Collaborative Functional Design Using Explainable Machine Learning (X-ML)

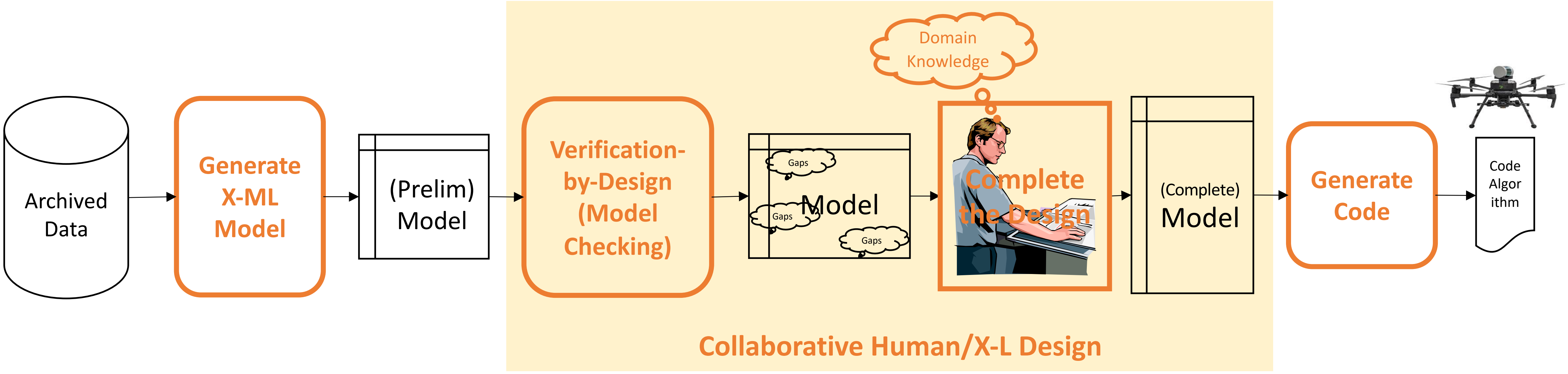
Lance Sherry, Jim Baldo, Brett Berlin, Oleksandra Snisarevska-Donnelly

lsherry@gmu.edu

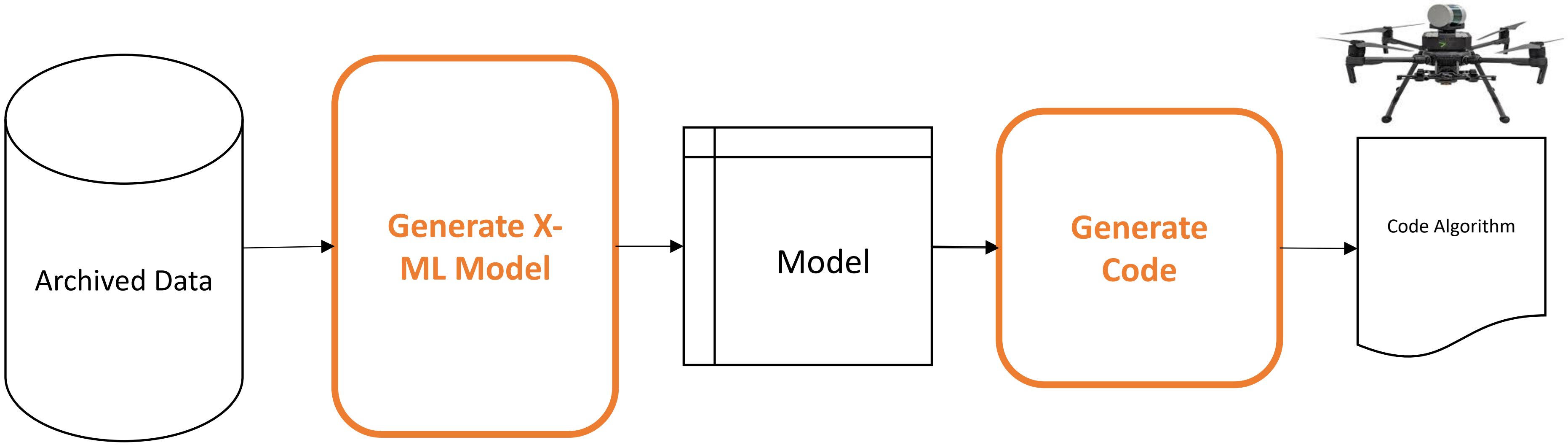
DE 1:35 – 02:05pm Oct 29



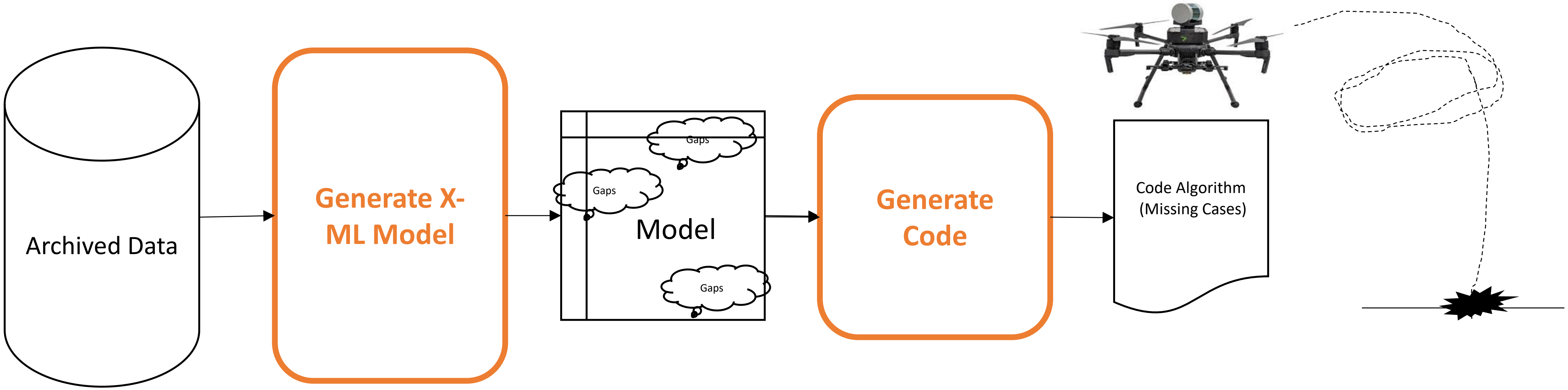

lsherry@gmu.edu



The AI Advertising



The Reality



Archived Data
Generate X-ML Model
Gaps
Gaps
Model
Gaps
Generate Code
Code Algorithm (Missing Cases)

The Solution:

Collaborative Functional Design Using Explainable Machine Learning (X-ML)

