NEXT STEPS FOR TRUSTWORTHY MACHINE LEARNING

DARREN COFER
HCSS
8 MAY 2023

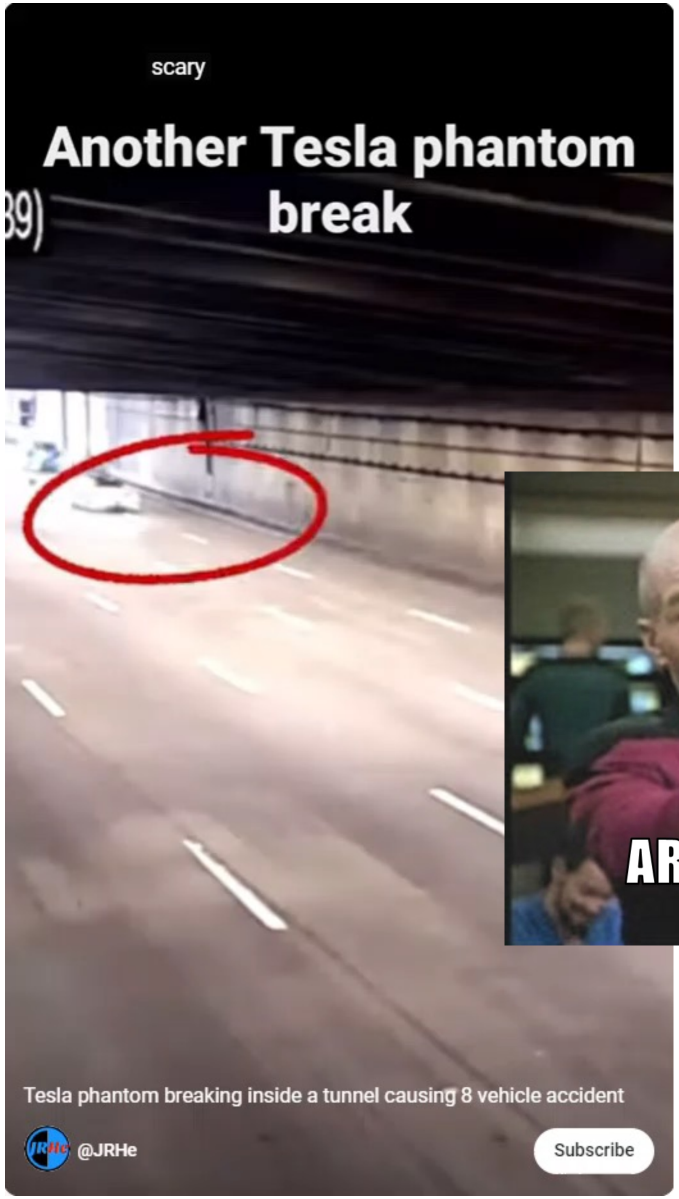
Collins Aerospace | APPLIED RESEARCH & TECHNOLOGY

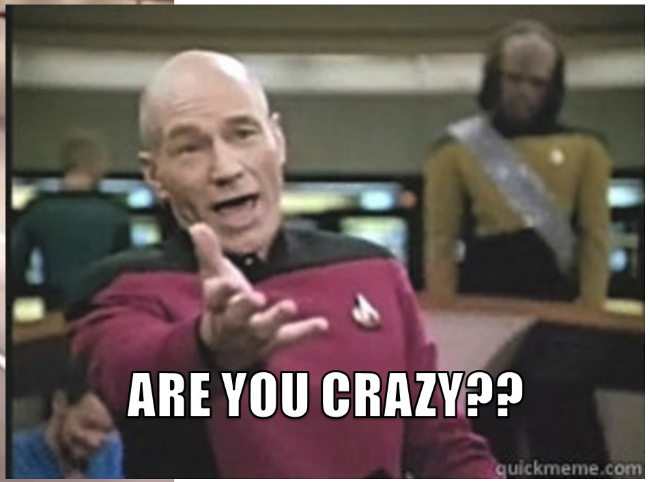
SO YOU WANT TO PUT A NEURAL NETWORK ON AN AIRPLANE…

OR ANY SAFETY-CRITICAL PLATFORM

Tesla car was on Autopilot when it hit a Culver City firetruck, NTSB finds

https://www.latimes.com/business/story/2019-09-03/tesla-was-on-autopilot-when-it-hit-culver-city-fire-truck-ntsb-finds





https://www.youtube.com/shorts/WVh5bxLBX58

TRUSTWORTHY MACHINE LEARNING

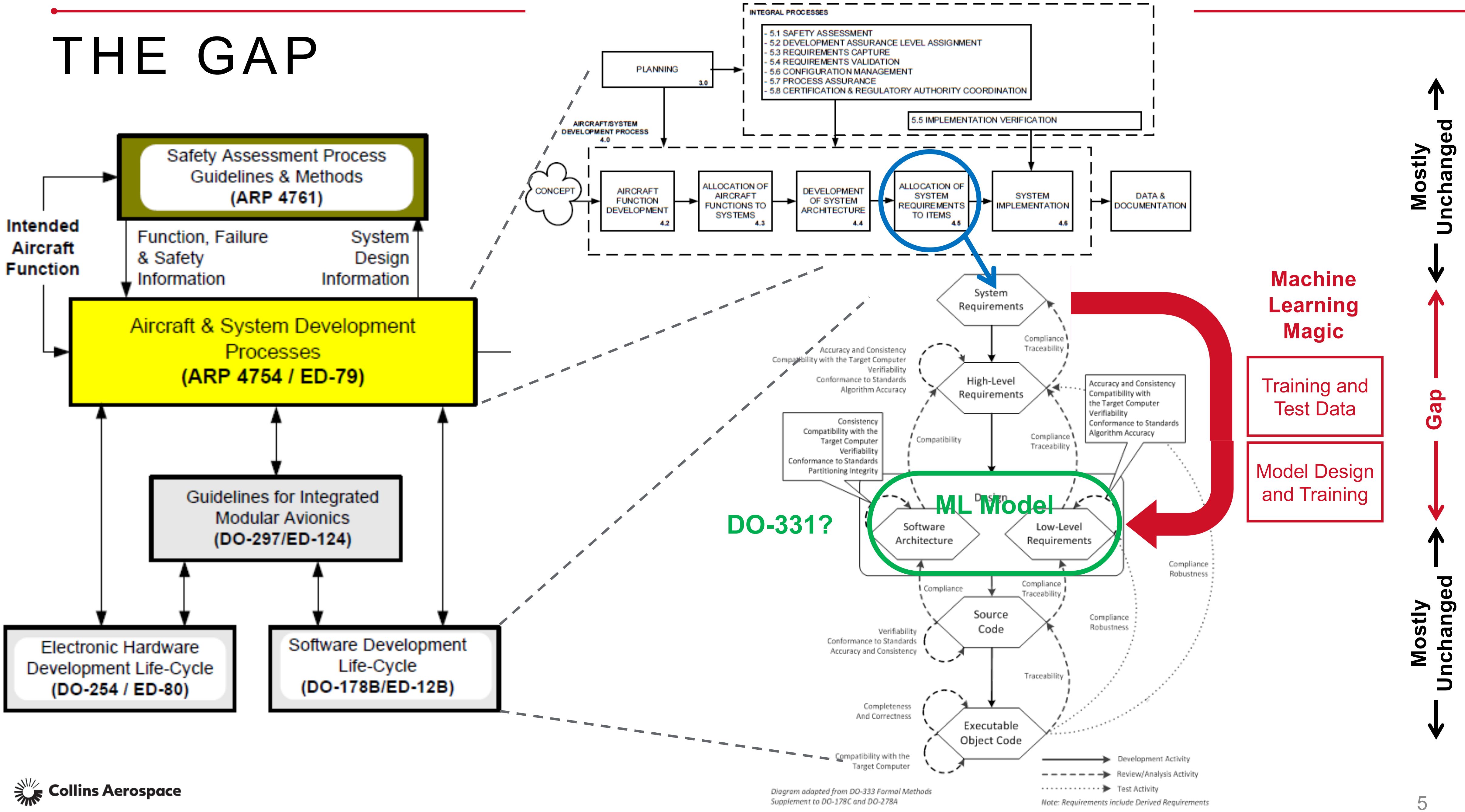
THE WAY FORWARD

- Past – gaps and barriers
- Present – mitigations and standards
- Future – roadmaps and next steps

