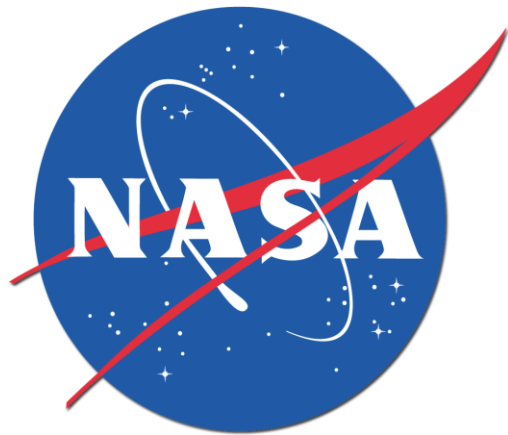


# Enabling Novel Collaborative Control Paradigms between Humans and Machines in Electric Vertical Takeoff and Landing (eVTOL) Aircraft



**Dr. Natasha Neogi, NASA Langley Research Center**

Dr. Mallory Graydon, NASA Langley Research Center

Dr. Jon Holbrook, NASA Langley Research Center

Mr. Jeffrey Maddalon, NASA Langley Research Center

Mr. Frank McCormick, Certification Services, Inc. (ret.)

[natasha.a.neogi@nasa.gov](mailto:natasha.a.neogi@nasa.gov)

NASA Langley Research Center, Hampton, Virginia, USA

# Outline

- What is human machine teaming (HMT)?
- What is (design) assurance?
- What do we need to know to build resilient teams with novel HMT paradigms?
- What lessons were learned?
- How do we move forward?

# What is human machine teaming?

# Human Machine Teaming

- Describes one or more machine agents and one or more human agents working in tandem interdependently to achieve a collective goal [1]
  - A machine agent is non-organic and powered by a computational algorithm
- To function in the role of a "teammate" with a human, a machine system must be able to perform its roles with some degree of operational independence [2]
  - Teammates need a level of autonomy—freedom to act and decide independently of others and the capabilities to scan their environment, analyze it, make decisions based on their assigned goal, and learn from what happens

1. O'Neill, T. A., Flathmann, C., McNeese, N. J., and Salas, E., "Human-Autonomy Teaming: Need for a Guiding Team-Based Framework?" Computers in Human Behavior, Vol. 146, (107762), 2023.

2. National Academies of Sciences, Engineering, and Medicine, Human-AI Teaming: State of the Art and Research Needs, The National Academies Press, Washington DC, USA, 2021.

# Tool vs. Teammate



"Zojirushi-riisinkeitin in action" by [villoks](#) is licensed under [CC BY-SA 2.0](#). Creative Commons, used with permission of villocks.