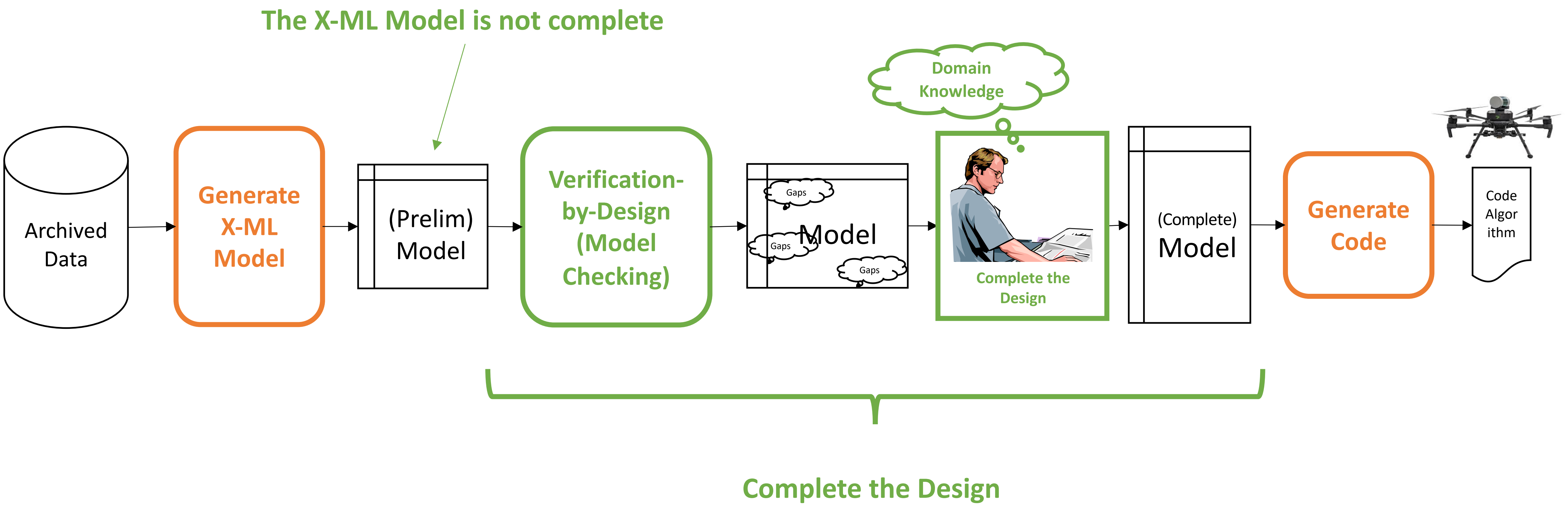


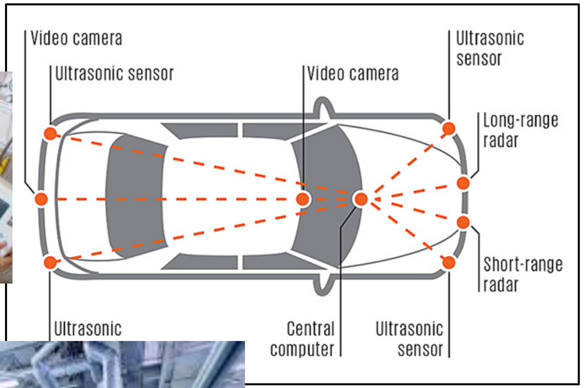
Table of Contents

Operationally Embedded Control Systems

- Embedded on vehicle or plant
- Provide Guidance and Control functions to perform Mission
- Complex
  - Over 200 input signals
  - Over 10 actuator command outputs

Operationally Embedded Control Systems

- Operationally embedded control systems are widely used:
  - military applications
  - vehicle guidance and control
  - robot manipulation
  - mission planning
  - health diagnosis



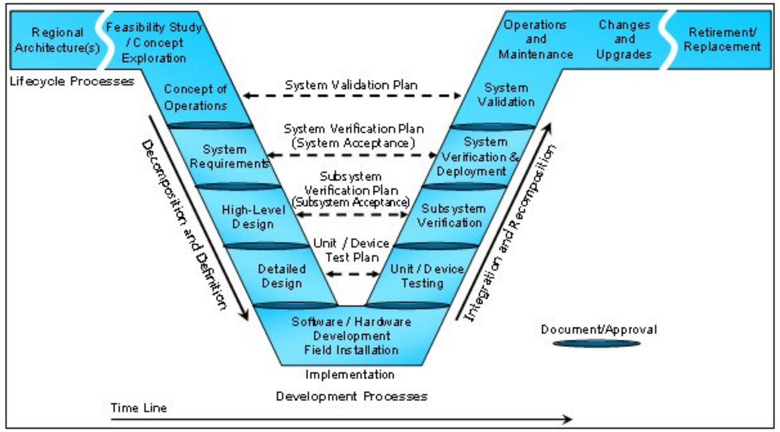
Operationally Embedded Control Systems

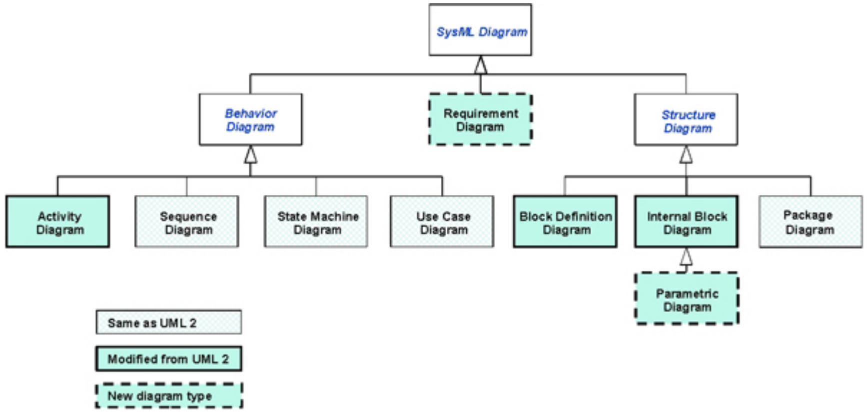
- functional behavior of a typical military operationally embedded control system is complex
  - over 200 input signals
  - 10 outputs actuator commands
- Many applications require operational reliability of at least five-nines for “airworthiness” approval
  - Interpretable:
    - System Description Documents
    - Training Manuals
    - Operator User-interfaces
  - Executable
  - Certifiable (e.g. DO-178)

Table of Contents

Operationally Embedded Control Systems

- Traditional System Engineering practice:
  - specifies the functional requirements for the operationally embedded control systems
  - a manual engineering process
    - engineers collaborate with operators to define what the automation should do at all times.
- Specifications take the form of functional descriptions of behavior
  - map inputs representing the state of the operational environment to outputs (commands to actuators or guidance/alerts on displays)
- Functional behavior requirements can be specified using several modeling language constructs including combinations of:
  - “shall” requirements
  - Action Diagrams
  - Functional Flow Block Diagrams
  - state-charts, logic diagrams
  - control law diagrams
- When the modeling construct is executable it can be used to:
  - automatically generate code
  - provide verification by analysis of the design





Operationally Embedded Control Systems

- Traditional System Engineering practice:
  - specifies the functional requirements for the operationally embedded control systems
  - a manual engineering process
    - engineers collaborate with operators to define what the automation should do at all times.
- Time consuming
  - 1-2 years in design, coding and testing
- Subject to errors
  - Three Design Error Archetypes
    1. Fail to cover situations that can occur in operations
       1. Missing input
       2. Missing combination of input/states
    2. Fail to specify the appropriate behavior to situations that can occur in operations.

Table of Contents

Machine Learning for OECS

- Generate functions with complex behaviors
  1. Process massive amounts of data (input: output pairs)
  2. Supervised Learning
  3. Test
  4. Deploy

- Capture complex behavior with little/no engineering effort
- Potential to significantly reduce development time!