PAST

GAPS AND BARRIERS

THE GAP

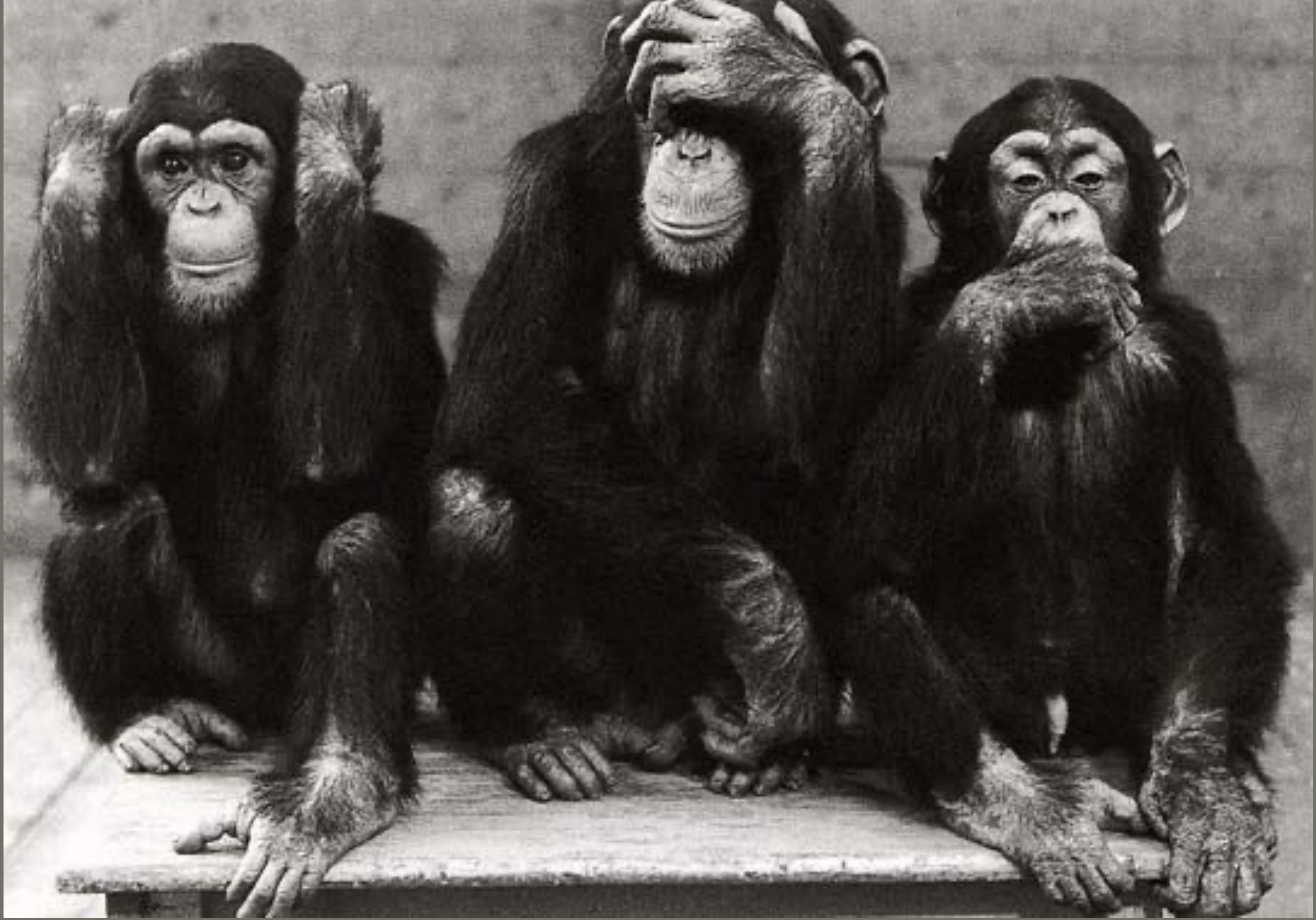


Collins Aerospace

ASSURANCE CHALLENGES FOR ML

- Are requirements complete?
  - Do we have enough training and test data?
  - How to assess completeness and representativeness of datasets?
- Structural coverage metrics for testing don’t work
  - Too easy to 100% coverage for a neural network
  - Can’t detect unintended behavior
  - Can’t detect missing requirements / insufficient data
- Traceability objectives are irrelevant
  - Neither ML model elements (e.g., layers, neurons, weights) nor individual lines of code represent design choices that can be traced back to specific requirements

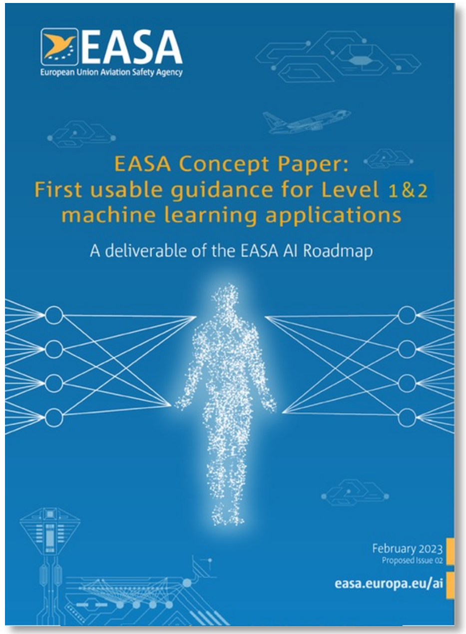
How do we show that no unintended behavior has been introduced during the training process that produces an ML Inference Model?

