Run-Time Assurance for Safety-Critical Flight Control Systems

A. Aiello, J. Berryman, J. Grohs, J. Schierman
Barron Associates, Inc., Charlottesville, Virginia

David Homan, Russ Urzi, Jonathan Hoffman
Air Force Research Laboratory
Control Systems Development and Applications

Safe and Secure Systems and Software Symposium (S5)
15-17 June 2010
Wright-Patterson AFB
AFRL's Flight Critical System Software Initiative (FCSSI)

Objective - Enable critical capabilities for Unmanned Aircraft Systems (UAS)

Airspace Integration

Dynamic Mission Op's

UAS Aerial Refueling

Technical Challenge – Current V&V practices limited in addressing certification of these advanced functionalities

Advancing Design-Time V&V Methodologies

New Certification Approaches: Feasible & Affordable

Developing Run-Time Assurance Approaches

Key Attributes
- Adaptive
- Mixed-Initiative
- Man-Machine Integration
- Systems-of-systems

Exhaustive Testing – Neither feasible nor affordable

Air Force Research Laboratory
With run-time assurance capabilities...

- Add additional measures of safety during operation to “cover” approximations and assumptions made in design-time V&V methods
- Designed to catch mistakes and recover from them during operation
- Helps reduce criticality of advanced software – which reduces their V&V costs

Exhaustive design-time testing infeasible for such highly complex systems

Autonomy, intelligence, nonlinear systems, non-determinism, other complexities present new challenges
**Advanced System:**
- High risk software (full-envelope, full-featured)
- Enabled at all times under nominal conditions (no problems detected)
- Intelligent, reconfigurable, learning, adaptive, non-deterministic...
- Difficult to fully certify at design time w/o great cost
Reversionary System:
- System that can be certified offline using traditional methods
  - Does not possess advanced elements that cannot be certified
- Envisioned to be full-up, full-envelope control system from previous, certified design
  - Some re-design/re-tune may be required, but should be minimized to reduce V&V costs
- Should provide recovery and “return-to-base” capabilities
**Elements of Run-Time Assurance “Wrappers”**

- **Monitoring & Switching Process:**
  - Continually observe state of feedback system
  - Determine if safety will be compromised under Advanced System
  - If so:
    - Disable Advanced System
    - Switch to Reversionary System
  - Monitor, switching code must be certified at design time
Supporting Code:
- Example: Fault detection - differentiate between poor performance due to airframe damage/failures and poor performance due to software/computing problems
  - If Advanced System has fault tolerant capabilities, may choose to allow it to run under hardware faults even if monitor indicates unsafe operations
- Wrapper manager – coordinate reversion for multiple RTA wrappers
- Certified at design time
Implementing Run-Time Assurance

- The safety argument
  - Assure system safety while running un-trusted software

  *Build RTA wrappers such that the system is as safe as if only the reversionary (trusted) software were running all the time*

- Design goals
  - Achieve soundness
    - Only information about the *trusted* software should be used
    - Decisions should not be based on information about the un-trusted software
  - Achieve completeness
    - Safety must be assured in the presence of arbitrary un-trusted software failures

- At each time step we ask only one question
  - If the system continues to operate the un-trusted software, can the trusted software be guaranteed to maintain safe flight if we then switch to the trusted system?
  - If not, safe flight may or may not be maintained if we revert
  - *Let’s never get into this situation!*
Basic Premise to Staying Safe

Program Overview  RTA Introduction  RTA Overview  CPD Program  Cost Studies  Conclusions

Reversionary Controller’s Certified Envelope

Nomenclature: Reversionary Safety Envelope (RSE)
Basic Premise to Staying Safe

Program Overview
RTA Introduction
RTA Overview
CPD Program
Cost Studies
Conclusions

Air Force Research Laboratory

Reversionary Controller’s Certified Envelope

Operation under reversionary control system

Current

Predict ahead – if apply advanced controller’s output, then revert, will system remain within certified envelope?
Basic Premise to Staying Safe

Do not revert

Within safe envelope – no need to revert, go ahead and apply advanced controller’s output

Reversionary Controller’s Certified Envelope

Within safe envelope – no need to revert, go ahead and apply advanced controller’s output
Basic Premise to Staying Safe

Reversionary Controller’s Certified Envelope
Basic Premise to Staying Safe

Reversionary Controller’s Certified Envelope

Predict ahead – no longer in safe envelope – do not apply advanced controller’s output - revert now
Basic Premise to Staying Safe

Program Overview  RTA Introduction  RTA Overview  CPD Program  Cost Studies  Conclusions

Reversionary Controller’s Certified Envelope

Current

Remain within Certified Envelope
How do we define envelopes?

What states, critical parameters should be monitored?
  - Critical sets may differ for different flight conditions

How do we predict?
  - How far in the future do we predict?
  - How long will it take for reversionary system to take control?

