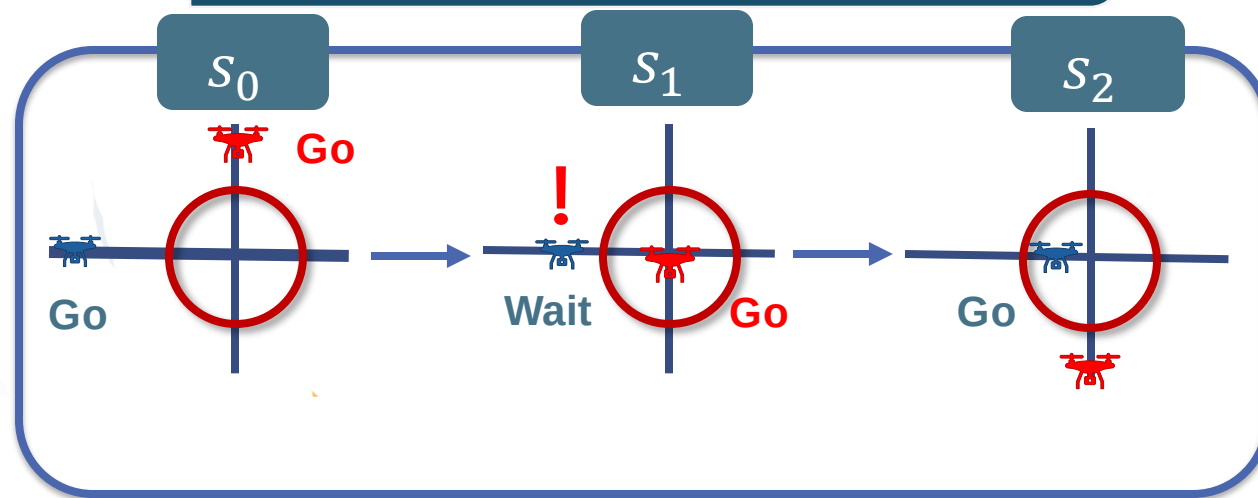
Methods

Identifying Bottlenecks in UAM Systems

Intersections act as bottlenecks, where conflicts between aircraft are likely to occur



Use shared scheduling protocols to determine when aircraft pass through intersections



By using local observations of intersection airspace, agents can efficiently resolve conflicts at intersections.

Applying Shared Scheduling Protocols

1

Requirements

Description

Aircraft on the same flight corridor maintain a safe separation from each other and can halt in midair.

