UNCLASSIFIED

Assuring Behavior of Autonomous (UxV) Systems

J. Lee, A. Prajogi, E. Rafalovsky, P. Sarathy

1. Problem Statement

Limitation of Standards

All Major DoD and Standards/Guidelines rely on comprehensive functional testing as a key method to assess Safety Certification compliance. Current civilian (DO-178B/C) and military (MIL-STD-882E, MIL-STD-516, JSSSEH) directives process guidance documents with absolutely no established objective (functional or otherwise).

- Based on defined processes to be followed to gather certification evidence
- Requires that every single possible scenario (test case) be tested to certify software for use

Combinatorial Testing Complexity

For major classes of UxS software the underlying problem space is often combinatorially large; the solution space is consequently uncountable as well. Comprehensive testing approaches are thus infeasible to achieve the desired level of confidence required for Safety Certification. Systems growing exponentially in both size and complexity render current verification methods infeasible. For future UxS to be included into mainstream civilian and military operational use, certification of these systems demands radically different paradigm

2. Background Relevance

Mixed Criticality Challenge

Mixed Criticality is a key driver of certification challenges for future UxS. Increased mission complexity requires autonomous capabilities in these UxS:

a) Exponential increase in size
b) Complexity of software applications
c) Autonomy of system

Operational Needs

For unmanned systems to begin to approach levels of utility similar to human-based systems, the following operational needs can be identified:

a) UxS will need to be mission effective under uncertain operating conditions
b) UxS will need to respond to dynamic changes in the environment
c) UxS will need to be survivable under adverse conditions

Key Characteristics

a) Real Time Decision Making: UxS will have to be capable of decision making in real-time to respond to changes in the operating conditions
b) Autonomous Operations: UxS will need the capability to operate in the absence of direct command and control (C2) from an external source
c) Situation Assessment: UxS will need to be capable of comprehending relevant portions of its operating environment and any changes within it that affects its mission
d) Adaptive Planning: UxS will need to use some form of planning to capture its real-time decisions and thus execute this plan in an effective fashion

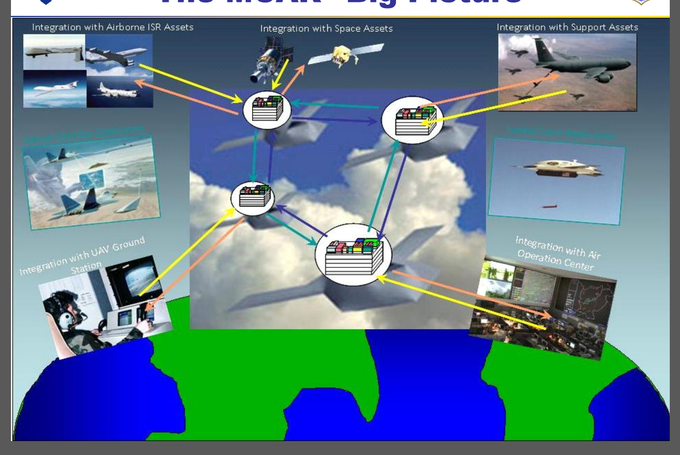
Complex software is necessary to meet these needs, utilizing algorithms that exhibit a high degree of self-directed behavior and involving exponentially large input and output state spaces.

3. Historical Context

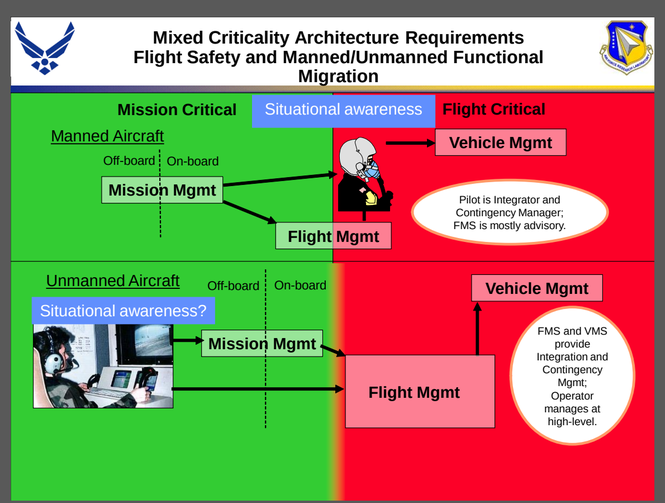
Mixed Criticality Systems

A mixed-critical system is an integrated suite of hardware, operating system and middleware services and application software that supports the execution of safety-critical, mission-critical, and non-critical software within a single, secure compute platform.