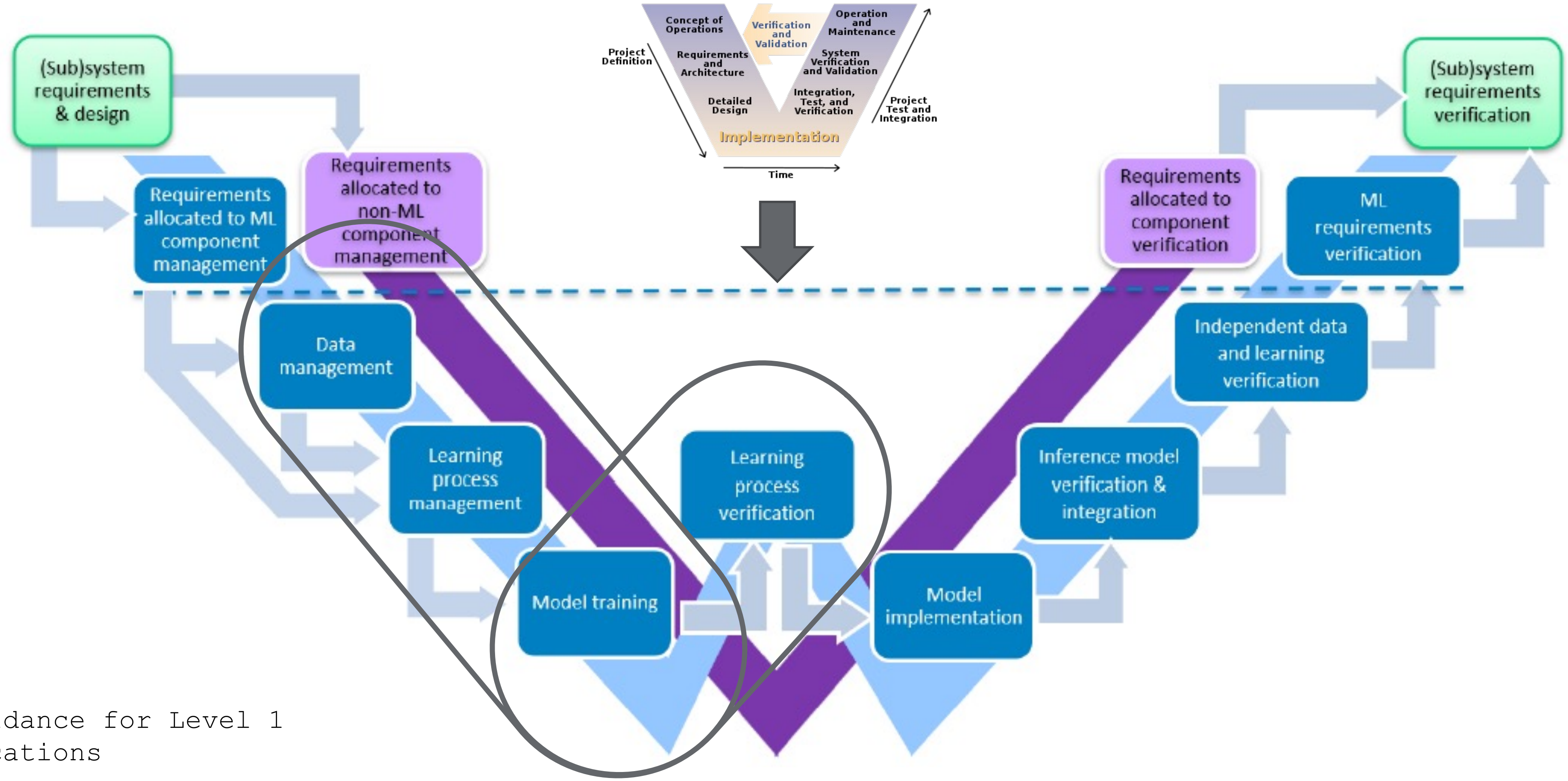


PRESENT

MITIGATIONS AND STANDARDS

LEARNING ASSURANCE PROCESS





Source: EASA First Usable Guidance for Level 1 and 2 Machine Learning Applications (Feb 2023, for comment)
https://www.easa.europa.eu/domains/research-innovation/ai

AS6983 PROCESS OUTLINE

SAE G34 / EUROCAE WG114

1. Learning Process

Use subsystem requirements to define Operational Design Domain (ODD) and training/test datasets

- Data generation/management
- Data is complete and representative relative to ODD
- Model training to achieve performance target

2. Verification of Trained Model

Show absence of unintended behavior

- Generalization
- Stability
- Robustness

3. Inference Model Implementation

Implement model functionality using traditional methods

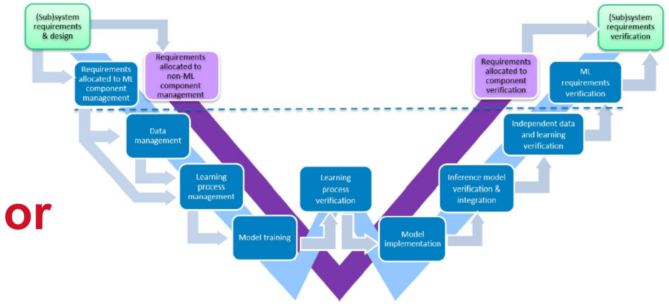
- Verification using traditional methods

4. Inference Model Integration/Verification

Show that implementation of inference model preserves properties of trained model

- Requirements verification
- Performance verification
- Robustness on target hardware
- Compatibility with target hardware

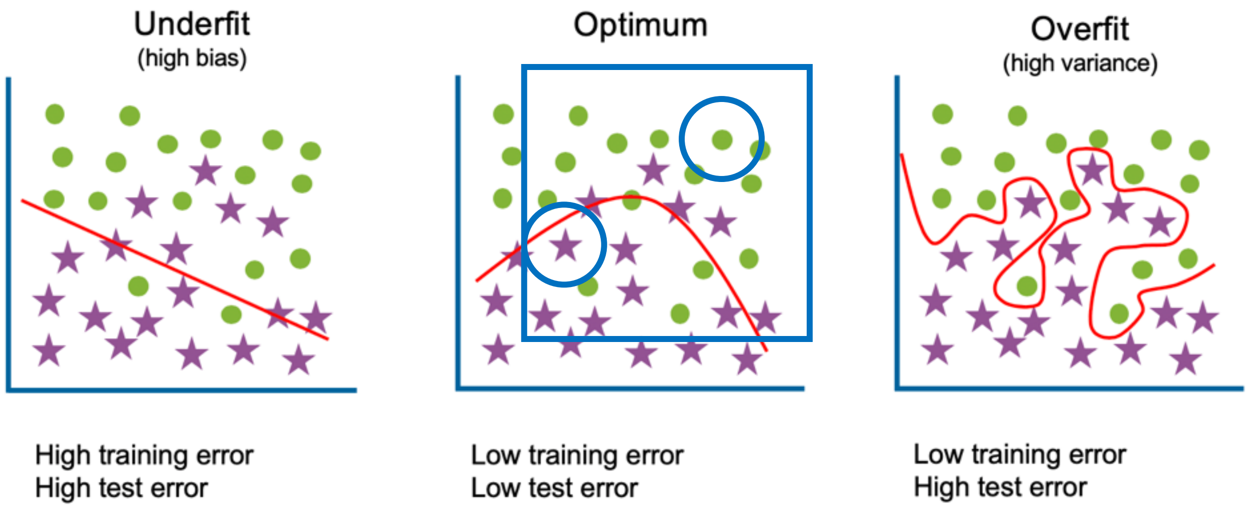
Directly related to unintended behavior



UNINTENDED BEHAVIORS

- Generalization
  - How does system respond to novel/unexpected inputs that were not included in training dataset?
  - In ODD but “between” concrete training points
- Stability
  - How does system respond to perturbations around training data points?
  - Can small input disturbances result in large output deviations?
  - Related to adversarial inputs
- Robustness
  - How does system respond to inputs near boundary of ODD?
  - Similar to abnormal inputs (robustness test cases in DO-178C)

Performance vs. Generalization



https://www.codingninjas.com/codestudio/library/bias-variance-tradeoff

COLLINS–EASA INNOVATION PARTNERSHIP CONTRACT

- **Title:** Formal Methods use for Learning Assurance (ForMuLA)
- April 2023
- https://www.easa.europa.eu/en/downloads/137878/en

GOALS

Influence: Proposed formal methods as anticipated means of compliance for a set of key certification objectives validated by EASA, positioning Collins as a tech leader in the area

Inform: Detailed discussion of FM technologies and applications specific to machine learning

Demonstrate: Practical application on an industrial use case from Collins MiS (remaining useful life estimation)