2

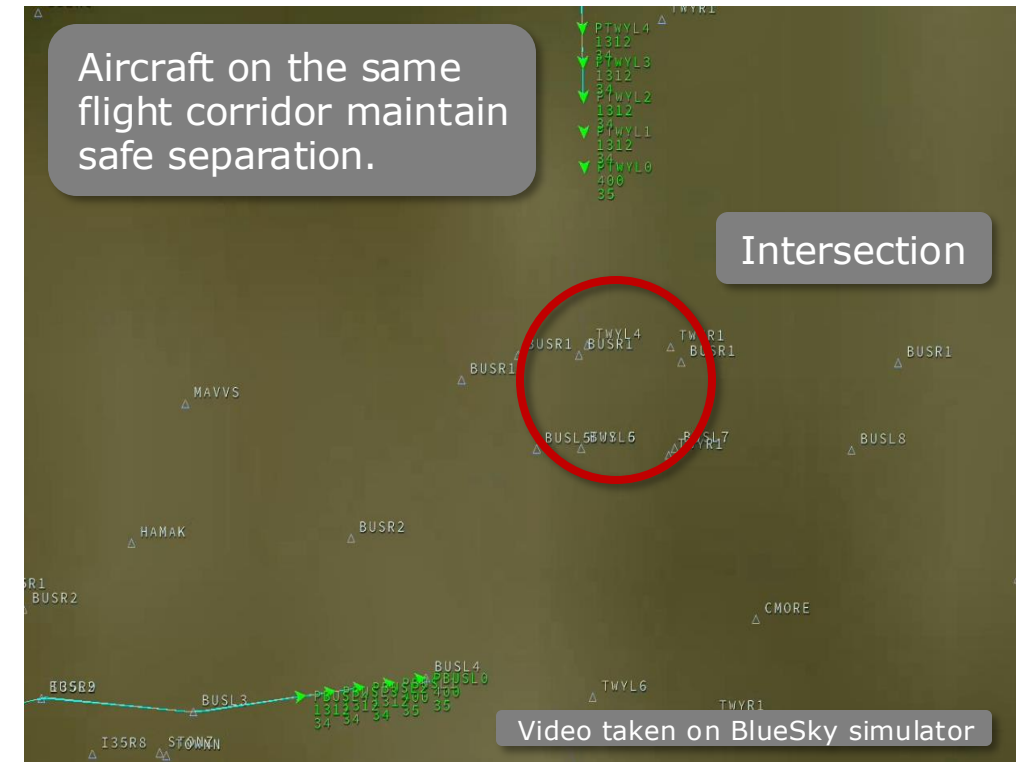
Approach

Group aircraft in the same flight corridor and assign turns to groups using shared scheduling protocols.

3

Guarantees

If all aircraft comply with the protocol, no aircraft will violate the safe separation constraint.