How do we switch to reversionary system?
  - Stability guarantees during control system switching?
  - Characteristics of transients?
  - Bumpless transfer control problem...
    - Integrator match-up issues, etc.
Only need information about certified envelope of reversionary system

- If using prior control design – most envelope information available
- Further refinement could be achieved through additional simulation analysis

Gain scheduled controller can define some envelopes on some of the states/parameters

Other parameters:
\( \alpha, \beta, p, q, r, \phi, \theta, \psi, \text{ etc.} \)
Option #1 – all processes performed on-line:

- Predict on-line, during flight using trusted, high fidelity full-DOF simulation
- Pros:
  - Most accurate, least conservative approach
  - Switch to reversionary system only when actually required
- Cons:
  - Most likely not feasible due to computation burden & time requirements

Predict ahead at each control update
Option #2 – all prediction processes performed off-line

- Use high fidelity simulation studies to generate “buffered” boundaries
- On-line, check that buffered boundaries are not violated

Pros:
- Least on-line computations required
  - No on-line predictions performed

Cons:
- Most conservative approach
- Requires greatest margins

Staying within “buffered” envelope ensures aircraft will always be safe

Nomenclature: 
Recovery Achievability Envelope (RAE)
Option #3 – Combination of Options #1 and #2

- Simplified prediction on-line with less conservative “buffered” envelope
- Example: use local linear model to predict ahead

Less “buffer” means more flight envelope can be used
Three main steps to reversionary process

- Inner-loop flight controller reversion
  - Transition from advanced inner-loop controller to reversionary controller
  - Maintain attitude stability of aircraft
Three main steps to reversionary process

- Outer-loop guidance reversion
  - Perform “evasive” maneuver
  - Much of this technology currently developed or in development
    - Terrain/collision avoidance, etc.
The Reversionary Process

Three main steps to reversionary process

- Outer-loop guidance reversion
  - Perform “evasive” maneuver
  - Much of this technology currently developed or in development
    - Terrain/collision avoidance, etc.

Autoland – “wave off” - pull up, go wings level

Diagram:
- Outer Loop Guidance
- Inner Loop Control

[Diagram showing the reversionary process with a flowchart and a plane illustration]
Three main steps to reversionary process

- Outer-loop guidance reversion
  - Perform “evasive” maneuver
  - Much of this technology currently developed or in development
    - Terrain/collision avoidance, etc.

Formation flight – carefully disengage from fleet to safe distance away

UAS Aerial Refueling
Three main steps to reversionary process

- **Outer-loop mission reversion (navigation)**
  - Decide whether to continue mission under crippled state
  - Fly altered, less critical mission (communications support, reconnaissance, etc.)
  - Or return to base
CerTA FCS Challenge Problem Demonstration Program

- Realizable, Real World Application of Run-Time Assurance Framework
- Validate Benefits of Advanced Verification & Validation Technologies
Run-Time Assurance Framework for CPD

---

Program Overview  RTA Introduction  RTA Overview  CPD Program  Cost Studies  Conclusions

---

Advanced System
Integrated Autoland Algorithm

Reversionary System
Automatic Ground Collision Avoidance System Algorithm

Encoded RAE
Generic Autoland RAE Design Space

Safety Monitor
RAEQuery Algorithm

---

Autoland Algorithm

AGCAS Algorithm

CPD RAE

Control

Trigger

State

Rockwell Collins

Barron Associates

Lockheed Martin

Air Force Research Laboratory
Recovery Availability Envelope Design Space

Scope of RAE Design Space
- Advanced System Operating Envelope
  - Verification of Advanced System Requirements
- Flight Envelope Where Recovery is Achievable
  - Validation of Safe Operational Design Requirements

Depth of RAE Design Space
- Number of States (Dimensionality)
- Design Space Recursion Depth
- State Input Ranges (State Space)
- State Variables Definitions

Evaluation of RAE Design Space
- “Oracle” Evaluation Function
  - Input: RAE State Vector
  - Output: “WithinRAE” Boolean

Determines Number of Potential Vertices
Determines Fidelity & “Well-Formed” Nature
Longitudinal States Which Contribute to System Safety

Provides Encoded RAE Responses to:

- Approach Speed Differences
- Vertical / Horizontal Wind Components
- Approach Angle of Attack Deviations
- Approach Corridor Positioning
- Sink Rate Based on Pitch Rate & Flight Path Angle