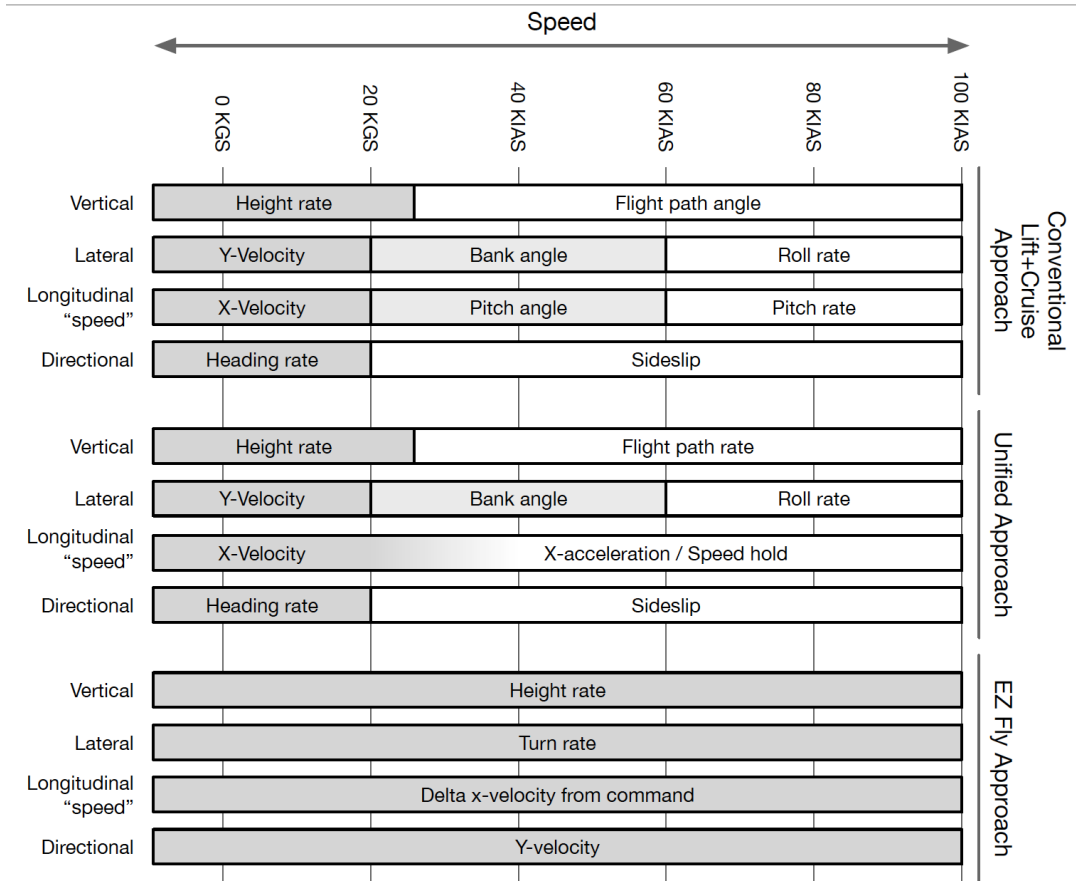


"B747-4D7 | Thai Airways | HS-TGR | HKG" by [Christian Junker | Photography](#) is licensed under [CC BY-NC-ND 2.0](#). Creative Commons, used with permission of Christian Junker.

# Function Allocation in HMT

- Novel function allocations can give rise to authority-responsibility mismatches
  - One agent is authorized to perform an activity, but a different agent is responsible for its outcome
  - Leads to extra communication and monitoring tasks
  - Example: proposed 'automated' hover mode when eVTOL is transitioning from wingborne lift to vertical lift
- Allocate functions to empower the HMT and not to set individual agents (or team) up to fail
  - Active monitoring vs. passive monitoring
  - Design independent cognitive limitations vs design dependent tools/tasks that are misaligned with cognitive systems

# Novel eVTOL Flight Control Paradigms [4]



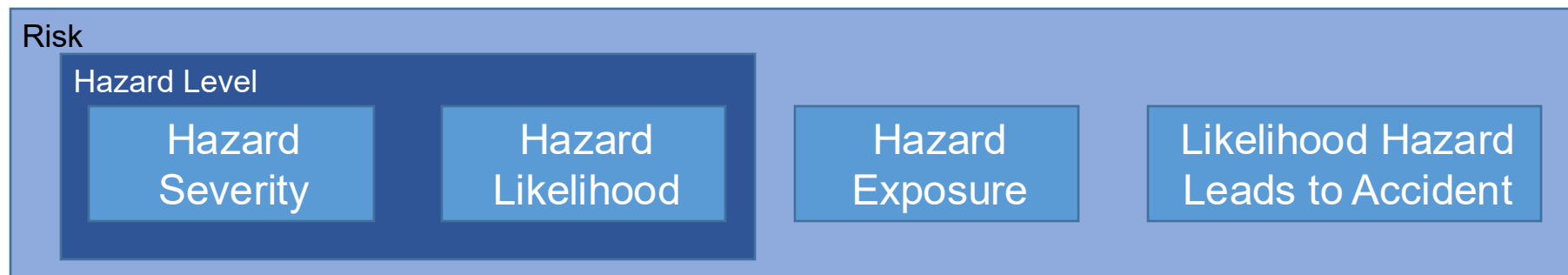
- Substitution-based function allocation strategies can paint a design-invariant picture of human capabilities and limitations
- Requires consideration not only of failure prevention but also preparing for and recovering from both expected and unexpected failures.