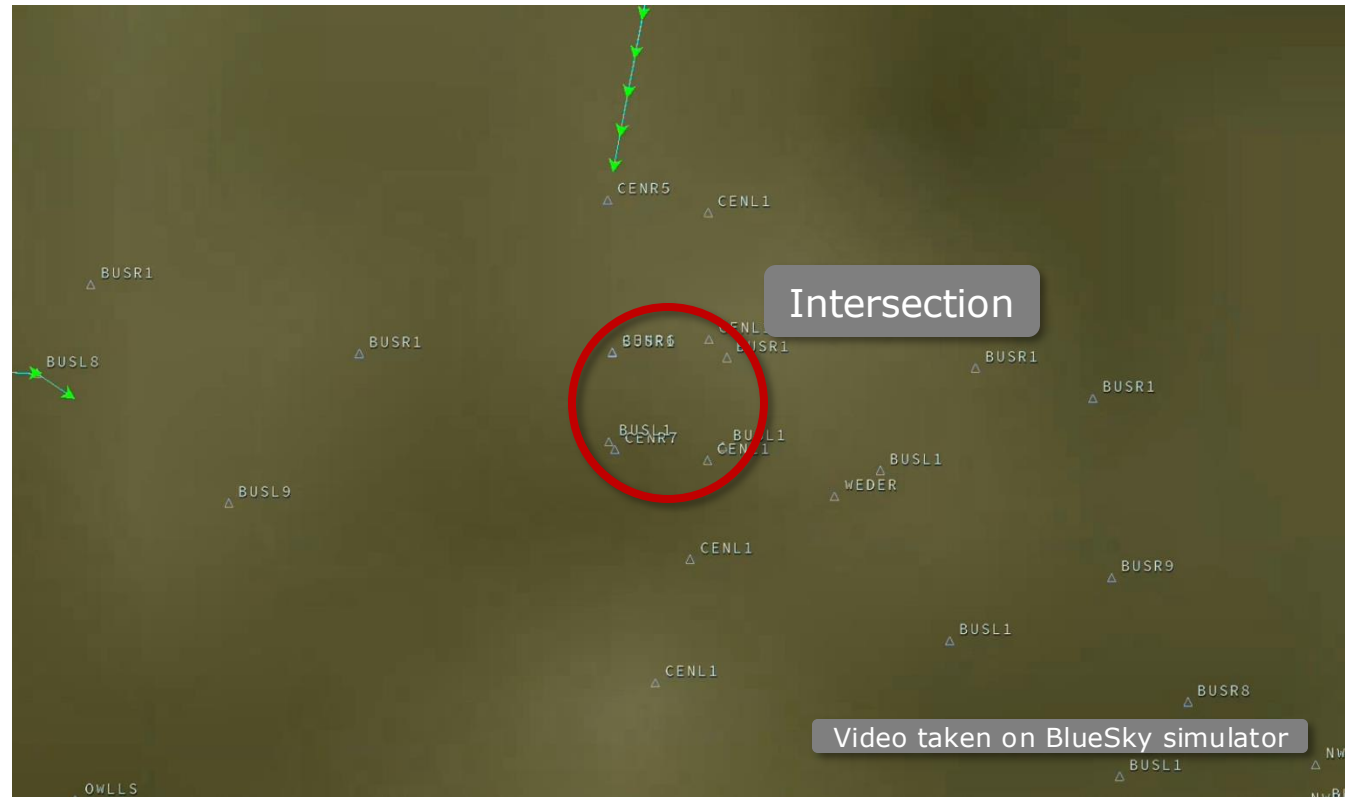


Round Robin



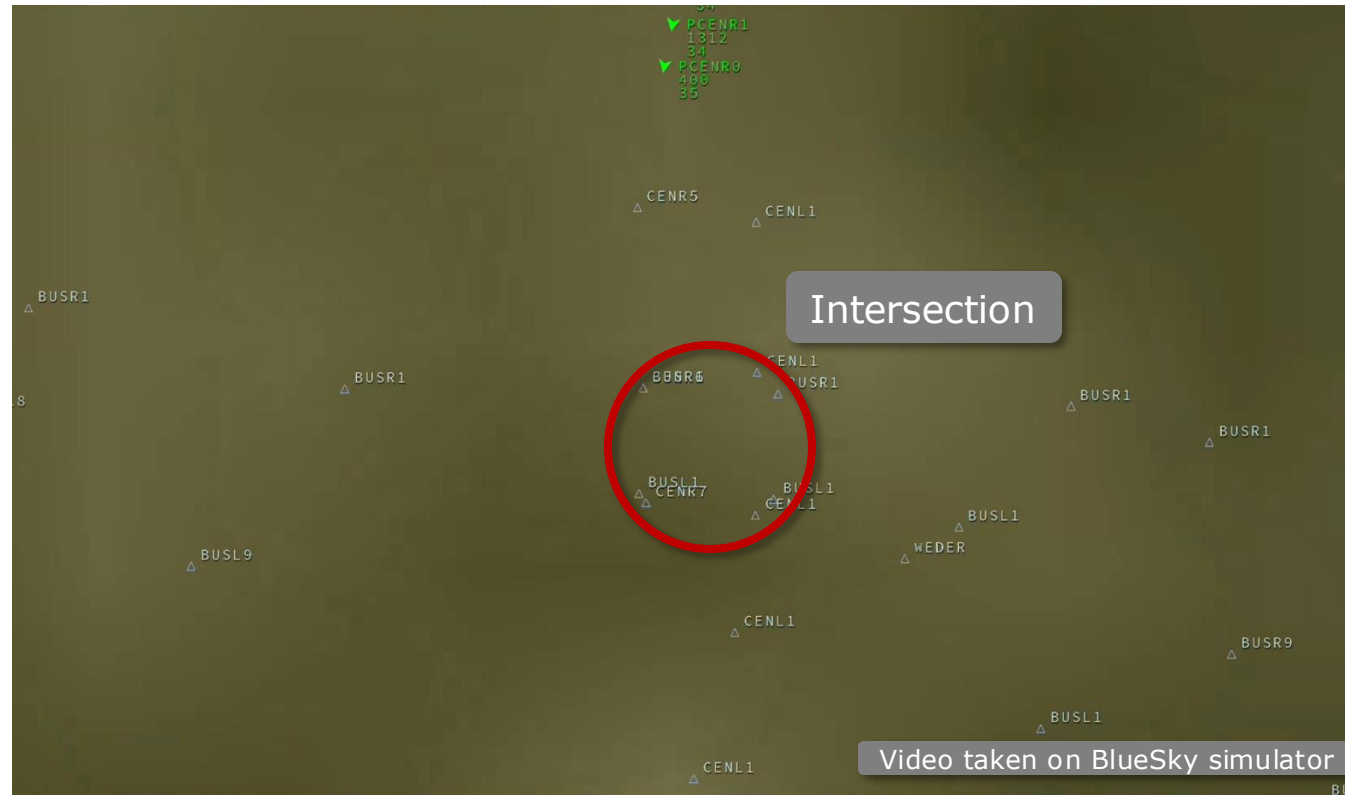
A centralized protocol that allows a subset of aircraft through the intersection at a time

Carrier Sense Multiple Access with Collision Detection



A decentralized first-come-first-served protocol. If two aircraft enter an intersection at the same time, they will halt for a random period.

