MODEL-BASED FAULT MANAGEMENT FOR AEROSPACE SYSTEMS: A LOOK BACKWARDS AND FORWARDS

Ali Zolghadri
Bordeaux University - CNRS | IMS LAB
FRANCE
Do we provide a control system methodology for practicing engineers?

(Ziegler and Nichols, 1942)
This talk could be of some interest to you if you are:

- A “theoretician” looking for an excuse for your math \((A)\), or
- A “practitioner” who needs to publish \((B)\), or
- Any linear combination of those \((C)\) : \[ C = \alpha A + (1 - \alpha)B \] with \(0 < \alpha < 1; \alpha \in \mathbb{R}\)

**What not**

- New “Results” and “Theorems”

**So what then**

- An overview of model-based fault management for aerospace systems and some observations on ”turning theory into practice in aerospace”: sharing with you my experience on that…
My current research interests (1)

- Fault management in Cyber Physical Systems: hybrid, interconnected, distributed and networked systems that should satisfy some complex specifications

\[
x_i^+ \in f_1(x_1, x_2, x_3, \ldots, x_n, u_1) \\
x_2^+ \in f_2(x_2, x_1, x_3, \ldots, x_n, u_2) \\
\vdots \\
x_n^+ \in f_n(x_n, x_1, x_2, \ldots, x_{n-1}, u_n)
\]

\[
x_i \in X_i, \quad u_i \in U_i, \quad i \in I
\]

Partial information about the states of neighboring systems through ranking functions \(\psi_i\).

Flight system

\[
\dot{x}(t) = f(x(t), u(t)) + [-w, w]
\]

Discrete model

Abstraction

External behavioral relationships

Symbol: aggregate of states
My current research interests (2)

- Fault management in aircraft systems with a focus on LOC-I (Loss of Control In-flight) situations, upset recovery, autonomous navigation for future aircraft…
Basic terminology (source: NASA - JPL)

FDIR: High-Priority Technologies for Space Autonomous Systems (Steering Committee for NASA Technology Roadmaps): Technology 4.5.1, Vehicle Systems Management FDIR.

Fault Management

Fault Protection

Fault Avoidance

In-Flight Examples include:
1) Robust Design Features
   - Simple Design (e.g., fixed solar array or HGA)
   - conservative design practices and performance margins
2) On-Board Autonomy
   - (e.g., attitude constraint checking).
3) Post-Launch Operations Processes and Procedures
   - These include the use of Flight Rules and executing commands on a spacecraft simulation test-bed.

Fault Tolerance

Fault Masking

Flight System Fault Tolerance Design Strategies include:
1) Graceful Degradation
2) Application of Redundancy
   - physical
   - functional
3) Fault Containment Regions
   - limit propagation
4) FDIR

FDIR

Ground-based FDIR

For Example:
1) Telemetry Alarm Checking
2) Telemetry Trending & Analysis
3) Contingency Plans & Procedures

Flight HW FDIR

For Example:
1) OV/UV Detect Circuits
2) Watchdog Timers
3) On-Chip Built-in-Tests
4) Memory scrubbing

Flight SW FDIR

For Example:
1) Sensor Threshold / Persistence
2) Diagnosis / Response
3) Re-plan & Continue Activity
- **Fault Detection & Identification (FDI):** Detect, isolate and estimate the severity of a fault.

- **Fault Tolerant Control (FTC):** continue to “control” the faulty system: provide, at worst, a degraded level of performance in fault situations.

  *Recovery: reshaping inner piloting loops (or/and control reallocation)*

- **Fault Tolerant Guidance (FTG):** if the available on-board control resources are limited, FTC would not be sufficient, change the objectives.