4. Feary, M., Kaneshige, J., Haworth, L., Lombaerts, T., Shish, K. H., Iwai, N., and Archdeacon, J., "Evaluation of Novel V/STOL Aircraft Control for Expected AAM Operations," Proc. AIAA Aviation, 2023.

# What is (design) assurance?

# Safety Associated Definitions [5]

- Safety: Freedom from accidents or loss events
- Accident: An undesired and unplanned event that results in a loss (including loss of human life or injury, property damage, environmental pollution, etc.)
- Hazard: A system state or set of conditions that, together with a particular set of worst-case environmental conditions, will lead to an accident (loss event)
- Hazard Level: A function of the hazard severity (worst case damage that could result from the hazard given the environment is in its most unfavorable state) and the likelihood (qualitative or quantitative) of its occurrence
- Risk: A function of the hazard level combined with (1) the likelihood of the hazard leading to an accident and (2) hazard exposure or duration

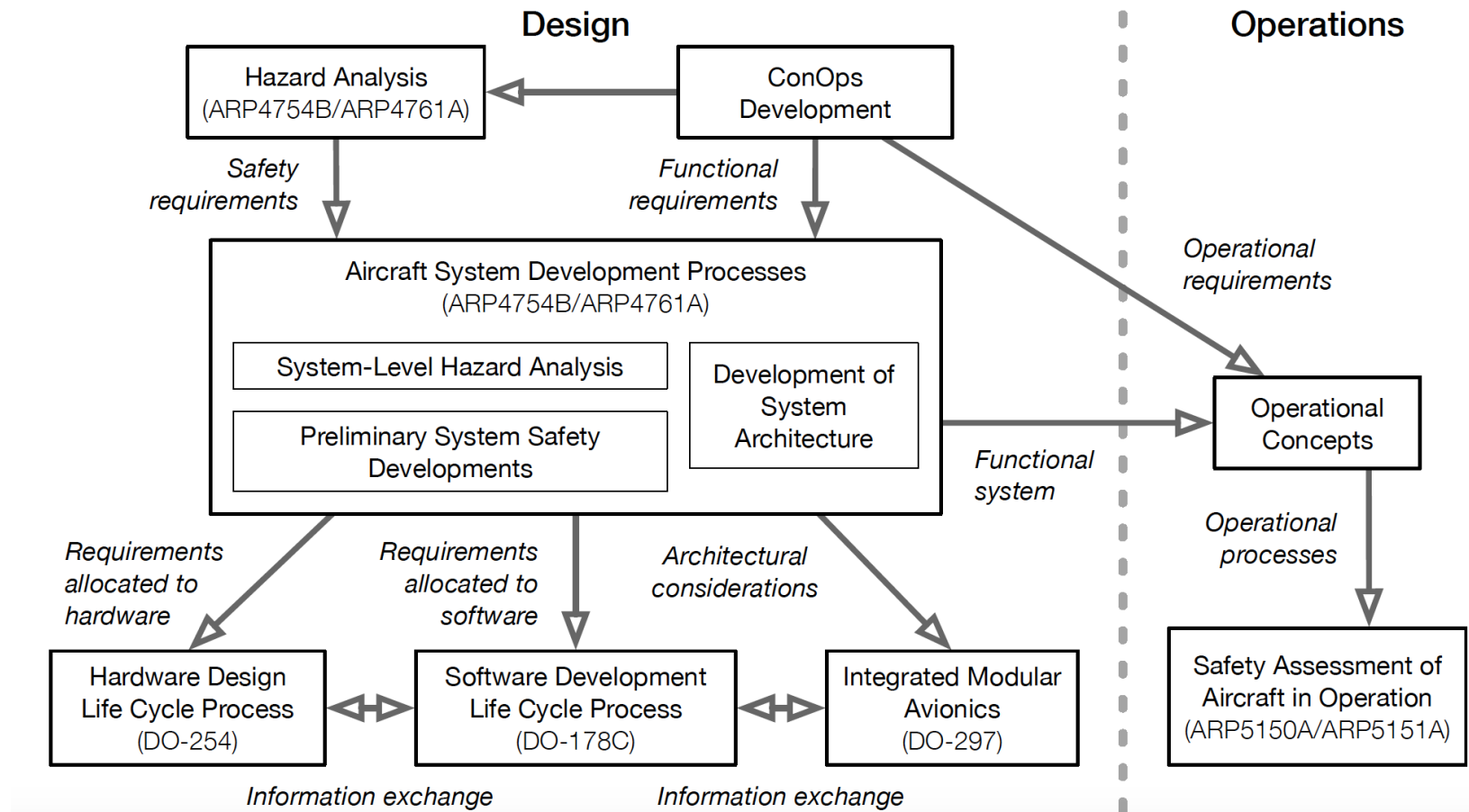


5. Leveson, N.G., 1995. Safeware: system safety and computers. Association for Computing Machinery, New York, NY, USA.

# Safety Engineering Process

- Hazard assessment plays a central role in safety engineering
  - Hazardous states are states of the thing being designed that designers must do something about (e.g., eliminate or mitigate)
    - Safety processes during aircraft design, aircraft operation, air traffic control operations, and airspace-related rulemaking concern different “things” ...
    - The “state” is a state of a thing over which you have *design authority* (i.e., whether a high-voltage part is exposed, but not where people put hands)
  - Hazardous states must be *identified* and their *severity* assessed
  - Hazardous states must be addressed in accordance with their severity
    - A hazardous state might be eliminated by design changes
    - *Safety requirements* might be defined to forestall entry into a state, or detect entry into the state, or facilitate exit from the state

# Aircraft Development & Safety Assurance Lifecycle



# Human Considerations in Design Assurance

- Neither crew reactions to a failure condition nor the impact on crew are always as designers anticipate
  - No well-established industry consensus best practices document describing the process for validating human–machine interactions of the aircraft safety assessment or establishing what kinds of justification for assessments are needed in those cases