Shortest Remaining Time First

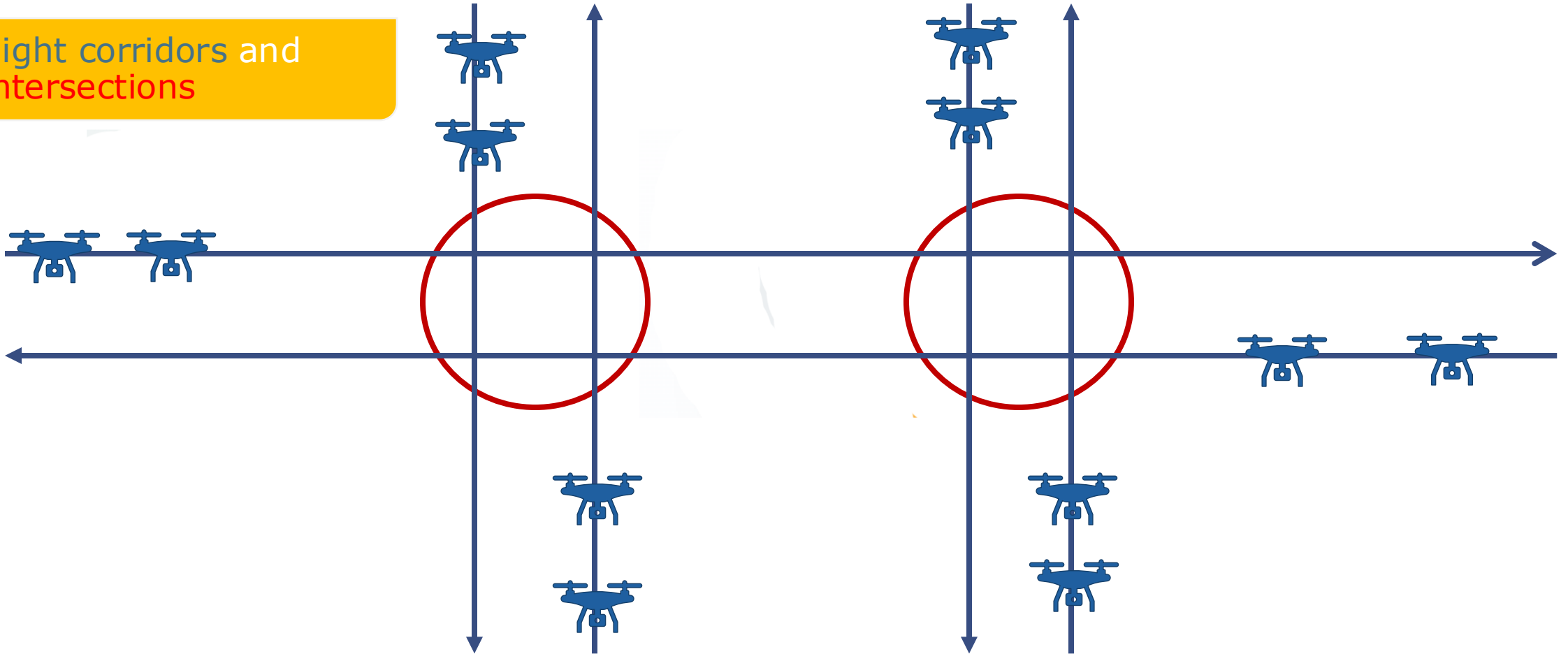


A decentralized first-come-first-served protocol. If two aircraft try to enter an intersection at the same time, the aircraft that is closest to the intersection center allowed to move first.