- Can this potential be fulfilled? What are the limitations

Machine Learning for OECS

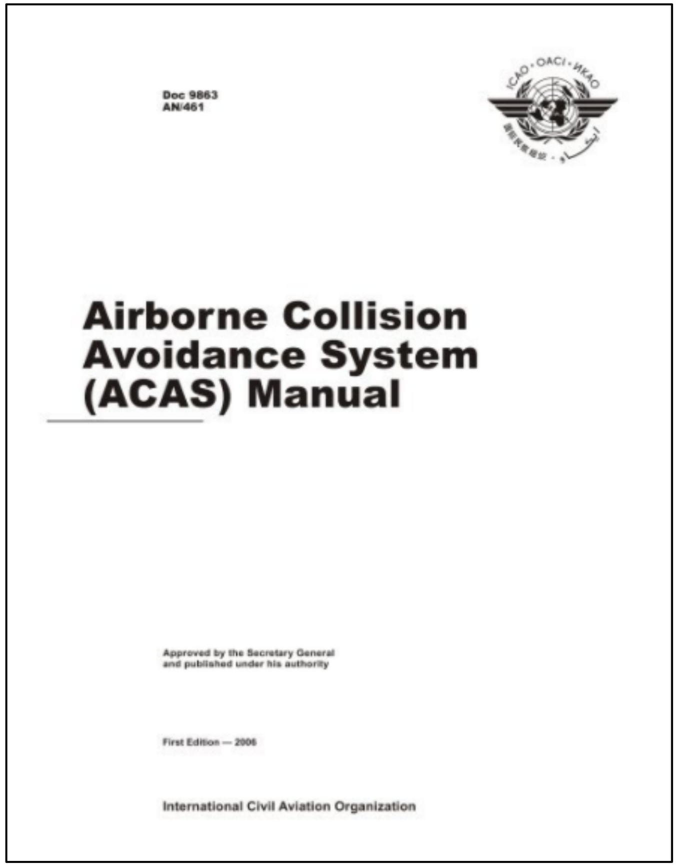
- AI/ML technologies provide **potential** to develop and field functions
  - Increased complex functional behavior
  - Lower Development costs and time
  - Safer (through increased complex functional behavior)
    - Better than humans?

- Can this potential be fulfilled? What are the limitations

Table of Contents

1. Operationally Embedded Control Systems (OECS)
2. Design and Testing for OECS
3. Machine Learning for OECS
4. Case Studies
5. Research Objective
6. Collaborative Functional Design Using X-ML

Case Study: ACAS



Doc 9863
AN/461

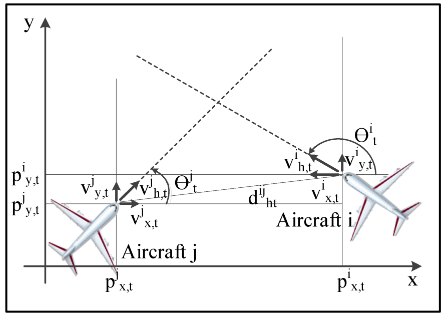
Airborne Collision Avoidance System (ACAS) Manual

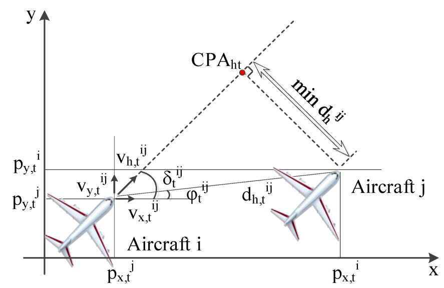
Approved by the Secretary General
and published under his authority

First Edition — 2006

International Civil Aviation Organization

- Decision logic
- Finite “collision geometry design-space”
- Each “collision geometry” has defined response





Case Study: HSCT – VNAV G&CF

- NASA High Speed Civil Transport (HSCT) Flight Management System (FMS)
  - HSCT FMS is based on MD-11 FMS

- Generated FOQA data from simulation flights
  - ~18,000 “Guidance and Control” scenarios
  - Scenarios included rare-events (e.g. Engine-out)
  - Data had to balanced to increase frequency of rare-event situations
    - ~ 26,000 “Guidance and Control” scenarios

Case Study: X-ML Model Accuracy Parameters

```
Precision: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Recall: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Fscore: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Support: [220 167 193 183 203 236 195 187 199 210 206 195 208]
Train Accuracy:  1.0
Test Accuracy:  1.0
```

```
Confusion matrix:
[[220   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0 167   0   0   0   0   0
 [  0   0 193   0   0   0   0
 [  0   0   0 183   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0 203   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0 236   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0 195   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0 187   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0 199   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0 210   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0 206   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0 195   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0 208]]
```

No Type I or II Errors

but …

- VNAV and ACAS were “missing” Behavior

1. Missing Situations
    1. Missing Inputs
    2. Missing Input/State combinations
2. Missing Situation-Behavior Mapping

What are the Gaps in X-ML Designs?

- X-ML Design is Missing Input
  - Design is ***absent one or more of the required inputs*** (i.e. sensors/data feeds) to identify one or more of the operational situations that must be covered by the operationally embedded system
- X-ML is Missing Input/State Combinations
  - Given all the required inputs, the design is ***absent one or more combinations of input states*** to respond to *all* the operational situations that must be covered by the operationally embedded system
- X-ML is Missing Mapping between Input/State Combinations to Behaviors
  - Given the required inputs to support all the combinations of input states and all the combinations of input states, the design is ***absent one or more the correct mappings*** between operational situations and appropriate behaviors

Table of Contents

1. Operationally Embedded Control Systems (OECS)
2. Design and Testing for OECS
3. Machine Learning for OECS
4. Case Studies
5. Research Objective
6. Collaborative Functional Design Using X-ML

