Transferring Aviation Safety Lessons to the Road

Mykel Kochenderfer and Robert Moss
Stanford University

July 12, 2021

1956 Grand Canyon Mid-Air Collision

Tribune Telephones

The Salt Lake Tribune

Showers Hit 'n' Run

Salt Lake City, Utah — Monday Morning, July 1, 1956

128 Perish in Crashes

Steel Strike Sparks Plea Of Mediator

U.S. Seeks Way To End Dispute

Smashing Winds Flay Eastern U.S.

Incredible Sight

'Copter Crew Finds Crash Bodies Piled

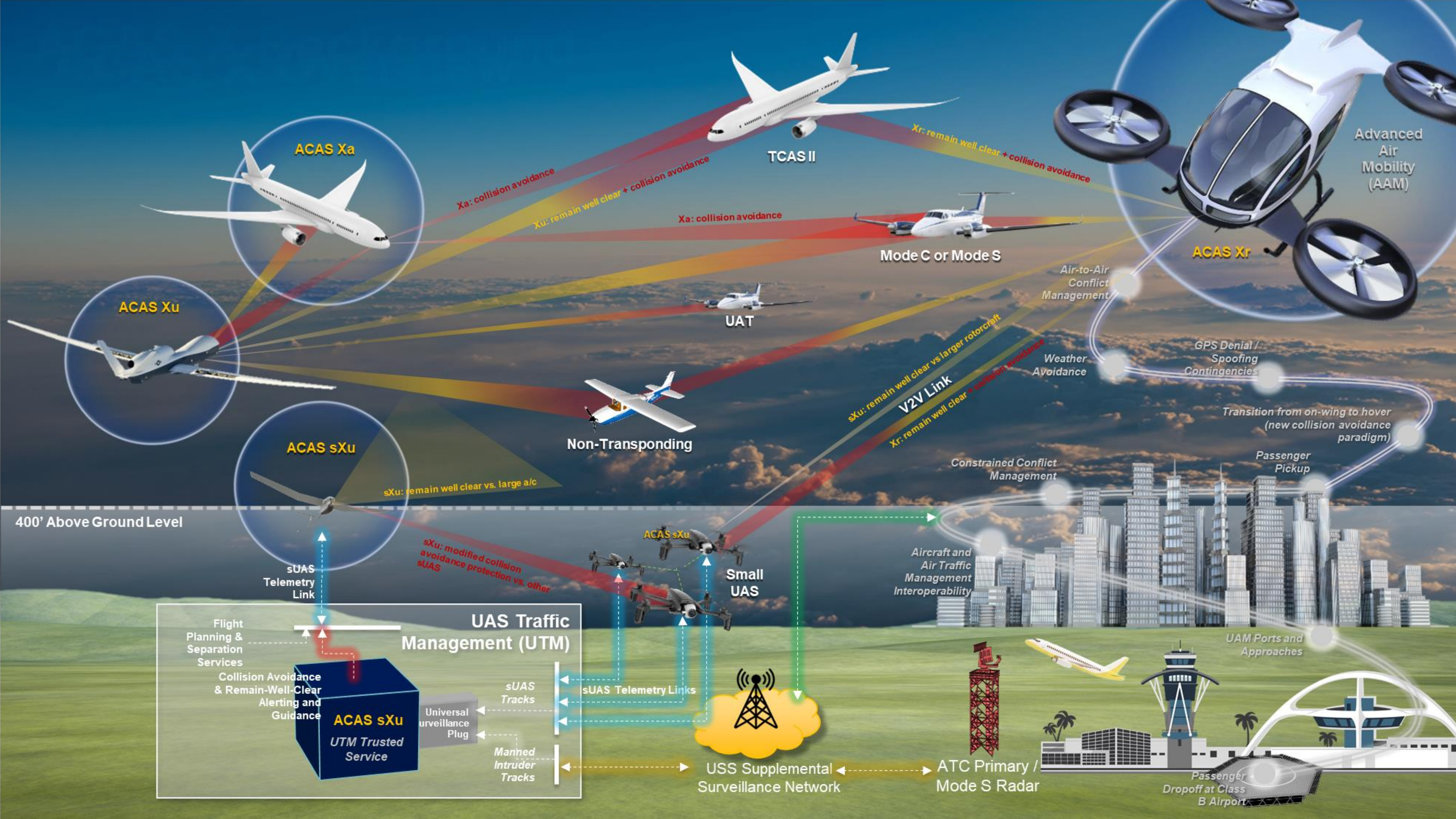
Wreckage Dots Wall of Canyon

Brothers Report

Fading Rays Of Sun Light Crash Debris

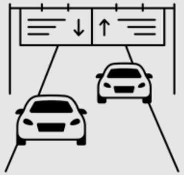
Pacific Southwest Airlines 727 / Cessna 172
San Diego, California, 1978
144 fatalities (including 7 on ground)