Results

Testing Environment

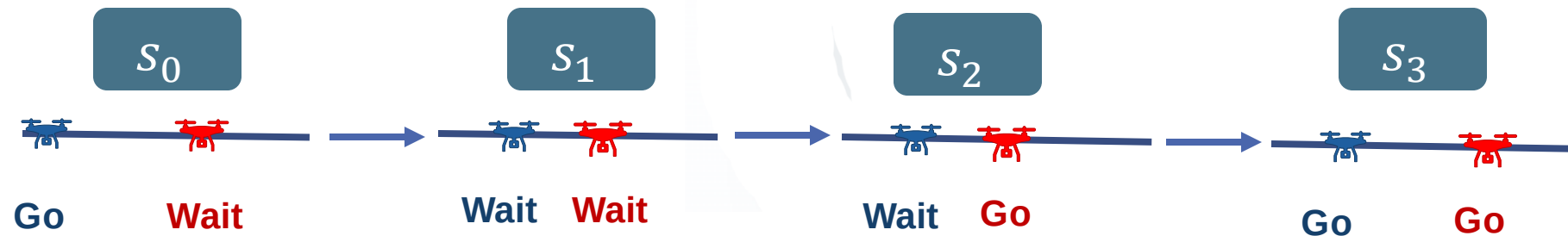
6 flight corridors and
2 intersections



Baseline Behavior

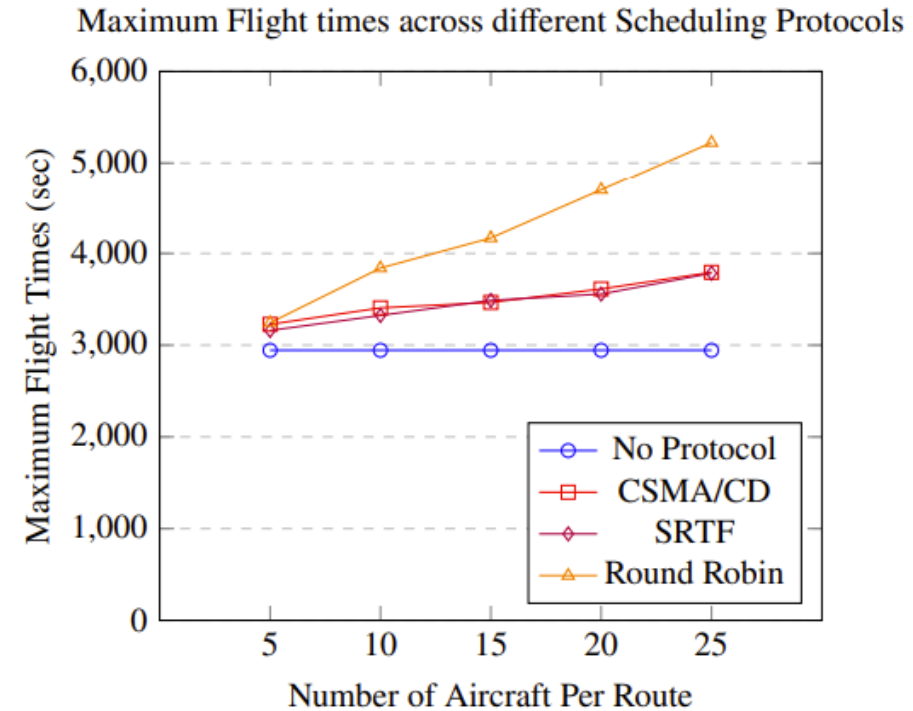
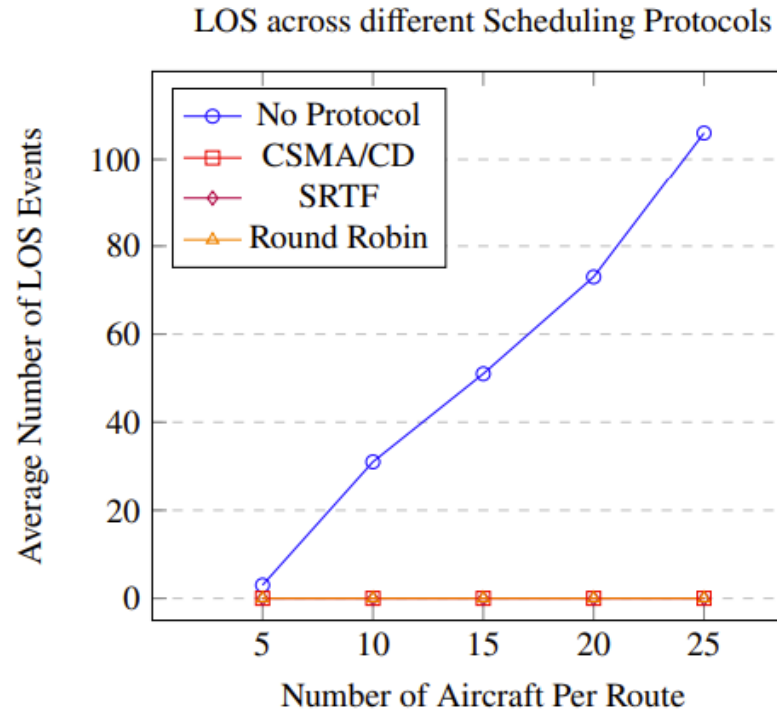
BASELINE