EASA European Union Aviation Safety Agency

Collins Aerospace – EASA
ForMuLA IPC extract

Collins Aerospace

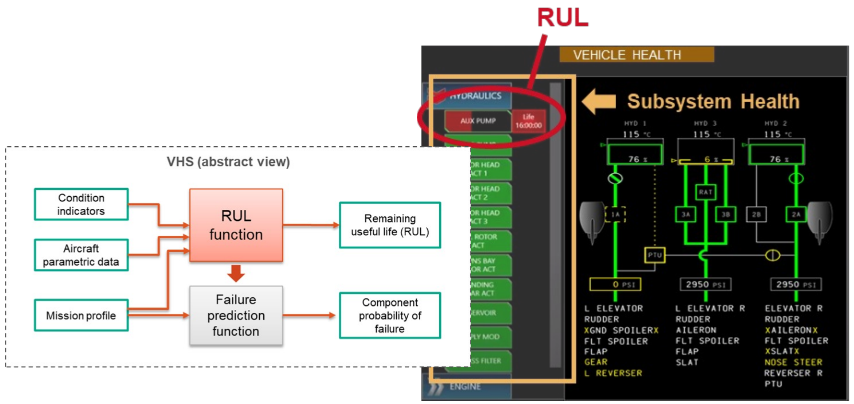
Table of Contents

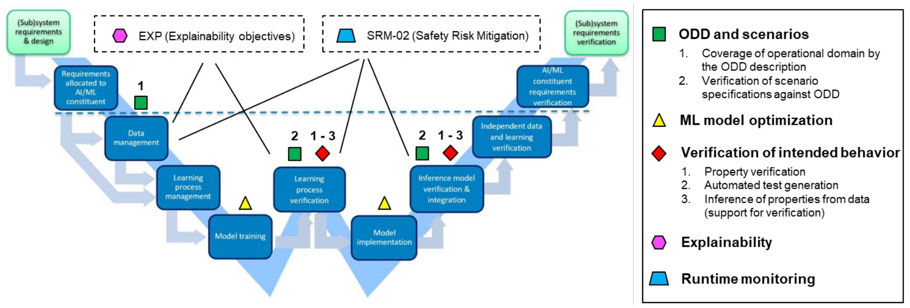
Acronyms 5
Glossary 6
List of figures 8
List of tables 9
Executive summary 10
1 Introduction 11
1.1 Background 11
1.2 Scope of the ForMuLA project 15
1.3 Outline of the report 15
2 Concept of Operations and use case 17
2.1 ConOps and use case selection 17
2.2 Definition of the ML-based system 20
3 Formal methods technologies for machine learning 30
3.1 What are formal methods? 30
3.2 Formal methods main definitions 31
3.3 High-level application categories of formal methods 34
3.4 Property specifications for machine learning 34
3.5 Formal methods technologies applied to machine learning 39
3.6 Scalability limitations of formal methods 45
3.7 Statistical methods 46
3.8 Hybrid verification procedures 48
4 Applications of formal methods specific to ML 49
4.1 Formal methods for supporting the learning process 49
4.2 Formal methods for improving ML model robustness 52
4.3 Other formal methods applications for machine learning 55
5 Assessment of the use of formal methods on the selected use case 60
5.1 Selection of FM applications to be demonstrated 60
5.2 Assessment framework 61
5.3 Data quality verification 66
5.4 Formal verification of the trained ML model 69
5.5 Results of applying formal methods on the RUL use case 89
5.6 Scalability and applicability assessment of formal methods 91
6 Main conclusions of the project 96
References 99

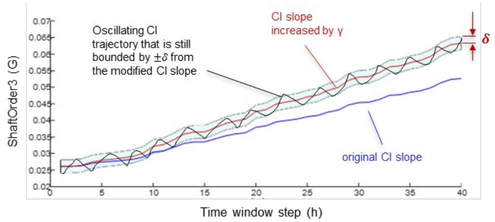
Collins Aerospace

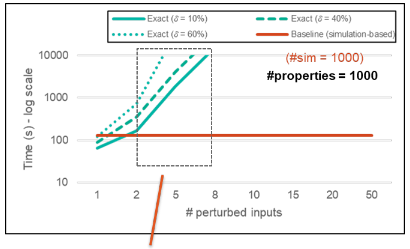
FORMULA – BRIEF OUTLINE

- Use case definition: Remaining Useful Life
  - Also included in VNN-COMP
- FM for ML – State-of-the-Art review
- FM applications for ML development and V&V
  - Mapped to relevant assurance objectives from EASA
- Practical demonstration of FM on the use case
  - Data quality verification (with statistical methods)
  - Property verification: stability, robustness, monotonicity
  - Scalability assessment

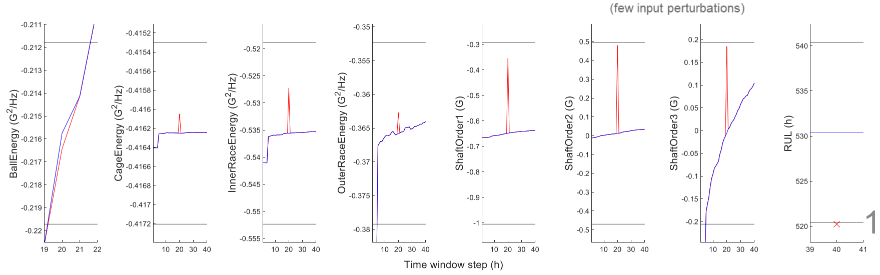






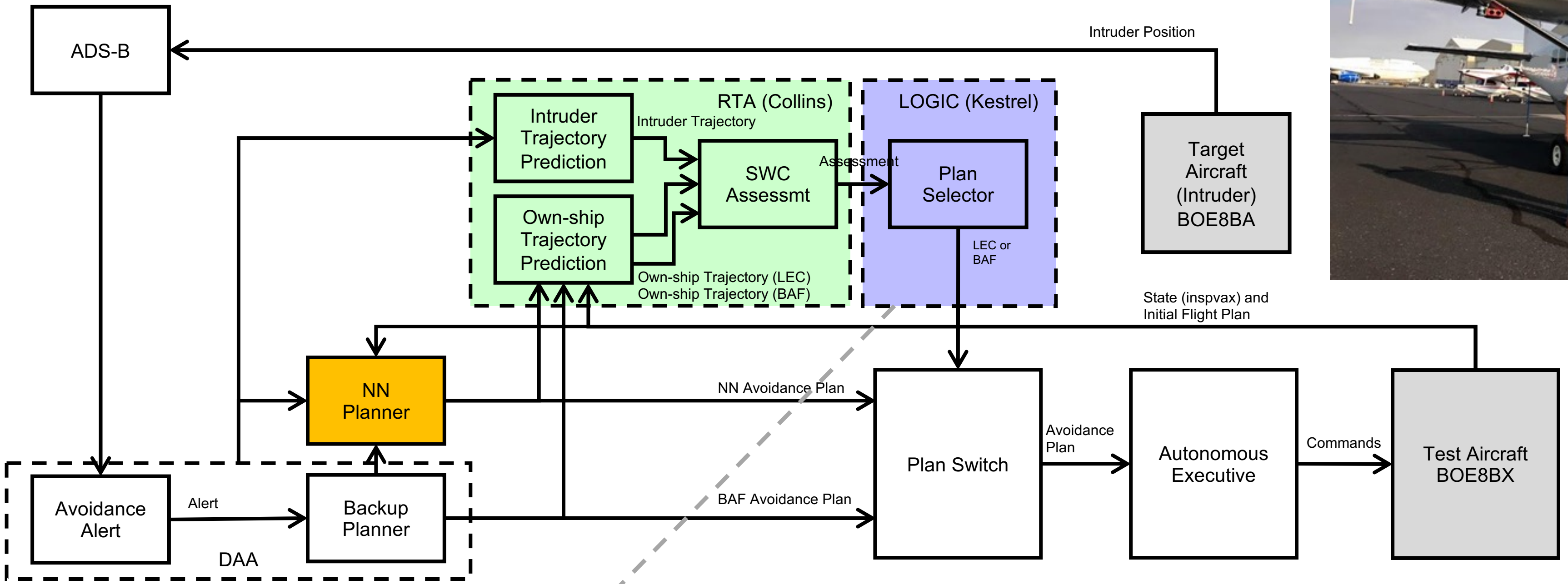


Exact method does not scale beyond simple properties (few input perturbations)



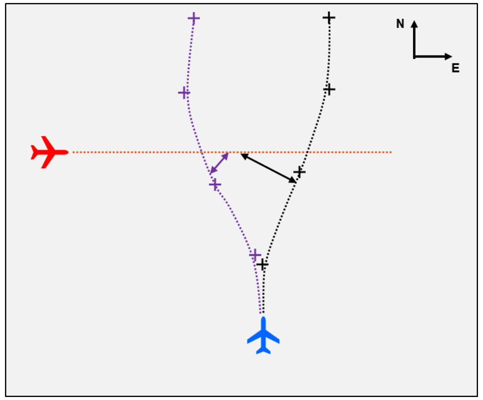
Collins Aerospace

COLLISION AVOIDANCE DEMONSTRATION



inhibiting	LEC valid	BAF valid	LEC safe	BAF safe	LEC tCPA > 179	BAF tCPA > 179	LEC pmd < BAF pmd	currtime > LEC.time + 3	currtime > BAF.time + 3	output
1	-	-	-	-	-	-	-	-	-	NO_PUBLISH
0	0	0	-	-	-	-	-	-	-	NO_PUBLISH
0	1	0	-	-	-	-	-	0	-	NO_PUBLISH
0	1	0		-	-	-	-	1	-	PUBLISH_LEC
0	0	1	-	-	-	-		-	0	NO_PUBLISH
0	0	1	-	-	-	-	-	-	1	PUBLISH_BAF
0	1	1	0	0	-	-	1	-	-	PUBLISH_BAF
0	1	1	0	0			0	-	-	PUBLISH_LEC
0	1	1	1	0	-	-	-	-	-	PUBLISH_LEC
0	1	1	0	1		-	-	-	-	PUBLISH_BAF
0	1	1	1	1	0	0			-	PUBLISH_LEC
0	1	1	1	1	1	0				PUBLISH_BAF
0	1	1	1	1	0	1	-	-	-	PUBLISH_LEC
0	1	1	1	1	1	1	1	-	-	PUBLISH_BAF
0	1	1	1	1	1	1	0	-	-	PUBLISH_LEC