Aeroméxico DC-9 / Piper Archer
Cerritos, California, 1986
82 fatalities (including 15 on ground)



ACAS Xa
TCAS II
Advanced Air Mobility (AAM)
Xa: collision avoidance
Xu: remain well clear + collision avoidance
Xr: remain well clear + collision avoidance
Xa: collision avoidance
Mode C or Mode S
ACAS Xr
ACAS Xu
UAT
Air-to-Air Conflict Management
GPS Denial / Spoofing Contingencies
Weather Avoidance
sXu: remain well clear vs larger rotorcraft
V2V Link
Xr: remain well clear + collision avoidance
Transition from on-wing to hover (new collision avoidance paradigm)
Non-Transponding
ACAS sXu
sXu: remain well clear vs. large a/c
Passenger Pickup
Constrained Conflict Management
400' Above Ground Level
ACAS sXu
sXu: modified collision avoidance protection vs. other sUAS
Small UAS
Aircraft and Air Traffic Management Interoperability
sUAS Telemetry Link
UAS Traffic Management (UTM)
Flight Planning & Separation Services
Collision Avoidance & Remain-Well-Clear Alerting and Guidance
ACAS sXu
UTM Trusted Service
Universal Surveillance Plug
sUAS Tracks
sUAS Telemetry Links
Manned Intruder Tracks
USS Supplemental Surveillance Network
ATC Primary / Mode S Radar
UAM Ports and Approaches
Passenger Dropoff at Class B Airport

Modeling and Simulation

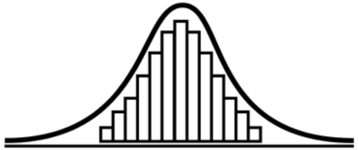


Real-World Testing

Information Sharing

Tools

Modeling and Simulation

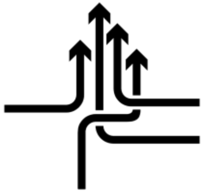


Standard cases

(mimicking operation in the real-world)

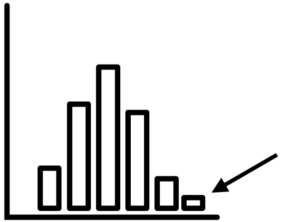
Safety critical cases

(emphasis on collisions/failures)



High-stressing cases

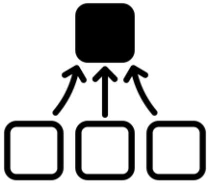
(known limitations of the system)



Long-tail edge cases

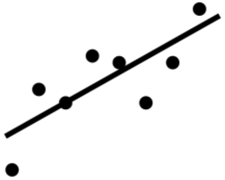
(known outliers)

Modeling and Simulation



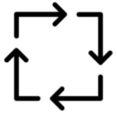
Aggregate results

(diversify quantitative metrics)



Surrogate modeling

(efficient tuning)



Repeat

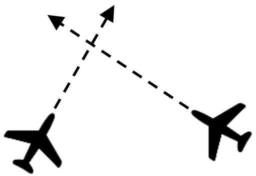
Visualizations

(high and low-level qualitative analysis)

Preference elicitation

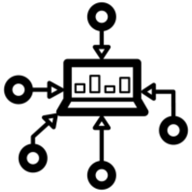
(incorporate analyst feedback)

Real World Testing



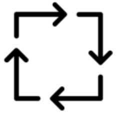
Design test cases

(covering nominal and stressing cases)

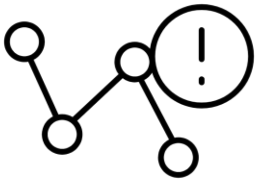


Data collection

(collect to analyze and playback)