Aircraft moving at a constant speed and altitude with sufficient separation between aircraft on the same flight corridor at takeoff



Aircraft on the same flight corridor halt to maintain safe separation

Tradeoff between Separation Assurance & Flight Times



Shared Scheduling Protocols can effectively enforce the safe separation constraint, but lead to an increase aircraft flight times.

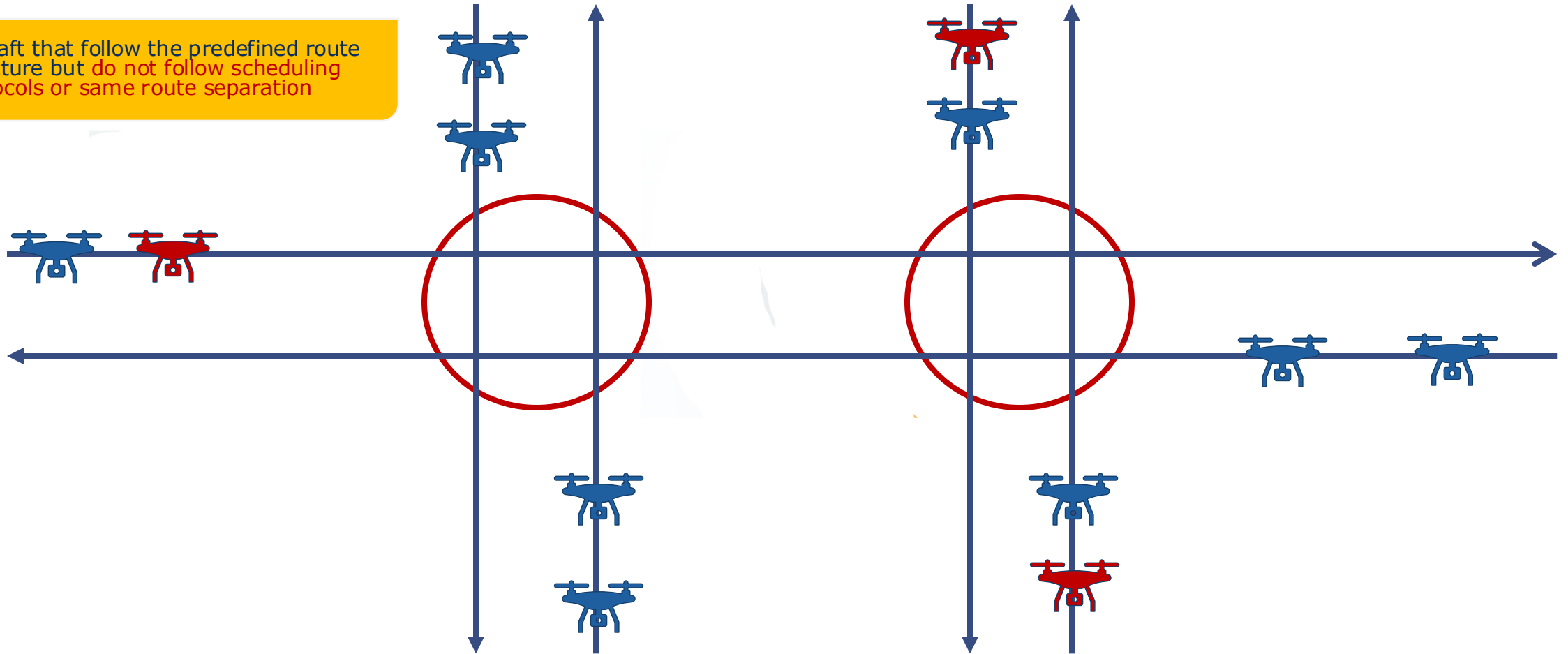
Percentage of Time Spent Halting

Protocol	5 Aircraft	10 Aircraft	15 Aircraft	20 Aircraft	25 Aircraft
CSMA/CD	4.020%	4.665%	7.119%	9.385%	11.771%
SRTF	2.393%	3.669%	6.230%	8.372%	10.820%
Round Robin	2.785%	10.653%	16.170%	21.936%	25.983%