<table>
<thead>
<tr>
<th>State</th>
<th>State Name</th>
<th>Reason for State Inclusion</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Axial Distance</td>
<td>Position Vector</td>
<td>ft</td>
</tr>
<tr>
<td>H</td>
<td>Vertical Distance</td>
<td></td>
<td>ft</td>
</tr>
<tr>
<td>Vx</td>
<td>Axial Velocity</td>
<td>Flight Path Vector</td>
<td>ft/s</td>
</tr>
<tr>
<td>Vz</td>
<td>Vertical Velocity</td>
<td></td>
<td>ft/s</td>
</tr>
<tr>
<td>Vc</td>
<td>Calibrated Airspeed</td>
<td>Ability to Change Flight Path (Aerodynamics)</td>
<td>kts</td>
</tr>
<tr>
<td>α</td>
<td>Angle of Attack</td>
<td>Resistance to Flight Path Change</td>
<td>deg</td>
</tr>
<tr>
<td>qb</td>
<td>Pitch Rate</td>
<td>Flight Dynamics &amp; Ability to Change Attitude</td>
<td>deg/s</td>
</tr>
<tr>
<td>Wt</td>
<td>Weight</td>
<td></td>
<td>lb</td>
</tr>
<tr>
<td>Thr</td>
<td>Thrust</td>
<td>Ability to Change Flight Path (Propulsion)</td>
<td>lb</td>
</tr>
<tr>
<td>θ</td>
<td>Pitch Attitude</td>
<td></td>
<td>deg</td>
</tr>
<tr>
<td>Iyy</td>
<td>Pitch Moment of Inertia</td>
<td></td>
<td>lb ft²</td>
</tr>
<tr>
<td>δe</td>
<td>Pitch Effector Deflection</td>
<td></td>
<td>deg</td>
</tr>
</tbody>
</table>
RAE for Autoland – Evaluation Function

Target Approach Corridor Cross Section & Example Evaluation Function 'RAETest_LongInert' Points

- x (ft) - Axial Distance
- h (ft) - Altitude Above Runway

Air Force Research Laboratory

Conclusions
Target Approach Corridor Cross Section & Encoded RAE 'CPD' & M Code Evaluation

<table>
<thead>
<tr>
<th>Max Depth</th>
<th>Min Depth</th>
<th>Vertices</th>
<th>Evaluations</th>
<th>Vertex Reuse</th>
<th>Encoded RAE Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>[5,4,2,4,2,2,2]</td>
<td>[3,2,1,1,1,1,1]</td>
<td>5960625</td>
<td>1811961</td>
<td>24043467</td>
<td>66699 x 3</td>
</tr>
</tbody>
</table>

Stateflow RAE Evaluation - # of Division Iterations

- Frequency
- # of Divisions to Reach RAE Decision

x (ft) - Axial Distance

h (ft) - Altitude Above Runway

RAE for Autoland – Encoded RAE
Run-Time Assurance Simulation Results

- **Autoland Run-Time Assurance Framework & Data**
  - Embedded Into Real Time Flight Software
- **Autocoded, Compiled & Integrated into Simulation**
- **Evaluation of Run-Time Assurance on Final Approach**
  - Variations in Winds
  - Variations in Turbulence
  - Failure Scenarios
    - Nominal Approach
    - Uncommanded Pitch Control Failures
- **Run-Time Assurance Mode Selection**
  - Run-Time Assurance Monitor Observing
  - Run-Time Assurance Monitor Armed

---

Nominal Approach

Pitch Control Failure: With & Without Run-Time Assurance
RTA Simulation – Nominal w/ Turbulence

Approach Corridor & Aircraft Position - cpd/sim_016_w0_wd0_t2_normal_a50_rarm.gend

Nominal w/ Moderate Turbulence
Advanced System Trajectory ≠
RTA Generic Autoland Trajectory
RTA Simulation – Pitch Failure – RTA Observing

Approach Corridor & Aircraft Position - cpd/sim_017_w0_wd0_t2_pitch_a50_robs.gend

Pitch Control Fail - 50' AGL
Run-Time Assurance Observing
Short Landing - 15 ft/s Sink Rate
RTA Simulation – Pitch Failure – RTA Armed

Approach Corridor & Aircraft Position - cpd/sim_018_w0_wd0_t2_pitch_a50_rarm.gend

Pitch Control Fail - 50' AGL
Run-Time Assurance Armed
Encoded RAE → Reversion
Successful AGCAS Recovery
Run-Time Assurance Cost Model Development

Cost: \( f(\text{Complexity}, \text{Criticality}) \)

- Baseline Advanced System Cost: 200
- Run-Time Assurance Infrastructure Cost: Reversionary System + RAE Design + Safety Monitor: 50 + 40 + 24 = 114
- Run-Time Assurance Advanced System Cost: 64

Baseline System Cost: 200
Run-Time Assurance Cost: 178
Run-Time Assurance Savings: 11%
Run-Time Assurance Accomplished Real-Time Evaluation of an Encoded RAE to Provide Safety for an Autoland System on Final Approach

**Autoland**
- Expansion of Autoland Design To Larger Flight Envelope
- Conjoined Encoded RAEs (Approach, Landing, Braking to Full Stop)

**Sense & Avoid (S&A)**
- Monitor Multi-Aircraft System Level Properties for Safety
- Extensive Conceptual S&A Techniques in Ongoing Projects

**Automated Aerial Refueling (AAR)**
- Monitor System Safety Margins During All AAR Phases
- Well Understood Breakaway Recovery Action

**Cooperative Flight**
- Monitor Multi-Dimensional Envelope for Formation Flight Separation
- Conflict Resolution Recovery Action Using TCAS-like Algorithms