Only one valid plan received : pick that plan
Time out
Neither is safe : select the least unsafe
Only one plan is safe : select that plan
Both plans are safe : select LEC
UNLESS LEC CPA is beyond assessment horizon
OR both CPAs are beyond horizon : select the least unsafe

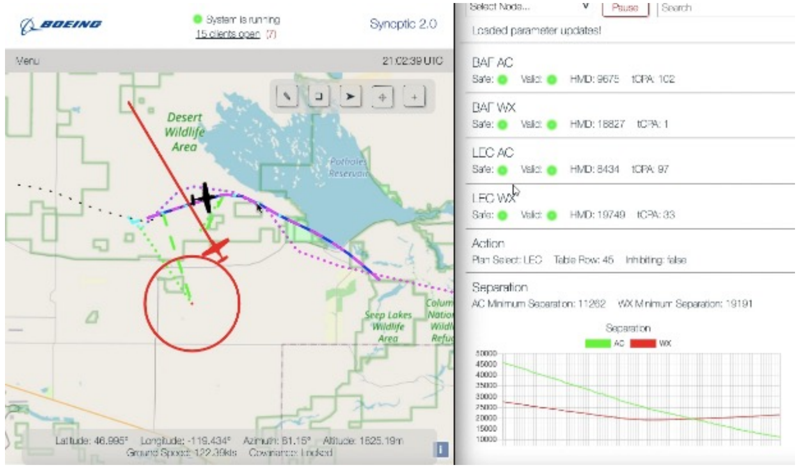


BAF: HMD: DISTANCE CPA: TIME
LEC: HMD: DISTANCE CPA: TIME
PLAN SELECT: BAF / LEC

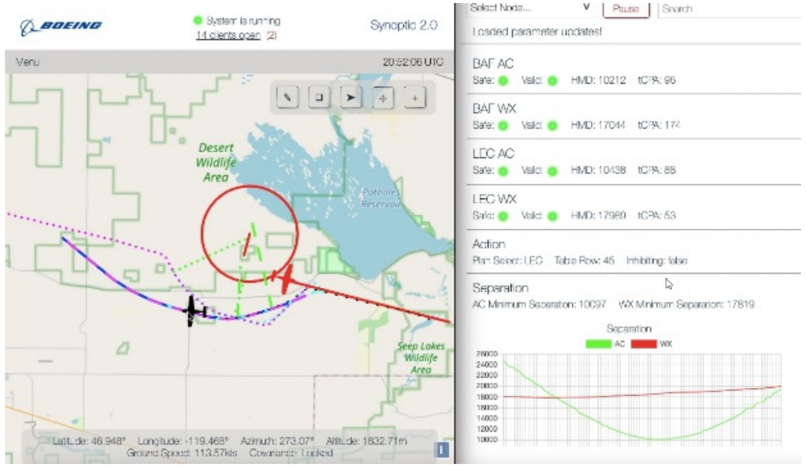
Aahaa!

Collins Aerospace

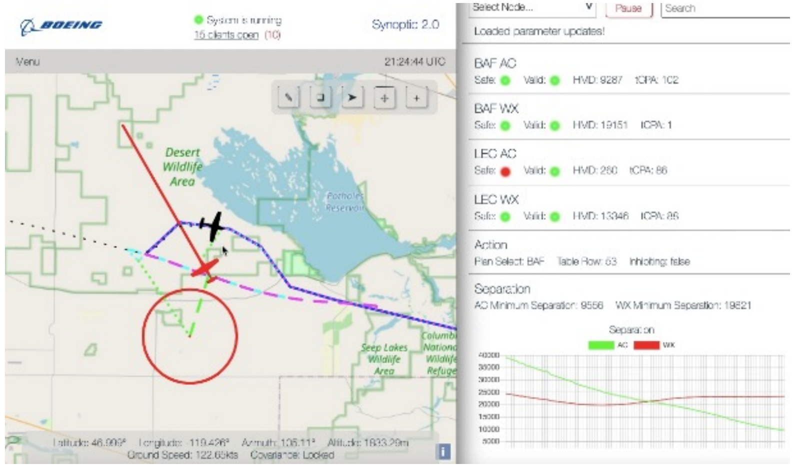
RUN-TIME ASSURANCE FOR MULTI-OBJECT COLLISION AVOIDANCE



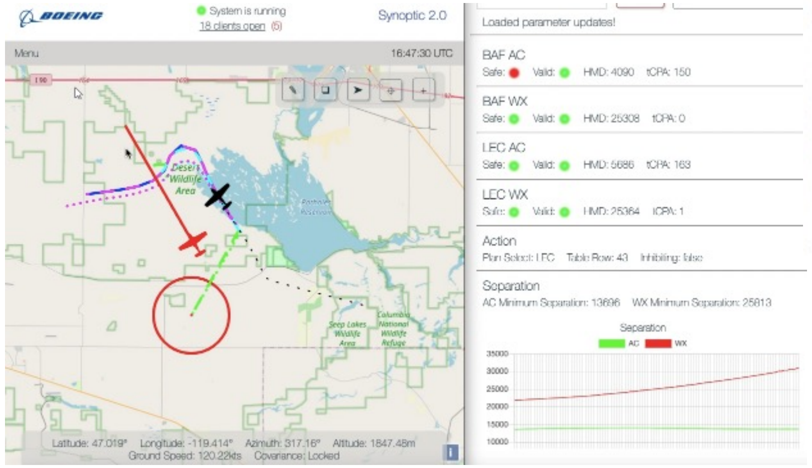
Nominal conditions with various encounter geometries