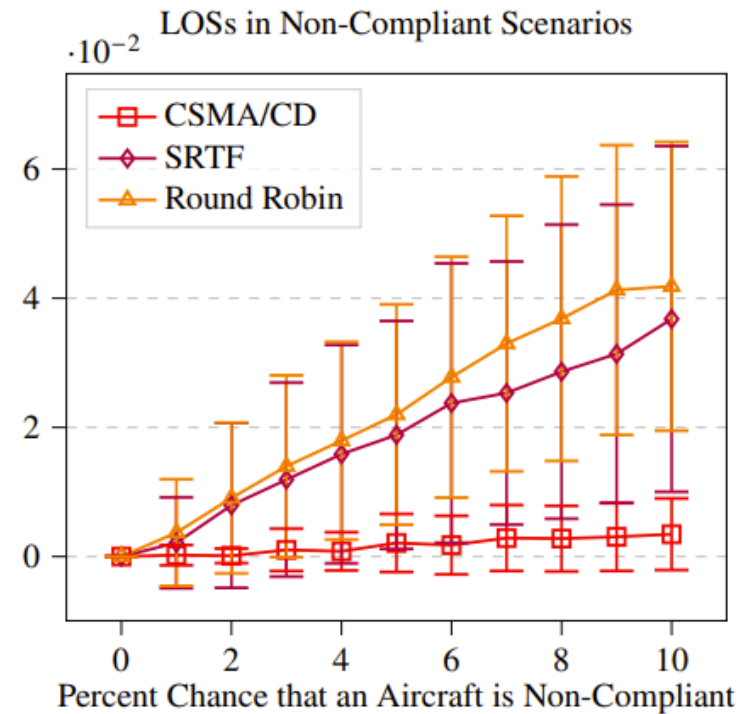
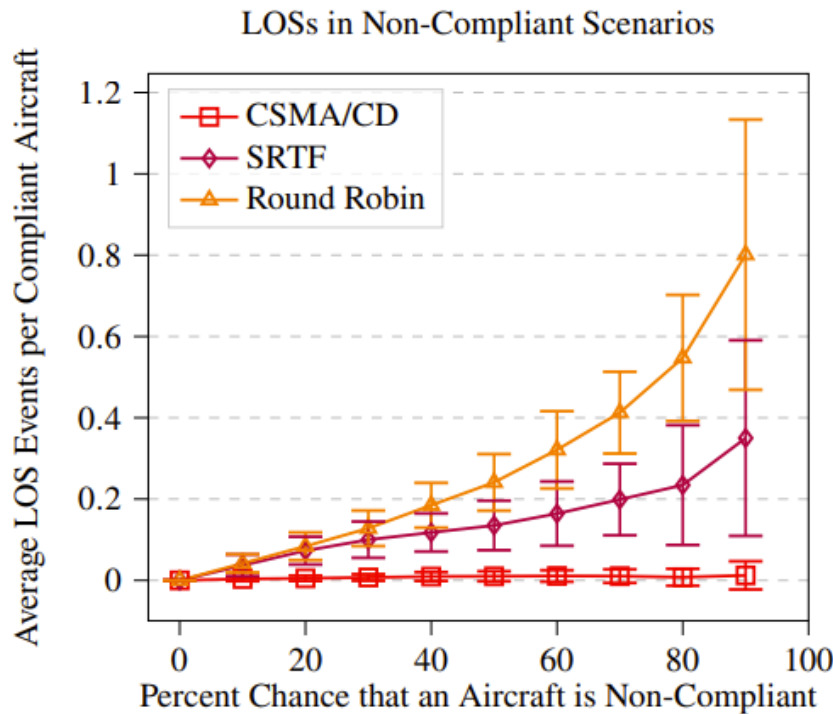
SRTF's First-Come First-Served logic combined with how it handles simultaneous access conflicts results in the lowest halting times.