Research Objectives

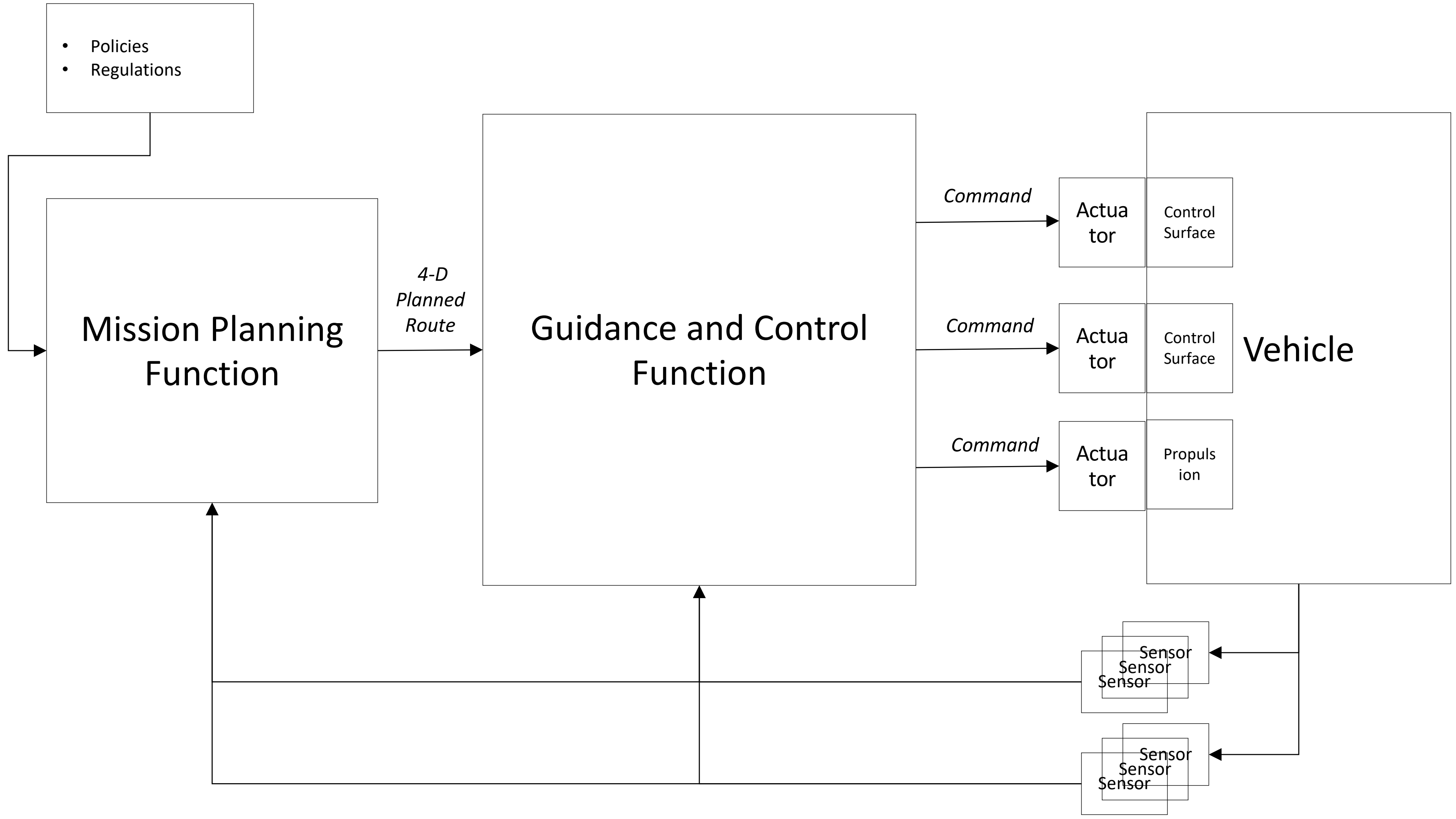
- Demonstrate development of DO-178 certifiable Operationally Embedded Control System
  - Explainable Machine Learning
  - Model must be:
    - Interpretable
    - Executable
    - Compatible with existing airworthiness certification standards

Table of Contents

X-ML OECS Method

- Exploit “natural” architecture of Guidance and Control Systems to generate X-ML algorithms
  - X-ML algorithms can be:
    - converted to functional behavior models
    - mapped to situation/intent/behavior modes