  *Recovery: reshaping outer guidance loops*
Shuttle mission STS-86

GUIDANCE

Consolidated flight data

NAVIGATION

CONSOLIDATED FLIGHT DATA

CONTROL

Sensors

Computed control signals

Actuators

J.L. Chrétien
Astronaut
NASA | ESA

August 29, 2019
**Sensors:** Flight parameters – Air Data Inertial Reference System

**GNC System**

**Navigation**
*where am I?*

**Guidance**
*where am I going?*

**Control**
*how do I get there?*

*Estimated states*

*Required changes in velocity, attitude …*

*Computed control signals*

**Actuators:** Propulsion system & Control surfaces

Fault Management
**What not:**
- Software simulations on representative benchmarks and models
- Hardware in the loop simulations
- Demonstrations on testbed platforms and ground test facilities
- Demonstration on flight simulators
- In-flight tests and evaluations
- ...

**So what then:**
- Tangible and marketable aerospace technologies which generate economic value: all software has been fully integrated with all operational hardware/software systems, flight-proven through successful mission operations.
- This means: **Entry into service: commercial flight, launching & space mission operations**...
Model vs. Reality

- The thing being modelled is a **system** in the physical world. **Models** in engineering science can be of different types providing different **designs**.
- Models have **formal** properties (not the systems) …

- So, what does it mean when a **fault** is detected in a **system** by a **model**-based design?

It “just” means that the “fault” is a violation of assumptions in the **model** revealed by the **design** used to detect it…

*Computer Science*: In formal verification one can only assert properties expressed in the modelling semantics. A **fault** means that the **model** fails to satisfy a given **specification**, and nothing else!

- Distinguish at all times the model and the real world
- Don’t fall in love with your model
- Don’t eat the menu!
- You will never strike oil by drilling through the map!

A sophisticated model is often not needed for fault management design, what is needed is, say, a “sophistically simple” model in terms of useful informational content to satisfy specifications.

This fits also into the Popper’s philosophy of science and his equation of simplicity with falsifiability … (Karl Popper, 1902-1994)
**Robust Yet Fragile (RYF)**

- **Robustness** is the capacity to preserve a certain property, or specification, in the presence of internal (components) and external (environment) uncertainty.

- **Fragility**

If you create robustness somewhere, you will create fragility somewhere else (*John Doyle, Caltech*).

Any model-based fault management labeled “Robust” is “RYF”!
How a robust design can become fragile

**Question (President):** My question is simple: How did we go so wrong?

**Answer (Alan Greenspan):** ... I discovered a flaw in the models, as we currently employ them, that I perceived is the critical functioning structure that defines how the world works ...

Alan Greenspan, Chairman of the Federal Reserve of the United States from 1987 to 2006.

---

We went wrong because models, as we currently employ them, are not able to capture the full array of governing variables in extreme situations that drive global economic reality... 
**Suddenly, robust models/designs that were working quite well since 20 years, failed...**
Across the aerospace and aviation engineering community, the development of fault management capabilities has been more of an “art” than a “science”: fault management systems are designed mostly through ad-hoc rules to avoid known problems, then extensively tested through simulations to help identify unexpected problems.

Fault management algorithms actually deployed in flight-critical systems are relatively simple …

A major barrier (and cost factor) for a new design is the certification requirements, particularly if the new design is structurally different to the in-service solutions.
**Fault monitoring**: cross checks, consistency checks, voting mechanisms, and Built-In Test techniques of varying sophistication

**Recovery**: hardware/software reconfiguration: hot or cold redundancy
Many examples where fault protection responded appropriately to transit behavior that was unexpected:

- **Galileo (1990-1995):** Despun Power Bus caused by debris shorts
- **Magellan (1990-1992):** Software flaw that caused heartbeat termination
- **Cassini (1993):** Attitude estimator transit during backup Star Tracker checkout
- **MER Spirit Rover (2005):** Potato-sized rock jammed in right rear wheel
- **Dawn (2008):** Cosmic ray upset of attitude control electronics
- **Kepler (2009):** Undervoltage due to unexpected power interactions at launch
Current state of practice in aeronautics