- Novel HMT concepts for eVTOLs represent a departure from traditional cockpit allocation of tasks to humans and machines
  - Tool vs. Team
  - Expect an increase in the number of crew response and impact inputs to functional hazard analysis that cannot be established solely by reference to long-established history
  - The inputs in these cases will need to come from dedicated research studies and fresh empirical evidence

What do we need to know to  
build resilient teams for novel  
HMT paradigms?

# Proposed UAM vehicles and operations

- Proposed Urban Aerial Mobility (AAM) missions are inspiring novel technology
  - Battery electric propulsion
  - VTOL capability with wing-borne cruise for efficiency
  - Simplified vehicle operations (SVO) and unified controls
    - One inceptor, two modes
    - Reduced training needs
  - *M-to-n* operations

- UAM missions might involve:
  - Flight over dense urban areas
  - UAM traffic management
    - UAM traffic corridors



NASA Lift Plus Cruise Concept Vehicle

# Resilience

- Systems that operate successfully in dynamic environments for which the range of possible states or conditions is not fully defined exhibit resilient performance
  - Fielded systems have largely relied upon humans to provide this resilience in the event of automation failure or a situation not anticipated or accounted for by automation designers



[Zeppelin Leader](#) by [jurvetson](#) is licensed under [CC BY 2.0](#).

# Functional Allocation and Resilience



- (Novel) HMT paradigms must act to enable (or at least preserve) resilience
  - Consider interdependencies across functions and tasks allocated to different agents
  - Consider human *and* machine simultaneously in allocation to support multi-agent interaction

# Challenges in Building Resilient HMTs

1. Limited data on how humans succeed in engendering resilience (i.e., not just looking at how humans fail, but what they do to make systems safe in everyday operations)
2. Limited understanding of how task allocations of functions between humans and machines foster human resilience (i.e., what interdependences between tasks help require a specific function allocation to enable human resilient performance)
3. Lack of interdisciplinary research and interaction needed to characterize how humans and machines affect each other when performing complex, safety-critical functions in unbounded and uncertain contexts

What (selected) lessons were learned?