Example OECS: Vehicle Guidance and Control Function

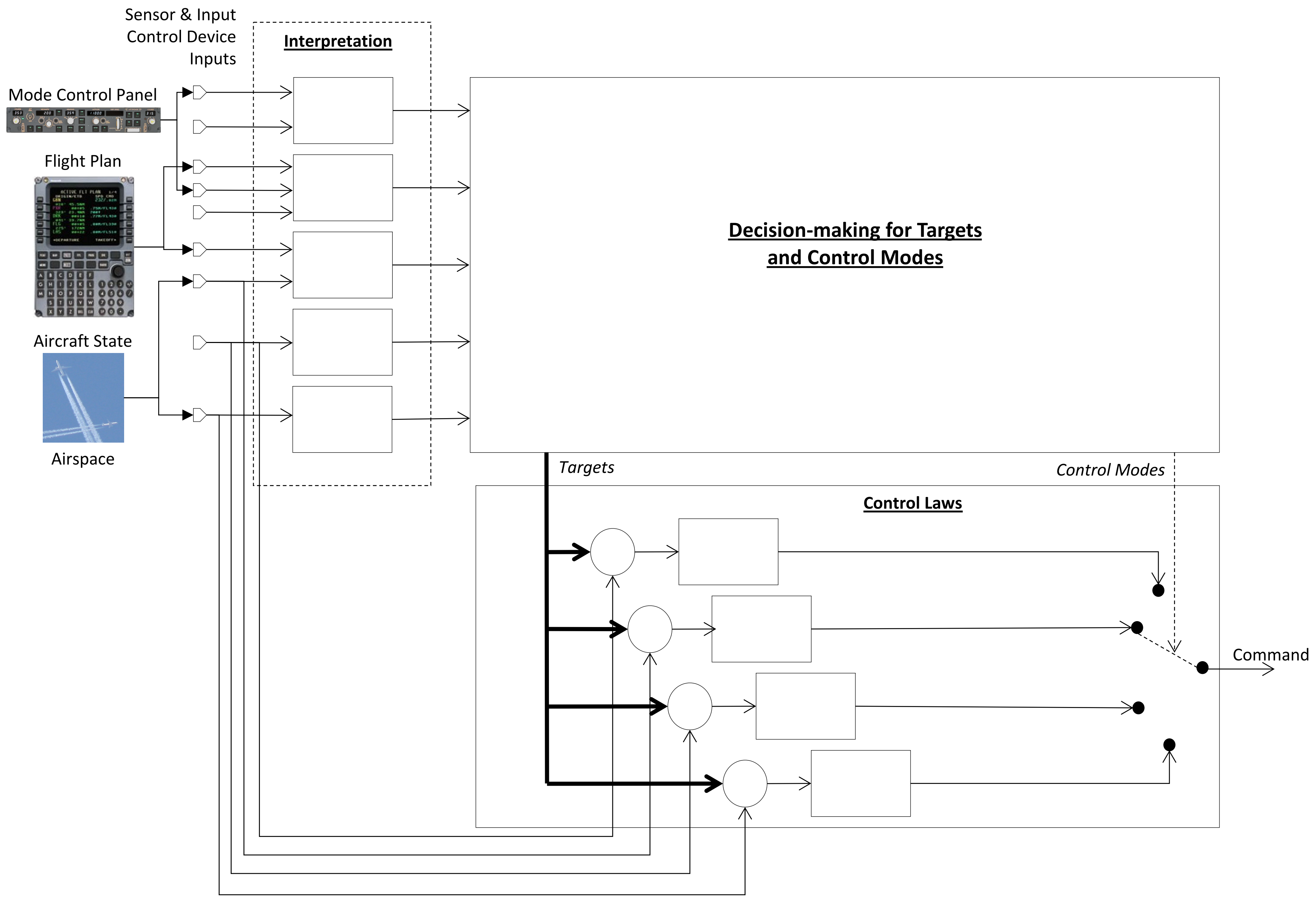


Example OECS: Vehicle Guidance and Control Function

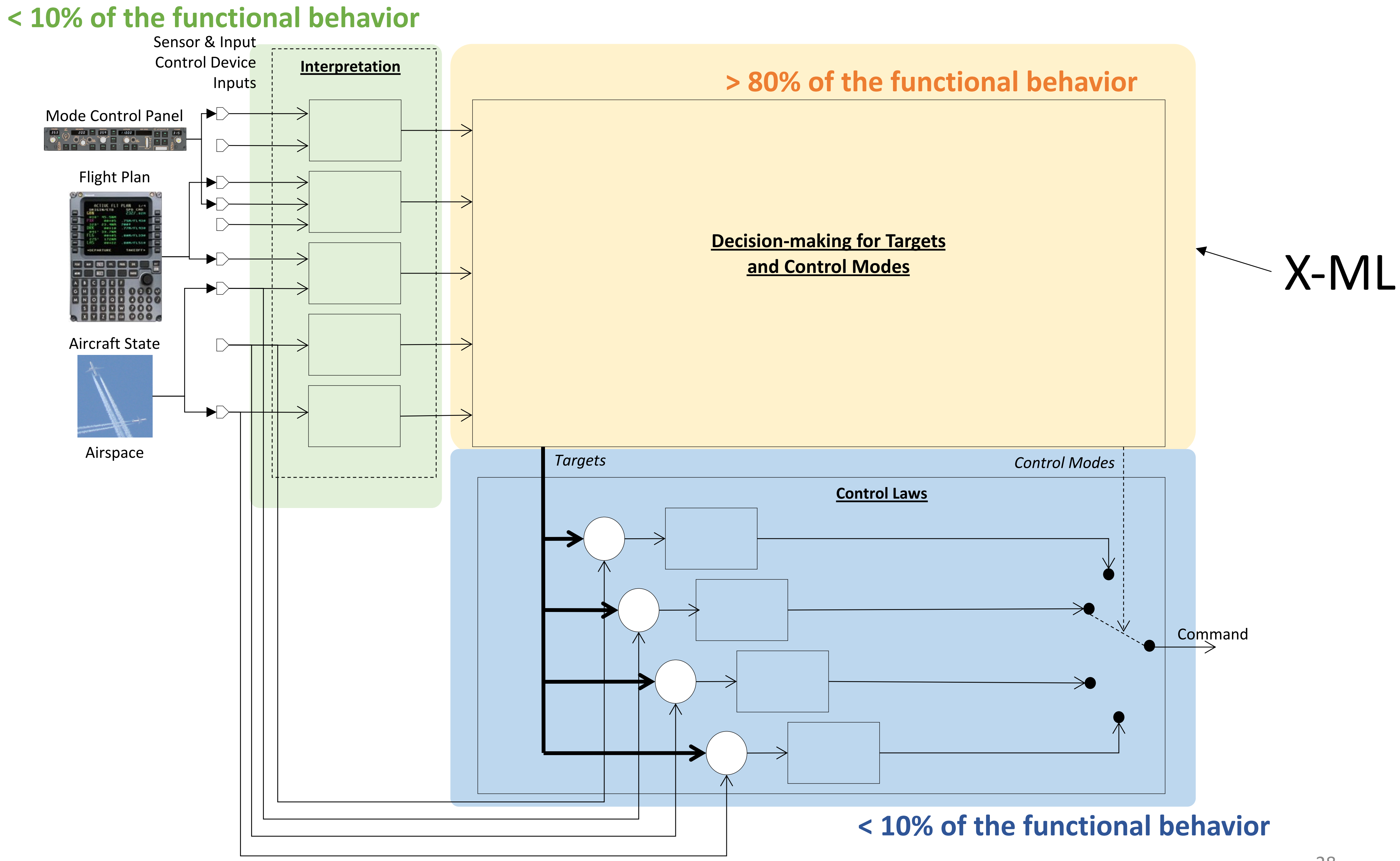
G&CF (Inputs, Outputs)	**Fixed Wing**	**Automobile**
Input: 4-D Planned Route	“Flight plan” • 4-D • Navigation Procedures • Air Traffic Control • Traffic avoidance • Terrain avoidance • Env. – Windshear	“Route” • 4-D • Roadway Rules • Signage and Traffic Lights • Traffic avoidance • Terrain avoidance • Env. – surface conditions, visibility
Output: Commands	• Elevator • Aileron • Rudder • Thrust	• Accelerator/Brake • Steering

Example OECS: Vehicle Guidance and Control Function

- Three components:
  1. Control Laws
     - Closed-loop control laws (continuous mathematics)
     - Designed based on models of vehicle and actuator dynamics
  2. Decision-making for Targets and Control Modes
     - Decision (logic)
     - Designed based on:
       - Closed-loop control law operational boundaries
       - Vehicle performance operational limits
       - Mission operational rules and constraints
  3. Interpretation
     - Translate sensor/user-interface input data into operationally meaningful mission data



Sensor & Input
Control Device
Inputs
Interpretation
Mode Control Panel
Flight Plan
Aircraft State
Airspace
Decision-making for Targets and Control Modes
Targets
Control Modes
Control Laws
Command



< 10% of the functional behavior
Sensor & Input Control Device Inputs
Interpretation
> 80% of the functional behavior
Mode Control Panel
Flight Plan
Aircraft State
Airspace
Decision-making for Targets and Control Modes
X-ML
Targets
Control Modes
Control Laws
Command
< 10% of the functional behavior

Decision-making (DM) for Targets and Control Modes (T&CM)

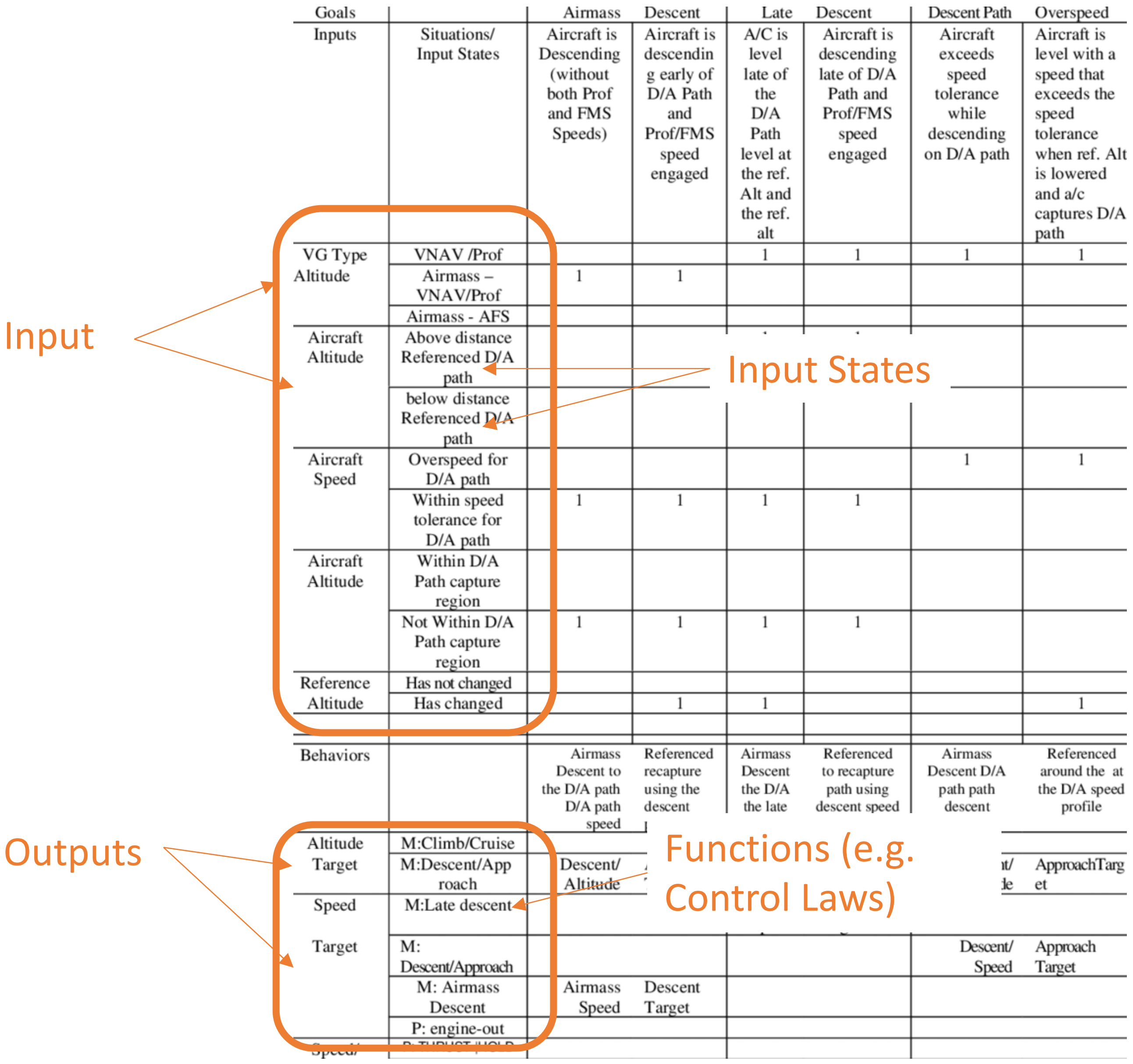
- Inputs to DM-T&CM
  - Inputs (States)
    - Examples
      - Landing Gear (Up, Down)
      - Flightphase (Taxi, Takeoff, Climb, Cruise ,. ….)
      - Aircraft Altitude (< Ref Alt – 250 ft, Between Ref Alt -250 ft and Ref Alt + 250 ft, > Ref Alt + 250 ft)
  - ~ 150 inputs (Avg 3 states)
- Outputs from DM-T&CM
  - Targets for Altitude, Airspeed, Vertical Speed
  - Control Mode