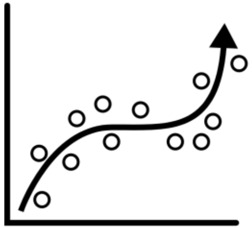


Repeat



Validate simulator

(compare playback to empirical data)

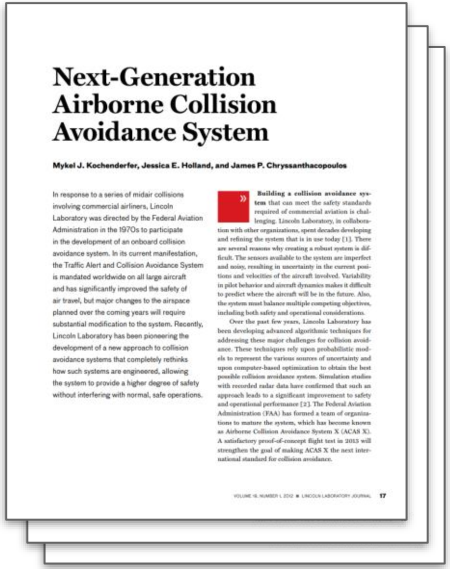


Reassess modeling assumptions

(incorporate real-world testing results)

Information Sharing and Collaboration

- Encourage publications of ideas
- Encourage open-source code sharing
- Encourage collaboration across industry, academia, and government
- Everyone benefits when ideas are shared and safety is improved



Next-Generation Airborne Collision Avoidance System

Mykel J. Kochenderfer, Jessica E. Holland, and James P. Chryssanthacopoulos

In response to a series of midair collisions involving commercial airliners, Lincoln Laboratory was directed by the Federal Aviation Administration in the 1970s to participate in the development of an onboard collision avoidance system. In its current manifestation, the Traffic Alert and Collision Avoidance System is mandated worldwide on all large aircraft and has significantly improved the safety of air travel, but major changes to the airspace planned over the coming years will require substantial modification to the system. Recently, Lincoln Laboratory has been pioneering the development of a new approach to collision avoidance systems that completely rethinks how such systems are engineered, allowing the system to provide a higher degree of safety without interfering with normal, safe operations.

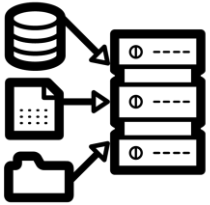
» **Building a collision avoidance system** that can meet the safety standards required of commercial aviation is challenging. Lincoln Laboratory, in collaboration with other organizations, spent decades developing and refining the system that is in use today [1]. There are several reasons why creating a robust system is difficult. The sensors available to the system are imperfect and noisy, resulting in uncertainty in the current positions and velocities of the aircraft involved. Variability in pilot behavior and aircraft dynamics makes it difficult to predict where the aircraft will be in the future. Also, the system must balance multiple competing objectives, including both safety and operational considerations.

Over the past few years, Lincoln Laboratory has been developing advanced algorithmic techniques for addressing these major challenges for collision avoidance. These techniques rely upon probabilistic models to represent the various sources of uncertainty and upon computer-based optimization to obtain the best possible collision avoidance system. Simulation studies with recorded radar data have confirmed that such an approach leads to a significant improvement to safety and operational performance [2]. The Federal Aviation Administration (FAA) has formed a team of organizations to mature the system, which has become known as Airborne Collision Avoidance System X (ACAS X). A satisfactory proof-of-concept flight test in 2013 will strengthen the goal of making ACAS X the next international standard for collision avoidance.

Tools

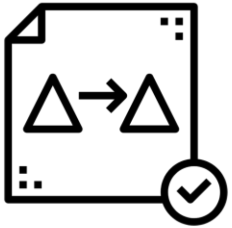
Modeling and simulation

(modular to extend to other use cases)



Data generation

(generate millions of example scenarios)



Validation

(stress test the safety system efficiently)

Visualization

(experiment analysis and publications)

Summary

- Automotive industry can learn from history of aviation safety
- Optimize and iterate in simulation, then validate modeling assumptions through real-world testing
- Emphasis on *sharing information* through publications and open-source code
- Create reusable and open software tools along the way; helps your team and the field as a whole.

SISL
Stanford Intelligent Systems Laboratory

sisl.stanford.edu | mykel@stanford.edu