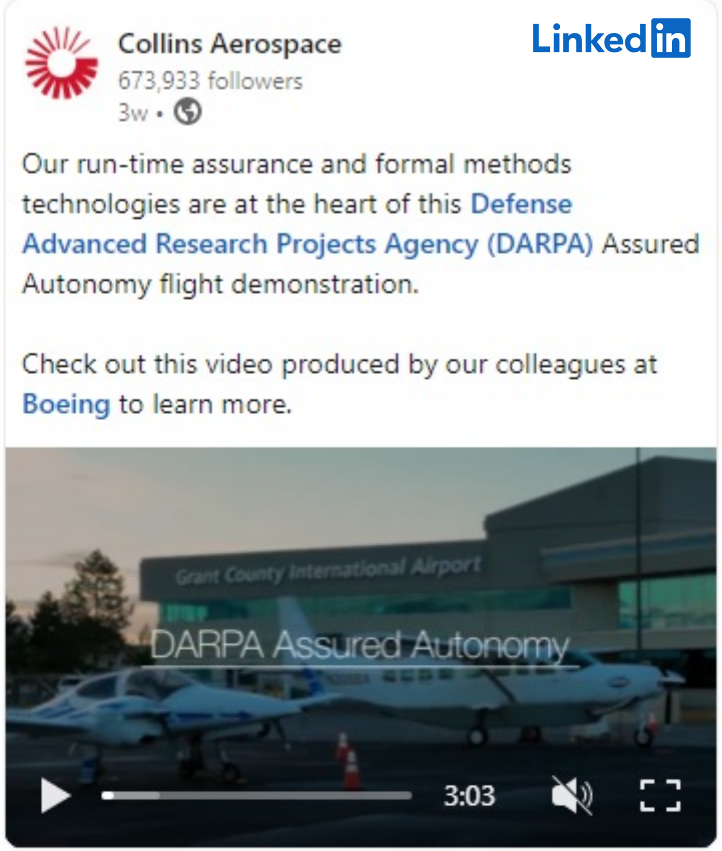
Dynamic (moving) weather cell



Intentionally defective NN



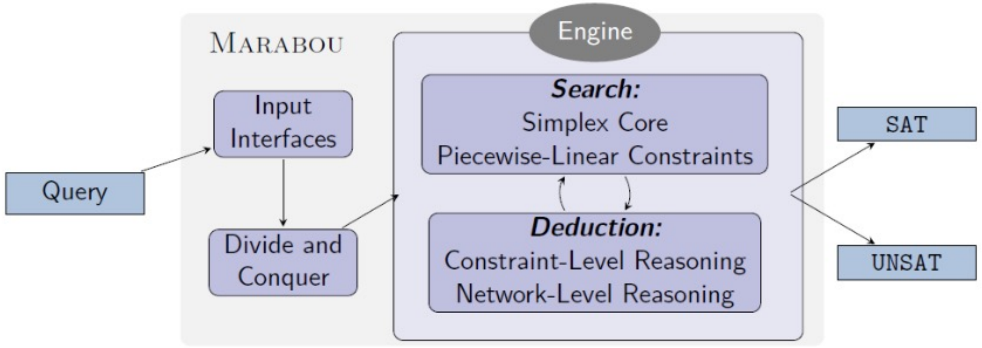
Replan to extend assessment horizon

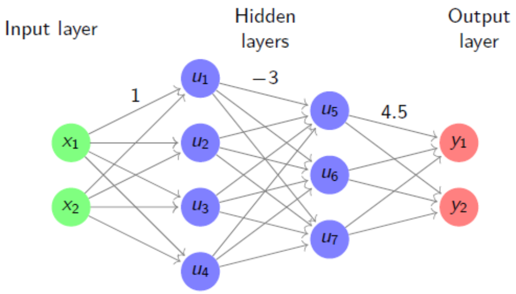


https://www.linkedin.com/posts/collins-aerospace_our-run-time-assurance-and-formal-methods-activity-7043652507977351168-Wldr

MODEL CHECKING BOEING NN (NFM 2023)

- Marabou used to analyze behavior of neural network collision avoidance algorithm
- Property verification and robustness analysis
- How to improve **Reinforcement Learning** results?
- Coverage of input space?





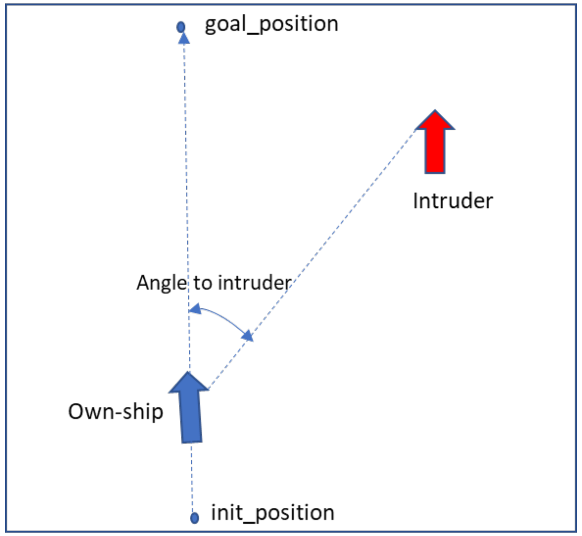
- Reachability properties:

  $x \in [x_l, x_u] \Rightarrow y \in [y_l, y_u]$

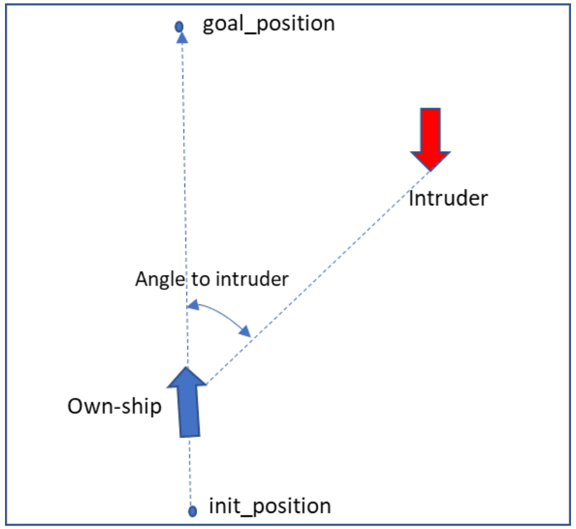
- Robustness properties:

  $\exists x' : |\mathbf{x} - x'| < \epsilon \wedge |\mathbf{y} - y'| > \Delta$

Results align with expectations:
Own-ship always turns away from intruder when they fly in the same direction and are dangerously close (< 2750m)



Unintended behavior:
Own-ship does not turn away from incoming intruder when they are at MIN_DIST



Cong Liu, Darren Cofer, Denis Osipychev, Verifying an Aircraft Collision Avoidance Neural Network with Marabou, NFM 2023

FUTURE

ROADMAP AND NEXT STEPS

AUTONOMY VERIFICATION ROADMAP (NASA)

- Report published January 2023
- Identifies V&V needs related to assurance and certification of technologies supporting autonomous operations, including machine learning
- Short/mid/long-term research needs
- https://ntrs.nasa.gov/citations/20230003734

NASA/TM-20230003734

AUTONOMY VERIFICATION & VALIDATION ROADMAP AND VISION 2045

Guillaume P. Brat
NASA Ames Research Center, Moffett Field, CA, USA

Huafeng Yu
Boeing Research & Technology, Santa Clara, CA, USA

Ella Atkins
University of Michigan, Madison, WI, USA

Prashin Sharma
University of Michigan, Madison, WI, USA

Darren Cofer
Collins Aerospace, Minneapolis-St. Paul, MI, USA

Michael Durling
GE Research, Niskayuna, New York, USA

Baoluo Meng
GE Research, Niskayuna, New York, USA

Christopher Alexander
GE Research, Niskayuna, New York, USA

Szabolcs Borgyos
GE Research, Niskayuna, New York, USA

Chuchu Fan
Massachusetts Institute of Technology, Cambridge, MA, USA

Kunal Garg
Massachusetts Institute of Technology, Cambridge, MA, US

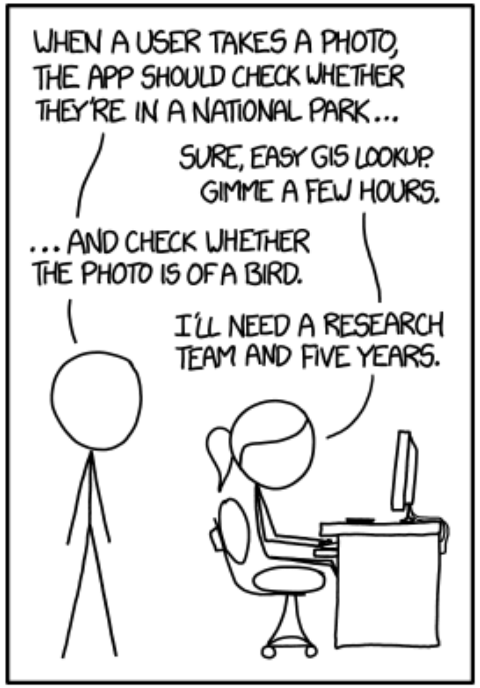
National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