# Design Assurance (I)

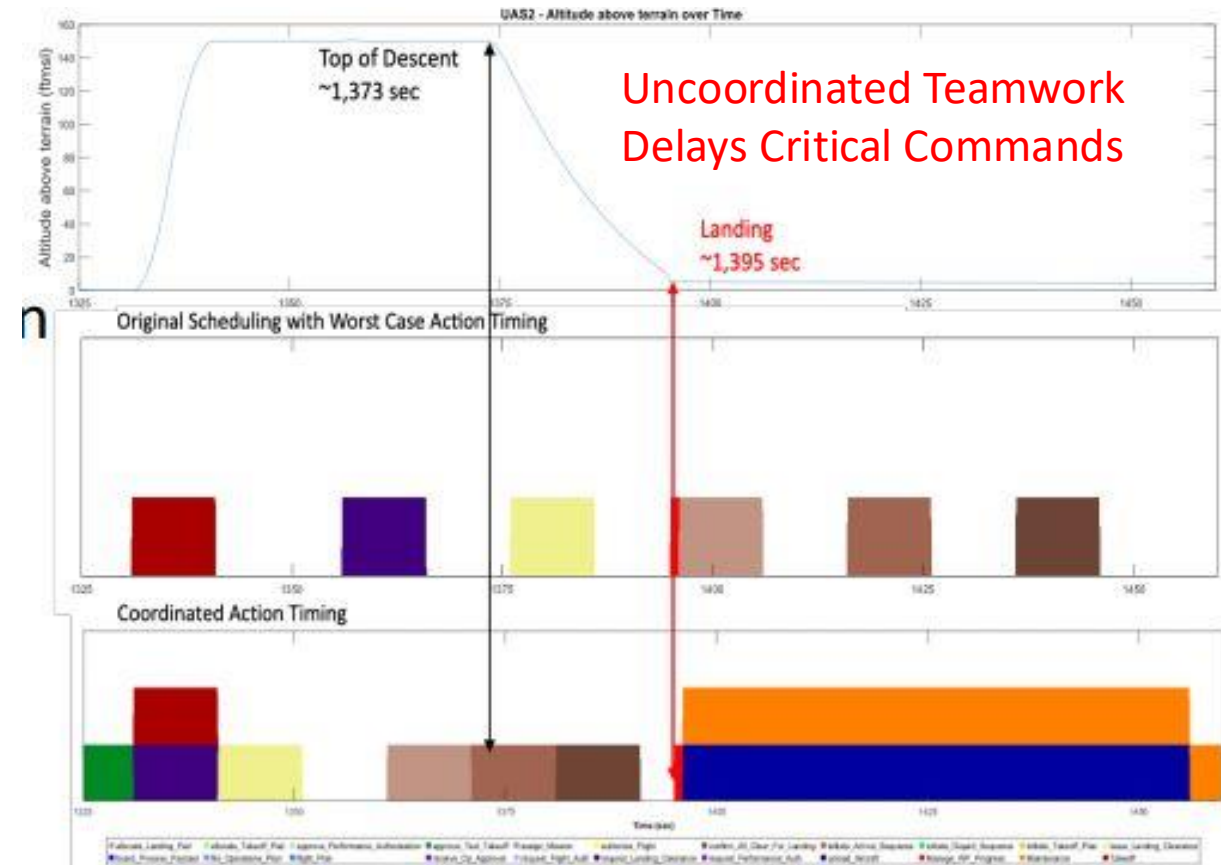
- Design assurance will keep you happy, but dissimilarity will keep you alive
  - **Functional** vs. design
  - **Parallel** vs. sequential
- Be wary of unstudied new control paradigms (including unified control) if they have not been evaluated for safety and handling qualities
  - Fundamental architectural differences help
  - If exotic flight controls paradigms misbehave, there should be some other plan B to allow for recovery/controlled descent
- Principled never-give-up approaches help
  - Any layer of defense can fail, no matter how much redundancy you put into it
  - Accept degraded functionality for each layer of fallback
  - Need to build in protections against cascading failures
  - Expand your envelope to consider the value of unconventional layers of protection