Introducing Non-Compliant Aircraft

Aircraft that follow the predefined route structure but do not follow scheduling protocols or same route separation



Introducing Non-Compliant Aircraft (LOS at Intersections)



Shared Scheduling Protocols can no longer guarantee safety, but does provide improvement over baseline (100% Non-Compliance)

LOS events in Non-Compliant Aircraft

Protocol	10%	20%	30%	40%
CSMA/CD	0.28	0.36	0.45	0.51
SRTF	2.98	5.26	6.3	6.36
Round Robin	3.39	6.04	8.07	9.98
Baseline	51	51	51	51

CSMA/CD's halting logic for handling simultaneous access conflicts leads to the fewest number of LOS events

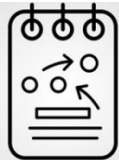
Next Steps



Use altitude adjustments to enforce safety constraints while allowing for higher traffic throughput



Consider adding traffic scheduling approaches like demand-capacity balancing to further reduce traffic congestion and halting times.



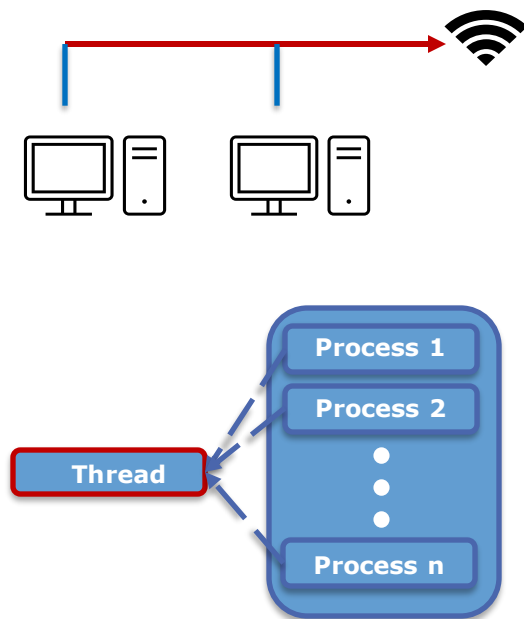
Consider more complex testing scenarios with more flight corridors and intersections.



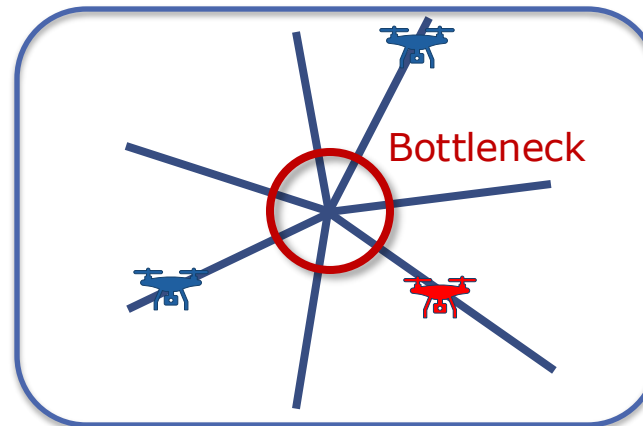
Integrate protocols with other objectives– Reduce Noise Impact heard on the ground.

Summary/Questions

Background & Introduction of concepts

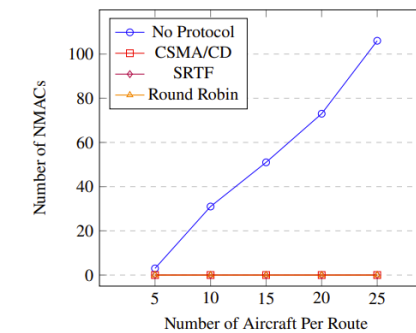


Applying concepts to UAM Environment

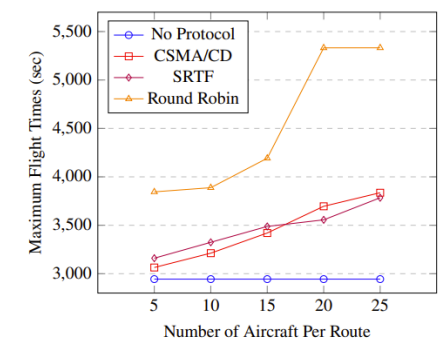


Results: Tradeoff between Separation and Flight Times

NMACs across different Scheduling Protocols



Maximum Flight times across different Scheduling Protocols



Summary/Questions

Surya Murthy
University of Texas at Austin
surya.murthy@utexas.edu