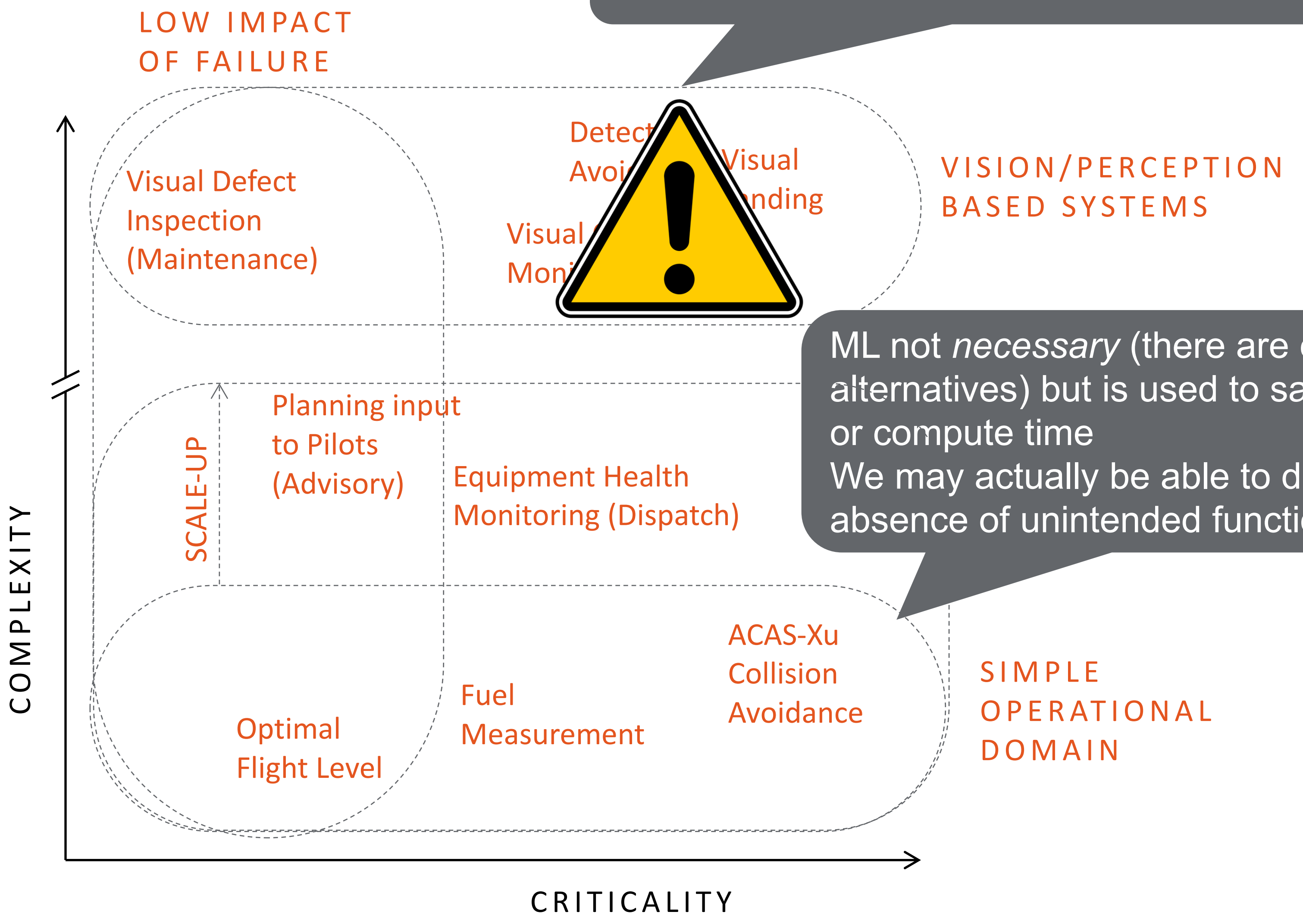
January 31, 2023

AVIATION ML APPLICATIONS

Perception is a strong point for ML/DNN, but also hardest to verify (size, requirements)



https://xkcd.com/1425/

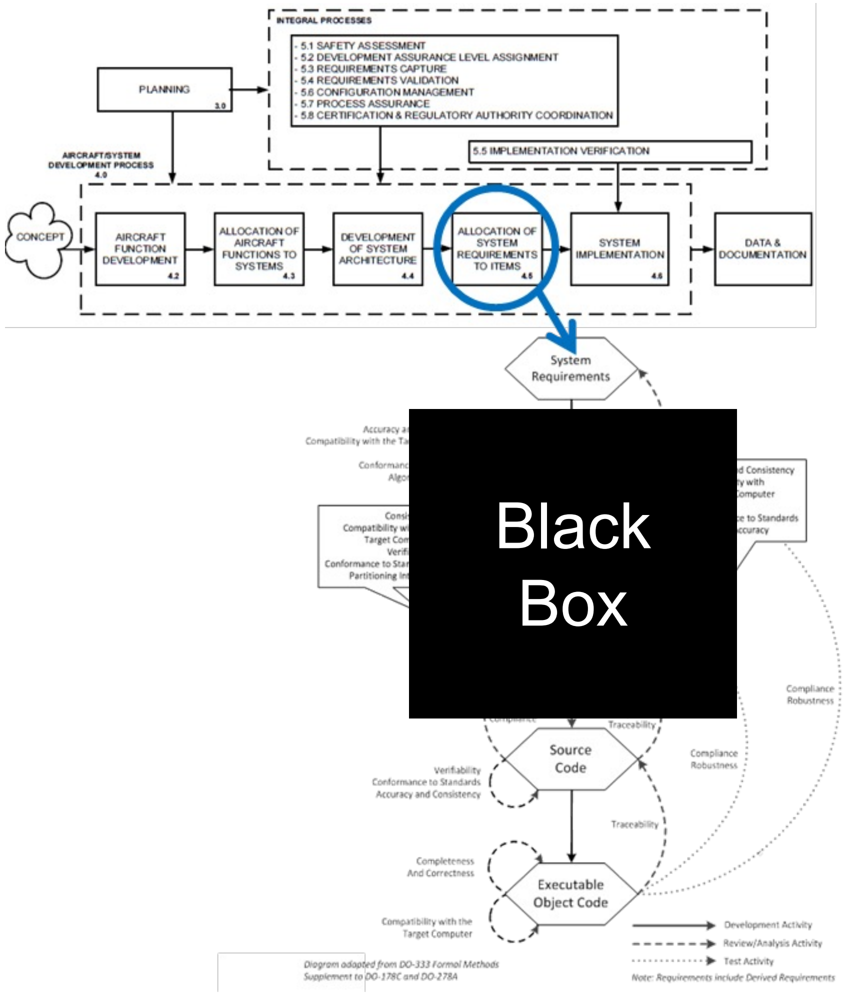


ML not *necessary* (there are existing alternatives) but is used to save memory or compute time
We may actually be able to demonstrate absence of unintended functionality

FIRST STEPS : LOW CRITICALITY

DAL D = MINOR

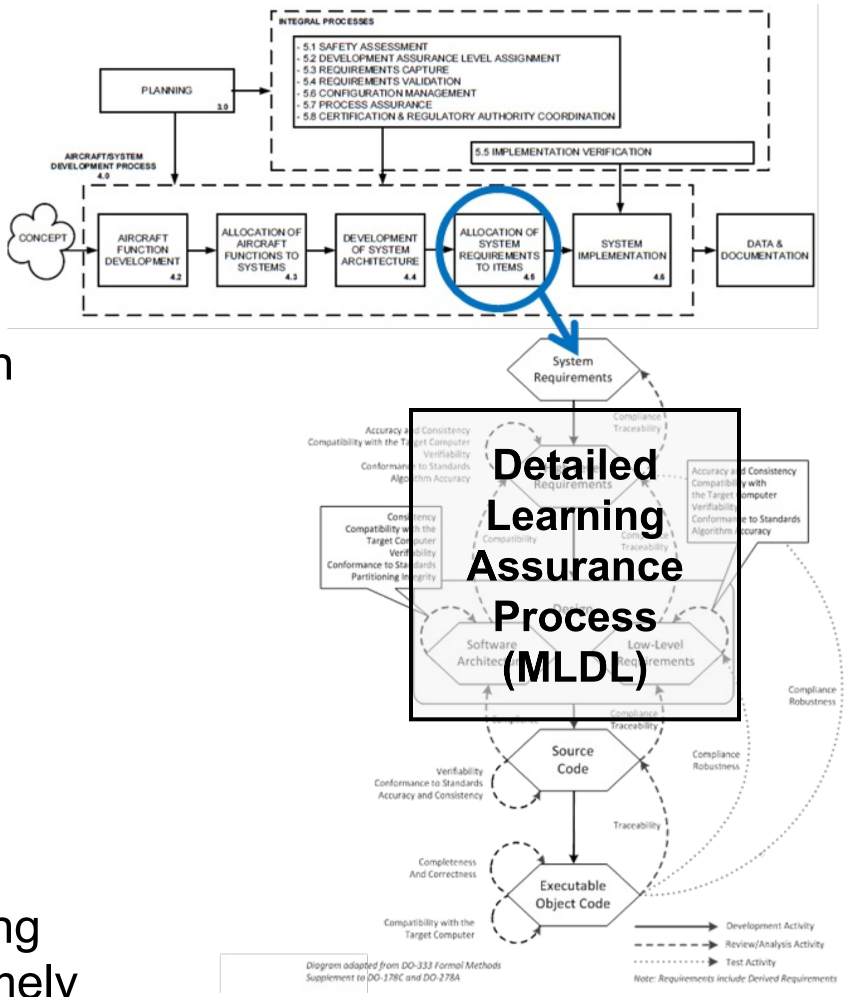
- Software is a “black box”
- Testing to show that software meets its requirements
    - Very little about implementation details
    - Nothing related to unintended function
- What else might be needed?
    - Operational Design Domain (ODD)
    - Training data set
    - Verification data set
    - Basic architecture (e.g., 3-layer feed-forward neural network with tanh activation functions, trained with PyTorch)