# Design Assurance (II)

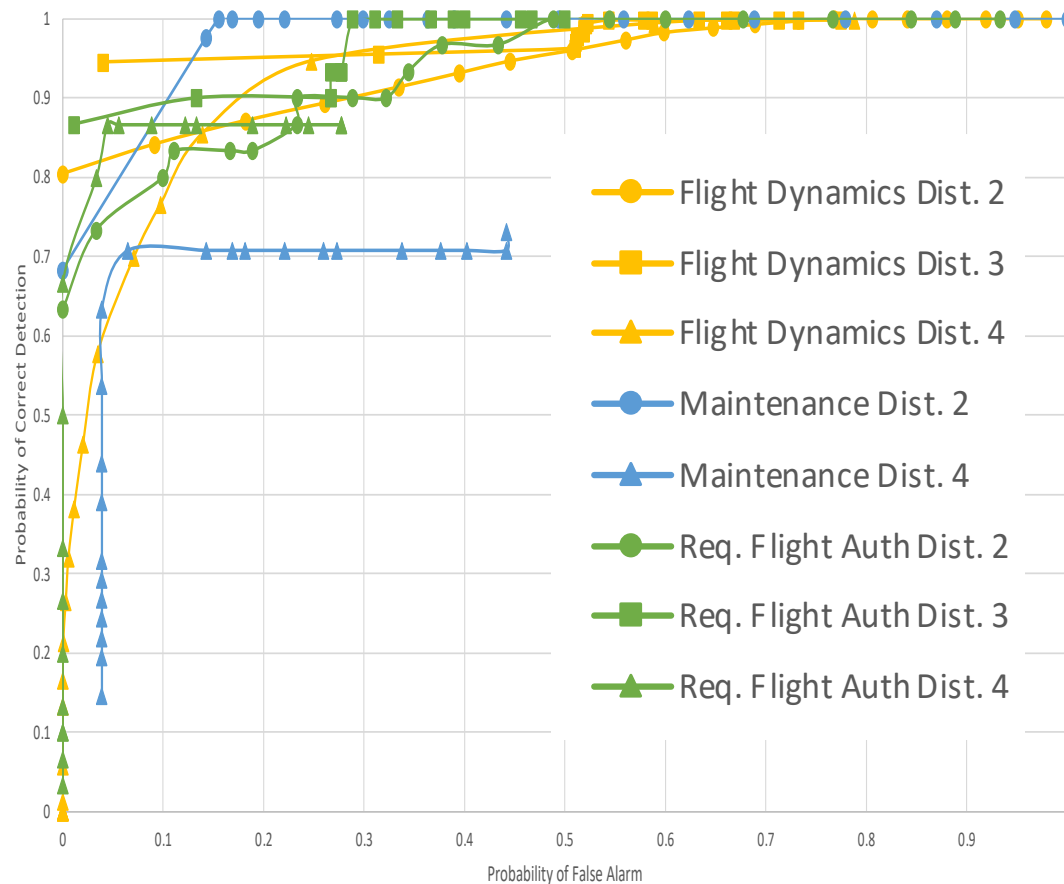
- Common cause/common mode failures need to be considered carefully in novel HMT paradigms with eVTOL
- Know what the strengths, weaknesses, and assumptions are in any model and modelling language
  - What is the level of abstraction, assumptions, and resolution/granularity of the model
  - Be wary of model-based systems engineering that acts as a single source of truth, as this becomes a source of common cause failures
  - Be wary of the conventions of the modelling language, as there may be assumptions/caveats built in (e.g., expressiveness, compatibility, etc.)

# Monitoring (I)

- Agents can control when they perform some of their tasks, but other actions are time-critical, and need to happen at specific times
  - Poor coordination and communication affects safety.
- CRM practice has evolved to overtly require monitoring that might be delayed or omitted under time pressure.
  - Similar concept might be needed in new operations



# Monitoring (II)



- No monitoring across a human-machine team can be perfect
  - The agents have different knowledge
  - The agents have different information
  - The monitoring occurs at a different time
- Effectiveness of monitoring varies based on information distribution and on the criterion for measuring monitoring accuracy
  - e.g., comparing against an operational objective vs against an idealized process model

# How do we move forward?

# Final Thoughts

- Role of human in HMT is substantially different, and many assumptions must be re-examined
- Implications of current policy and regulation on the feasibility of such operations must be properly understood
- Ability to apply performance- and risk-based criteria to system approvals at the operational level (e.g., multi-aircraft systems, remote piloted system, etc.) could provide a more holistic approach
- Resilient humans are currently relied upon to deal with unforeseen, unplanned, and unanticipated hazards and circumstances
  - Pilot must either have ability and situational awareness to intervene OR
  - Alternate means must be sought to reproduce this resilience
- Risk tolerant deployment of novel HMT paradigms may be way forward