Formalism for Capturing DM for T&MC

- Situation-Goal-Behavior (SGB) Tables
- Properties of SGB Table
  - Formal model
    - Executable
    - Analyzable (e.g. logical inconsistencies)
  - Captures “Operations” from Operators Perspectives
    - What is doing now?
      - Goal
      - Behavior
    - Why is it doing that?
      - Situations

Goals		Airmass	Descent	Late	Descent	Descent Path	Overspeed
Inputs	Situations/ Input States	Aircraft is Descending (without both Prof and FMS Speeds)	Aircraft is descendin g early of D/A Path and Prof/FMS speed engaged	A/C is level late of the D/A Path level at the ref. Alt and the ref. alt	Aircraft is descending late of D/A Path and Prof/FMS speed engaged	Aircraft exceeds speed tolerance while descending on D/A path	Aircraft is level with a speed that exceeds the speed tolerance when ref. Alt is lowered and a/c captures D/A path
VG Type Altitude	VNAV /Prof			1	1	1	1
	Airmass – VNAV/Prof	1	1				
	Airmass - AFS						
Aircraft Altitude	Above distance Referenced D/A path			1	1		
	below distance Referenced D/A path						
Aircraft Speed	Overspeed for D/A path					1	1
	Within speed tolerance for D/A path	1	1	1	1		
Aircraft Altitude	Within D/A Path capture region						
	Not Within D/A Path capture region	1	1	1	1		
Reference Altitude	Has not changed						
	Has changed		1	1			1
Behaviors		Airmass Descent to the D/A path D/A path speed	Referenced recapture using the descent profile	Airmass Descent the D/A the late profile	Referenced to recapture path using descent speed	Airmass Descent D/A path path descent	Referenced around the at the D/A speed profile
Altitude Target	M:Climb/Cruise						
	M:Descent/Approach	Descent/ Altitude	Approach Target	Descent/ Altitude	ApproachTarget	Descent/ Altitude	ApproachTarget
Speed Target	M:Late descent			Late Speed	Descent Target		
	M: Descent/Approach					Descent/ Speed	Approach Target
	M: Airmass Descent	Airmass Speed	Descent Target				
	P: engine-out						
Speed/	P: THRUST |HOLD						

Situation-Goal-Behavior (SGB) Model for Op Embedded Control System



Goals		Airmass	Descent	Late	Descent	Descent Path	Overspeed
Inputs	Situations/ Input States	Aircraft is Descending (without both Prof and FMS Speeds)	Aircraft is descendin g early of D/A Path and Prof/FMS speed engaged	A/C is level late of the D/A Path level at the ref. Alt and the ref. alt	Aircraft is descending late of D/A Path and Prof/FMS speed engaged	Aircraft exceeds speed tolerance while descending on D/A path	Aircraft is level with a speed that exceeds the speed tolerance when ref. Alt is lowered and a/c captures D/A path
VG Type	VNAV /Prof			1	1	1	1
Altitude	Airmass – VNAV/Prof	1	1				
	Airmass - AFS						
Aircraft Altitude	Above distance Referenced D/A path						
	below distance Referenced D/A path						
Aircraft Speed	Overspeed for D/A path					1	1
	Within speed tolerance for D/A path	1	1	1	1		
Aircraft Altitude	Within D/A Path capture region						
	Not Within D/A Path capture region	1	1	1	1		
Reference Altitude	Has not changed						
	Has changed		1	1			1
Behaviors		Airmass Descent to the D/A path D/A path speed	Referenced recapture using the descent	Airmass Descent the D/A the late	Referenced to recapture path using descent speed	Airmass Descent D/A path path descent	Referenced around the at the D/A speed profile
Altitude Target	M:Climb/Cruise						
	M:Descent/Approach	Descent/ Altitude					ApproachTarget
Speed	M:Late descent						
Target	M: Descent/Approach					Descent/ Speed	Approach Target
	M: Airmass Descent	Airmass Speed	Descent Target				
	P: engine-out						

How to read an SGB:

- Inputs
- Input/States
- Outputs
- Output/Functions

Situation-Goal-Behavior (SGB) Model for Op Embedded Control System

Goals		Airmass	Descent	Late	Descent	Descent Path	Overspeed
Inputs	Situations/ Input States	Aircraft is Descending (without both Prof and FMS Speeds)	Aircraft is descendin g early of D/A Path and Prof/FMS speed engaged	A/C is level late of the D/A Path level at the ref. Alt and the ref. alt	Aircraft is descending late of D/A Path and Prof/FMS speed engaged	Aircraft exceeds speed tolerance while descending on D/A path	Aircraft is level with a speed that exceeds the speed tolerance when ref. Alt is lowered and a/c captures D/A path
VG Type	VNAV /Prof			1	1	1	1
Altitude	Airmass – VNAV/Prof	1	1				
	Airmass - AFS						
Aircraft Altitude	Above distance Referenced D/A path						
	below distance Referenced D/A path						
Aircraft Speed	Overspeed for D/A path						1
	Within speed tolerance for D/A path	1	1	1	1		
Aircraft Altitude	Within D/A Path capture region						
	Not Within D/A Path capture region	1	1	1	1		
Reference Altitude	Has not changed						
	Has changed		1	1			1
Behaviors		Airmass Descent to the D/A path D/A path speed	Referenced recapture using the descent profile	Airmass Descent the D/A the late	Referenced to recapture path using descent speed	Airmass Descent D/A path path descent	Referenced around the at the D/A speed profile
Altitude Target	M:Climb/Cruise						
	M:Descent/Approach	Descent/ Altitude	pproach arget				roachTarg
Speed	M:Late descent						
Target	M: Descent/Approach						roach et
	M: Airmass Descent	Airmass Speed	Descent arget				
	P: engine-out						
Speed/	P: THRUST |HOLD						