The MCAR "Big Picture"



Mixed Criticality Architecture Requirements Flight Safety and Manned/Unmanned Functional Migration



4. Objectives and Scope

Alternative Assurance Paradigm

a) Defines a finite set of functional characteristics or traits that are of primary importance for UxS operations
b) Includes notions that are typically safety-related (failures of subsystems functions) but also mission failures (loss of survivability and vehicle integrity)
c) Defines a means of establishing criteria and thresholds for each

Candidate Implementation Methods

Develop verification methodology spanning the software lifecycle. This implies the evidence to be collected covers each of the phases of the software engineering lifecycle, namely:

a) Requirements
b) Design
c) Implementation
d) Verification

For specific software architectural constructs, identify the impact of using bounded assurance with respect to testing and verification methods

5. Proposed Approach

Assurance Methodology

a) Define set of safety critical function chains relevant for assessment of behavior assurance
b) For each function chain develop a set of bounds in terms of relevant input and output state spaces
c) Establish verification methods for each functional chain to ensure compliance

Compliance Methods

- Assess Behavior: For each phase of the software lifecycle, identify and establish methods to assess the behavior of each functional chain including:
  a) Requirements Traceability
  b) Design Analysis
  c) Implementation Guidance
- Test Verification
- Evaluate Boundedness compliance: Using the assessments made for each lifecycle phase, generate a composite evaluation of system behavior assurance

6. Key Components

Mixed Initiative C2

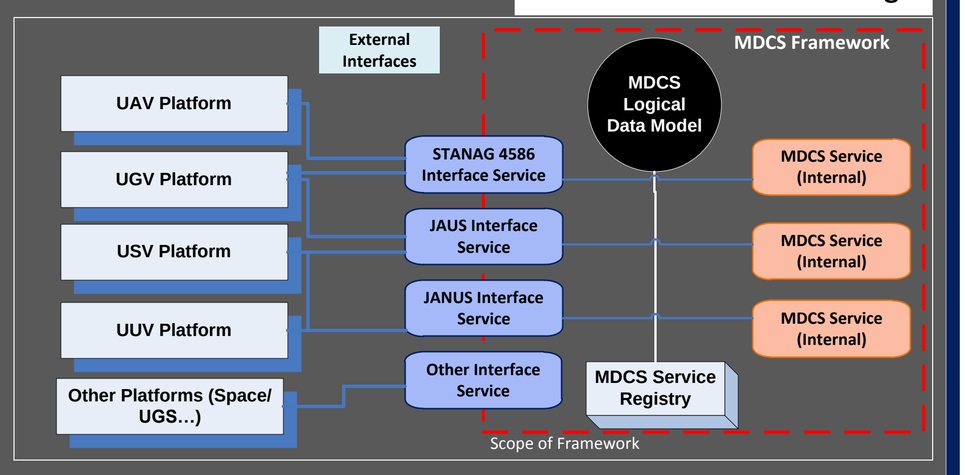
Autonomous Operations
- Reduces C2 latencies
- Reduces comm bandwidth
- Mitigates effects of comm. loss
- Compresses mission execution timeline
- Increases survivability
- Allows direct and/or distributed C2

Automated Processing
- Improves accuracy
- Reduces human error
- Supports operator decision making
- Reduces operator workload
- Reduced manpower and staffing needs

Human Decision Making
- Capable of meta-reasoning
- Most reliable decision-making framework
- Best use of human expertise and experience

Flexible Levels of Autonomy
- Configuration of autonomy spectrum
- Ack-Approve-Auth-Override-Manual
- Configure at planning-time or real-time
- Dynamic configurability
- Actual selections will vary over time and situation

Standards Based Design



Functional Multi-level Testing

SYS 0
- Service Testing / Unit Testing
  - Specification Compliance
    - CCS Compliance
    - Verifies fulfillment of Service Description requirements
  - Verifies software meets its individual requirements
  - Incoming and outgoing
  - Specific safety tests performed as needed
  - Has no official entry or exit criteria

SYS I
- Service Container Testing / SWIT
  - Testing of Applications and Services
  - Verifies compliance to Service Container SRS
  - Integration of the individual software units
  - Official verification of the SRS requirements
  - Implementation of specific safety tests should be performed as needed
  - Regression testing will be performed only for functionality successfully implemented in prior sprints

SYS II
- Segment Testing / SIT
  - Integrated Service Containers tested
  - SIT verifies integration of system components and external interfaces
  - Verifies SSS requirements
  - Regression testing performed on functionality implemented in prior sprints
  - Entrance criteria: Successful completion of SWIT and SIT
  - Exit criteria : No Priority 1 or 2 defects & No Priority 3 defects that combine to form a Priority 1 or 2 defect

SYS III
- SIL Acceptance Testing / FQT
  - Service Containers integrated and installed at RB SIL
  - Verification that the system meets the system level requirements
  - All safety related and critical SSS verified
  - Entrance criteria is the following: Successful completion of Formal SIT
  - Test Readiness Review Complete
  - Exit criteria is the following:
  - No Priority 1 or 2 defects& No Priority 3 defects that combine to form a Priority 1 or 2 defect

7. Expected Outcome and Next Steps

Expected Outcomes

- Reduction in Test Burden: Significant reduction in total number of test cases needed to fully assess behavior assurance
- Responsive Assurance Framework for Advanced UxS Operations: Proposed framework allows for adaptability and customization to different UxS platform configurations

Next Steps

- Develop multiple instantiations of approach for different UxS configurations
- Develop usage guideline for methodology

NORTHROP GRUMMAN

UNCLASSIFIED