# Future Directions

- Depending on whether you're looking for the best human-machine teaming paradigm or for a fully autonomous solution, your (research) needs will vary
- The use of artificial intelligence and/or machine learning in these novel systems will pose additional difficulties
- Need to characterize the human contribution to safety to enable novel human-machine paradigms
- Need a systematic and repeatable way to explore, characterize, allocate, and assess potential functional allocations in human-machine teaming paradigms
- Need interdisciplinary teams of experts across increasingly autonomous systems, human behavior, data sciences and domains in order to design, develop, deploy, assess, and perform research on systems with novel human-machine teaming paradigms

# Questions?

[natasha.a.neogi@nasa.gov](mailto:natasha.a.neogi@nasa.gov)

# Backup slides

# Explainability

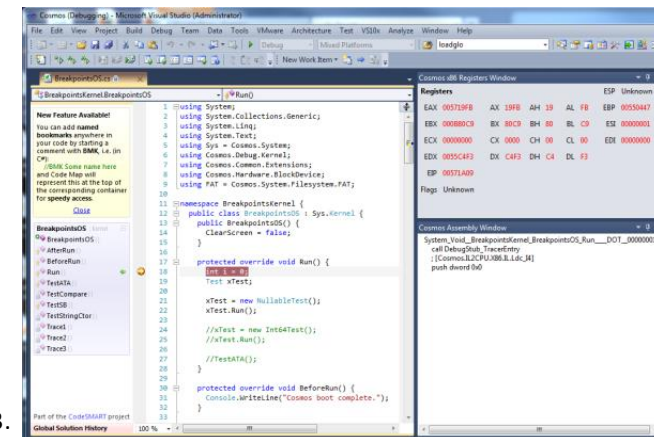
- Explainability means different things to different people [3]:

1. Forensic Explainability: “Black Box” capture of the ‘state’ and ‘environmental conditions’ of the machine agent to provide engineers with enough visibility to determine why, given certain inputs, the machine agent behaved the way it did
2. Explainability for Debugging: Internal operation as well as details about input data given to developers during testing to understand performance failures
3. User Explainability: Explain to users what is going on, using language they can understand

May help to build trust but does not help establish trustworthiness of the machine agent.

3. Goodloe, A., “Assuring Safety- Critical Machine Learning-Enabled Systems: Challenges and Promise”, IEEE Software Engineering, August 2023.

[Boeing 747 interior \(42310015010\)](#)" by [Thom C](#) is licensed under [CC BY 2.0](#). Creative Commons, used with permission of Thom C.



[File:Cosmos debugger in action.png](#)" by [Chad Hower](#), [CCO 1.0](#). Creative Commons, Used with Permission of Chad Hower.

# What is a hazard, anyway?

- There are competing definitions of “hazard”
  - Aviation uses “hazard” haphazardly, but may mean *intrinsic hazard*: the capacity to cause harm (but perhaps not in this design)
    - High voltage is a hazard (although it might be well contained by design)
  - In systems safety, a *hazardous state* (hazard) is a system state that, under worst-case environmental conditions, will lead to a loss
    - A device with an exposed part carrying high voltage is in a hazardous state
  - Extrinsic hazards also exist
- There is the related aviation concept of *failure conditions*
  - Not quite a hazardous state, but close ...
- There is the unrelated aviation concept of “hazardous” severity

# Aviation's poor definition of "hazard"

- In ARP4761A, a "hazard is":  
"A condition resulting from failures, external events, errors, or a combination thereof where safety is potentially affected."

*This definition derives from substantially similar definitions in, e.g., AC 23.1309-1E.*

- This definition is flawed:
  - The condition of improving weather potentially affects safety ... in a good way
  - The external event of Martians attacking with ray guns would threaten safety ... but we don't worry about this in design
- Engineers should focus on states the thing they're designing shouldn't be in

Human machine teaming requires juxtaposing the most complex algorithmic models in existence with the most complex natural systems in existence, and accounting for how they affect and are affected by each other in contexts that are unbounded, high consequence, and defined by uncertainty.

-Dr. Jon Holbrook