FIRST STEPS : LOW COMPLEXITY

SIMPLE NEURAL NETWORK

- Criticality up to DAL A supported
- Limited to NN with “simple” ODD (or “small” model)
    - Small number of well-defined scalar-valued sensor inputs with max/min range
    - I.e., not images
- What else might be needed?
    - Full learning assurance process, similar to AS6983 MLDL or EASA First Usable Guidance (for DAL A-C anyway)
    - Rigorous data management process
    - Demonstrations of generalization, stability, robustness
    - Justification for absence of unintended behavior/function (using formal methods, traditional mathematical reasoning, or extremely dense training/testing data sets)
- *Complex NN will not yet be able to satisfy these objectives*



TOOLBOX FOR TRUSTWORTHY ML

- **Formal Methods** for robustness/generalization (α,β-Crown, NNV, Marabou, Verinet, as well as other best-in-class tools identified in the VNN-COMP)
- **Manifold-based testing** based on computation of the lower-dimensional manifold where real-world input data is concentrated
- **Run time assurance** architecture based on the principles of ASTM F3269-17
- Input **out-of-distribution monitoring** such as Sketching Curvature for Out-of-Distribution Detection (SCOD) method combining online and offline methods
- **Gradient-based analysis** of the function implemented by the NN to determine the upper bound of outputs between the training data points
- NN **property inference and coverage** based on extracting patterns of neuron decisions as preconditions that imply certain desirable output properties
- **Input quantization**: For some low-complexity systems, it is possible to obtain sufficient accuracy by quantizing inputs to the actual training data set, limiting or eliminating generalization issues.
- **Input coverage testing**: For some low-complexity systems, it is possible to obtain sufficient coverage of the input operational design domain (ODD) at high resolution.

SCENARIO-BASED COVERAGE

Scenario scripting language

↓

Scene modeling

↓

Rendering

↓

Labeled synthetic training dataset

Can we move from “pixel space” to “scenario space” to create new metrics for completeness of training datasets?

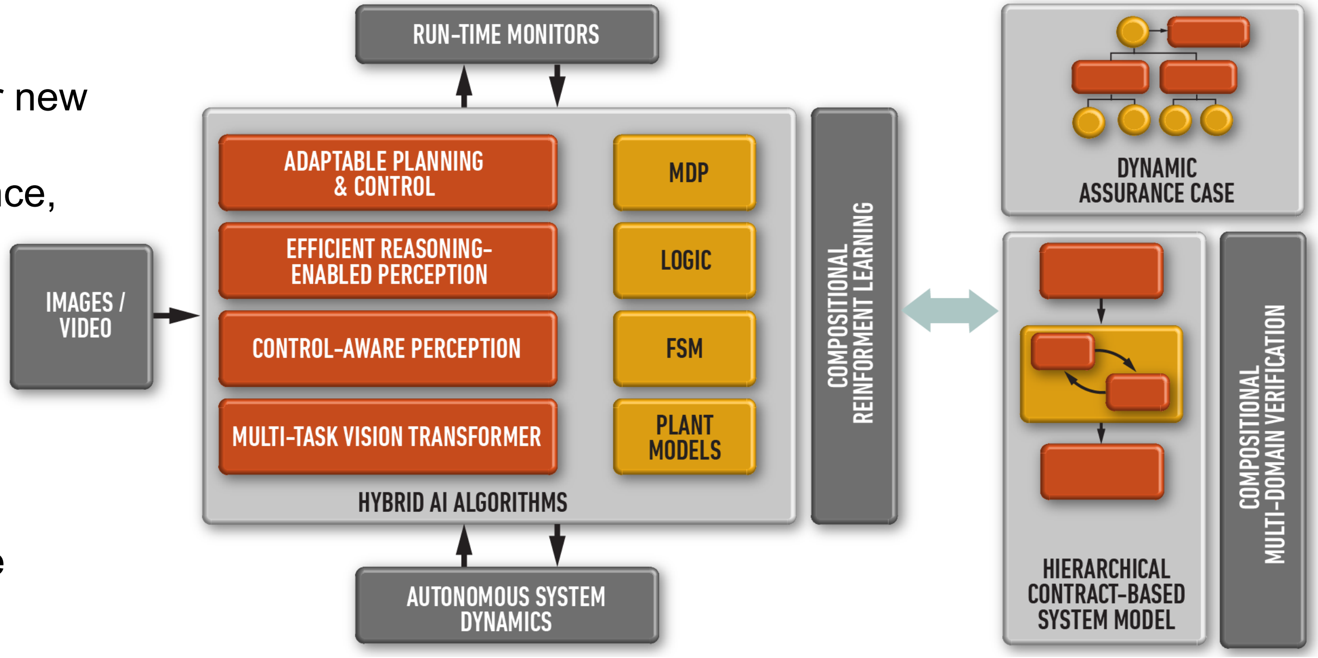
Model	Parameters (million)	FPS	AP test (%)
YOLO7-Tiny	6.2	286	38.7
YOLOv7	36.9	161	51.4
YOLOv7-X	71.3	114	53.1
YOLOv7-W6	70.04	84	54.9
YOLOv7-E6	**97.2**	**56**	**56.0**
YOLOv7-D6	154.7	44	56.6
YOLOv7-E6E	151.7	36	56.8

Current vision ML architectures are HUGE

ASSURED NEURO SYMBOLIC LEARNING AND REASONING (ANSR)

NEURO-SYMBOLIC PERCEPTION, ACTION & REASONING (NEUROSPAR)

- Current approaches to machine learning rely on unsustainable growth in data requirements to achieve performance improvements yet often lack needed assurance
- Hybrid AI approaches that **leverage both data-driven learning and symbolic domain-based reasoning** offer new capabilities to meet the trustworthiness needs of DoD applications such as autonomous intelligence, surveillance, and reconnaissance (ISR)
- Compositional reasoning over multi-domain contracts, design-time and run-time verification and monitoring, dynamic assurance case approach to hybrid AI
- Verifiable and efficient hybrid AI algorithms enabling co-design of perception and control, novel compositional framework for RL with hybrid AI accommodating multiple symbolic representations and accounts for information limitations



SUMMARY

- Machine learning presents many unique assurance challenges in the aviation environment (mostly related to unintended behaviors)
- EASA (European Aviation & Space Administration) has initiated work to address these challenges, including First Usable Guidance concept paper
- The SAE G34 / EUROCAE WG114 joint committee is moving forward to produce industry consensus certification guidance that is intended to address the challenges posed by AI/ML, enabling its use in increasingly autonomous aircraft
- High complexity ML functions (vision) will continue to be a challenge in applications that require the highest levels of assurance
- But we can make progress now on simple / low criticality ML functions