Situation = combination of Input States

Behavior = Combination of Functions

How to read an SGB:

- Inputs
- Input/States
- Outputs
- Output/Functions
- Situations (combinations of Input States)
- Behavior (combinations of Functions)

Situation-Goal-Behavior (SGB) Model for Op Embedded Control System

Goals		Airmass	Descent	Late	Descent	Descent Path	Overspeed
Inputs	Situations/ Input States	Aircraft is Descending (without both Prof and FMS Speeds)	Aircraft is descendin g early of D/A Path and Prof/FMS speed engaged	A/C is level late of the D/A Path level at the ref. Alt and the ref. alt	Aircraft is descending late of D/A Path and Prof/FMS speed engaged	Aircraft exceeds speed tolerance while descending on D/A path	Aircraft is level with a speed that exceeds the speed tolerance when ref. Alt is lowered and a/c captures D/A path
VG Type Altitude	VNAV /Prof			1	1	1	1
	Airmass – VNAV/Prof	1	1				
	Airmass - AFS						
Aircraft Altitude	Above distance Referenced D/A path						
	below distance Referenced D/A path						
Aircraft Speed	Overspeed for D/A path						1
	Within speed tolerance for D/A path	1	1	1	1		
Aircraft Altitude	Within D/A Path capture region						
	Not Within D/A Path capture region	1	1	1	1		
Reference Altitude	Has not changed						
	Has changed		1	1			1
Behaviors		Airmass Descent to the D/A path D/A path speed	Referenced recapture using the descent profile	Airmass Descent the D/A the late	Referenced to recapture path using descent speed	Airmass Descent D/A path path descent	Referenced around the at the D/A speed profile
Altitude Target	M:Climb/Cruise						
	M:Descent/Approach	Descent/ Altitude	Approach Target				roachTarg
Speed	M:Late descent						
Target	M: Descent/Approach						roach et
	M: Airmass Descent	Airmass Speed	Descent Target				
	P: engine-out						
Speed/	P: THRUST |HOLD						

Situation = combination of Input States

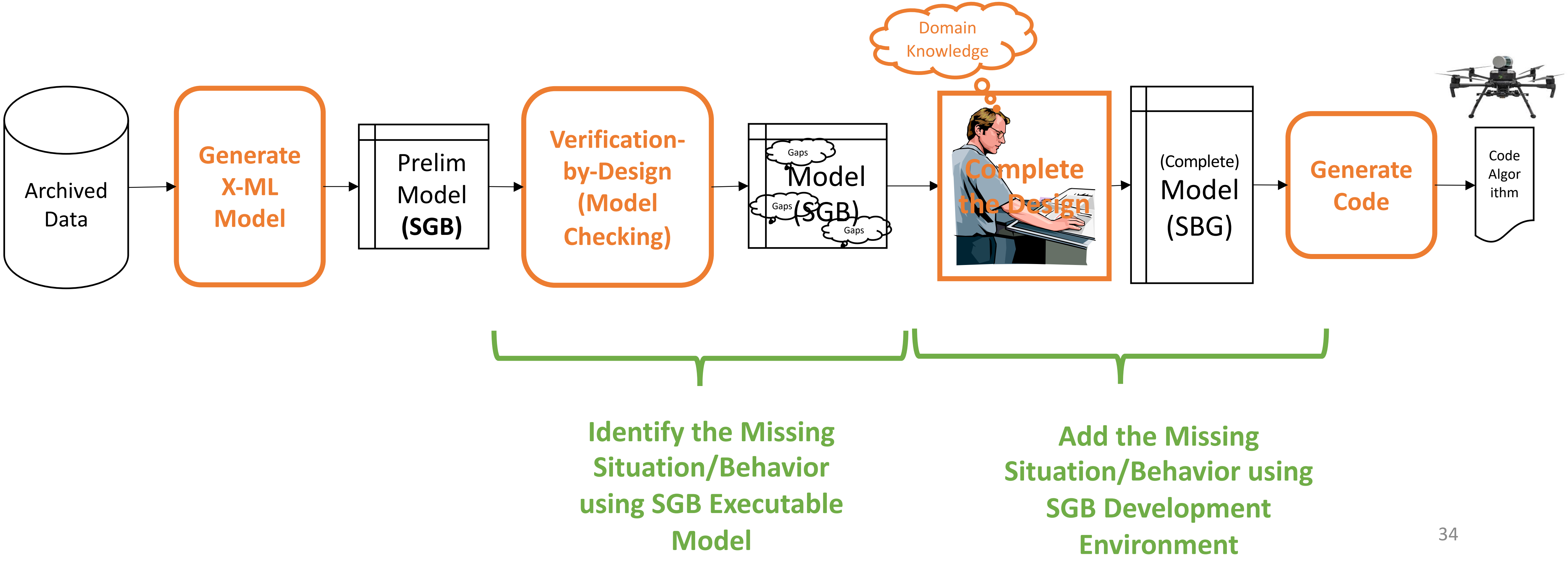
Behavior = Combination of Functions

SGB: one-stop-shopping for Certification (DO-178)

- Design-by-Verification
  - Logical Completeness
  - Logical Consistency
- Missing "behavior"

SGB: auto-generation of code

Collaborative Functional Design Using X-ML

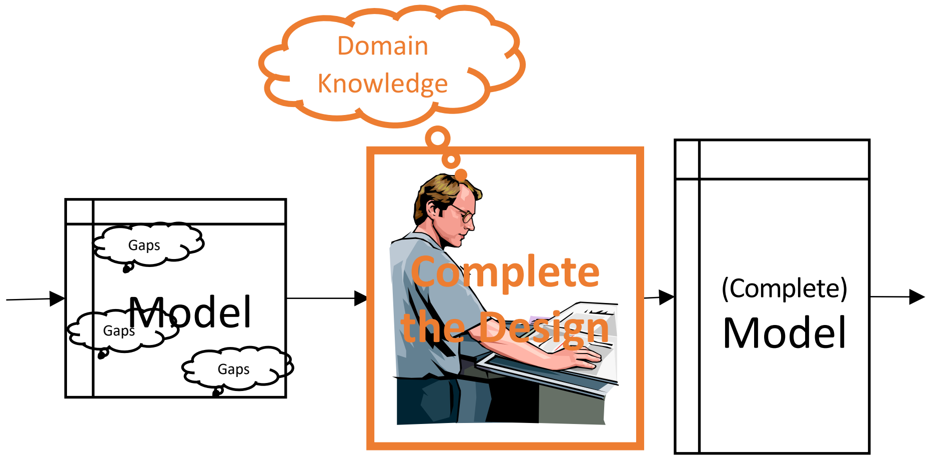


What are the Gaps in X-ML Designs?

- X-ML Design is Missing Input
  - Design is ***absent one or more of the required inputs*** (i.e. sensors/data feeds) to identify one or more of the operational situations that must be covered by the operationally embedded system
- X-ML is Missing Input/State Combinations
  - Given all the required inputs, the design is ***absent one or more combinations of input states*** to respond to *all* the operational situations that must be covered by the operationally embedded system
- X-ML is Missing Mapping between Input/State Combinations to Behaviors
  - Given the required inputs to support all the combinations of input states and all the combinations of input states, the design is ***absent one or more the correct mappings*** between operational situations and appropriate behaviors

Mitigating Gaps in X-ML Designs

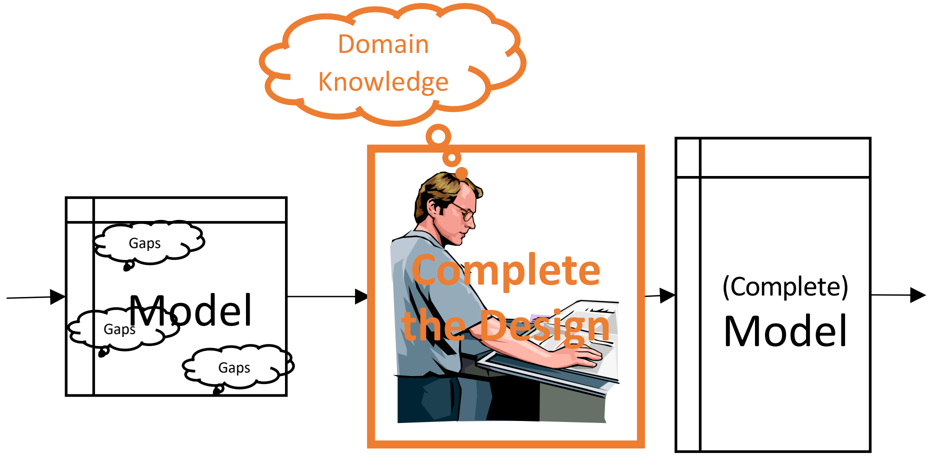
- <u>X-ML is Missing Input/State Combinations</u>
  - Given all the required inputs, the design is ***absent one or more combinations of input states*** to respond to *all* the operational situations that must be covered by the operationally embedded system
- Use SGB Model to generate Complete Design
  - Generate all the combinations of Input/States
- Use SGB Model to generate Consistent Design
  - Make sure no duplicate combinations of Input/States



Mitigating Gaps in X-ML Designs

- <u>X-ML is Missing Mapping between Input/State Combinations to Behaviors</u>
  - Given the required inputs to support all the combinations of input states and all the combinations of input states, the design is ***absent one or more the correct mappings*** between operational situations and appropriate behaviors

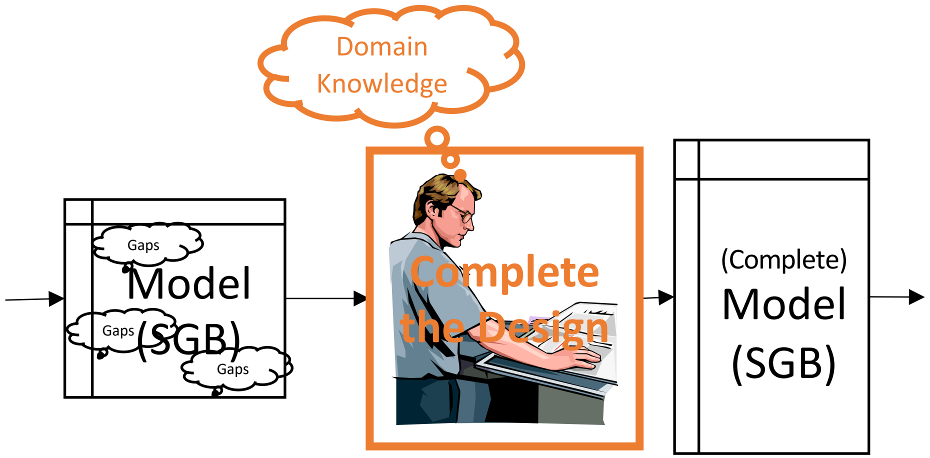
- Use SGB Model to generate Complete Design
  - Check each legal Behavior



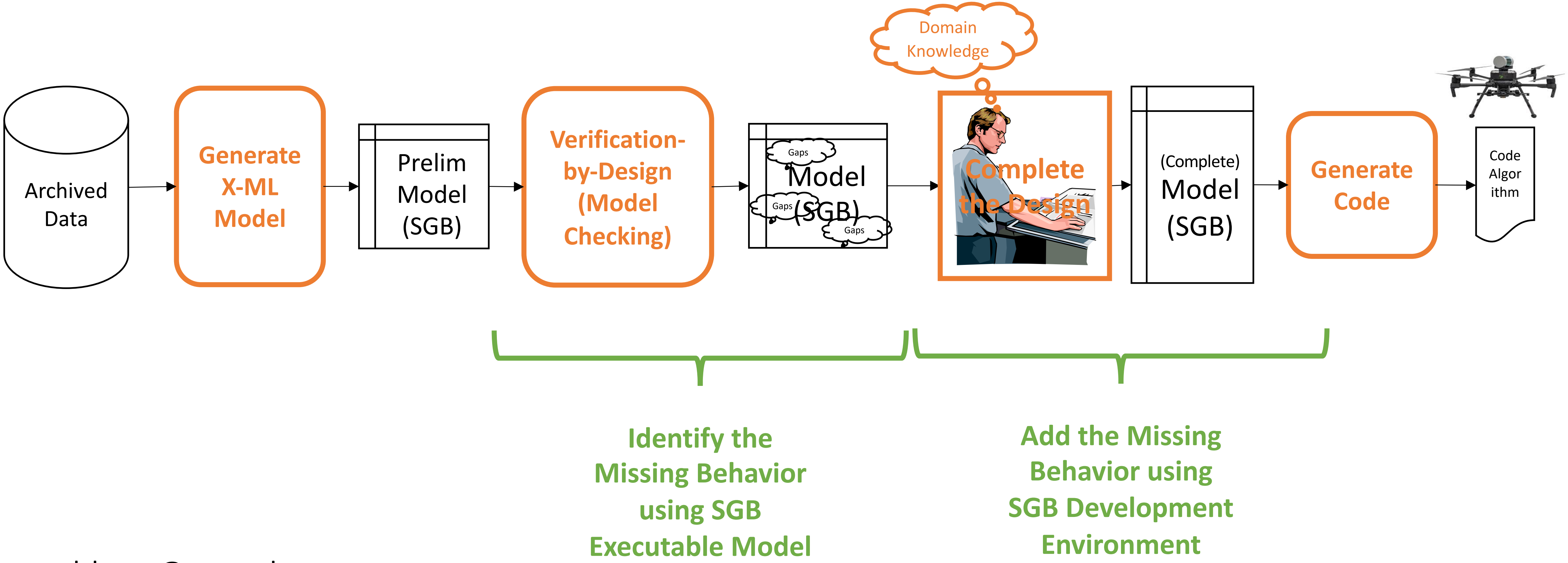
Mitigating Gaps in X-ML Designs

- <u>X-ML Design is Missing Input</u>
  - Design is ***absent one or more of the required inputs*** (i.e. sensors/data feeds) to identify one or more of the operational situations that must be covered by the operationally embedded system

- Scenario Analysis/Use Cases
- Hazard Analysis



Realistic Use of X-ML:
Collaborative Functional Design Using X-ML



lsherry@gmu.edu