- Flight conditions
- Based on standby redundancy

**FCC**

Monitored Signals

> Confirmation Time

Fault detection

**System reconfiguration**

Flight conditions-based thresholds
Example: Sensor Fault Tolerance and Air Data and Inertial Reference System (ADIRS)
Advanced academic model-based FDIR

The field may sometimes appear as a collection of disparate topics, tricks and modifications to the earlier works…
Model-based FDI

- System model
- Uncertainty model
- Fault model
- Measurements
- Monitoring specifications

Design:
Monitoring algorithm

Design:
Decision making

Residuals / estimated faults

Alarms

- Structured failure detection filter
- Parity space projection
- Dedicated Observer, Unknown Input Observer …
- Parameter estimation approaches
- Data-based, signal-based methods…
- Stochastic methods (probability estimation for diagnosis…)
- Geometric methods
- Frequency domain methods ($H_\infty / H_\omega$, min-max optimization, …)
- Self-organizing and self-learning maps, machine learning
- Non-linear diagnosis observers
- LPV design, LMI formulation, multiple-model strategies
- Sliding Mode schemes
- Set membership techniques
- …

Early 1970s

Fixed/Adaptive thresholds; statistical tests…

Today
Example: $H_\infty / H_\cdot$ design and Generalized $\mu$

Design (mini-max / muti-objectives)

Analysis (diagnosis-oriented, generalized $\mu$)
Model-based Recovery

- Very active area since more than two decades: Control allocation issues, fault compensability, Vehicle stabilization following failures and damage, Reconfigurable and fault tolerant flight control …

Emergent topics:

- Loss of Control Inflight (LOC-I) issues; upset prevention and recovery

- **Mainly in the USA** : Researches at NASA Ames and Langley with top US researchers and universities >> many AIAA publications (American Institute of Aeronautics and Astronautics) publicly available at: [https://ntrs.nasa.gov/](https://ntrs.nasa.gov/)

- Europe is behind the schedule …
In the US: NASA programs (NASA Ames and NASA Langley): Advanced Air Vehicles Program (AAVP), Airspace Operations and Safety Program (AOSP) and Integrated Aviation Systems Program (IASP), NASA Aviation Safety Programs (AvSP) …

In Europe: Smart FDIR programs at the European Space Agency (ESA); European projects (FP7 and H2020): ADDSAFE, RECONFIGURE, VISION …

And many other investigations around the world …
To sum up

The **theory** is putting continually on the market a huge number of various designs, techniques and methods related to fault management (academic overproduction!)

Moreover, many successful aerospace **demonstrations** exist.

**Aerospace and aviation industry** provide numerous grounds where advanced fault management is needed to support conventional industrial practices.

However, today, few **real applications** can be identified…

**Why is this ?**
TRL scale

Technology Readiness Level

- TRL 9: System Test, Launch & Operations
- TRL 8: System/Subsystem Development
- TRL 7: Technology Demonstration
- TRL 6: Technology Development
- TRL 5: Research to Prove Feasibility
- TRL 4: Basic Technology Research
- TRL 3: TRL 2: TRL 1:

Death Valley

Industry

Academic Research

Money

Knowledge

August 29, 2019
Practical design considerations (output $W_1$)

- Practical design considerations
- Trimmable regions
- Determinism
- Complexity of the design
- Available on board control resources
- Poor excitation
- Fault detectability and model observability
- The number of input parameters
- Clear procedure for step-by-step tuning of the design
- The number of input parameters
- Implementation and computational time
- Modularity; Capacity of adaptation and genericity
- Modularity; Capacity of adaptation and genericity
- Unusual and non-standard flight situations
- Initializations
- Scalability
- Robustness & Fragility
- Fault compensability
- Initializations
- Clear procedure for step-by-step tuning of the design
- Implementation and computational time
- Modularity; Capacity of adaptation and genericity
- Unusual and non-standard flight situations
- Initializations
- Scalability
- Robustness & Fragility
- Fault compensability
A primary requirement for certification of commercial aerospace systems is that the systems operate *deterministically*: given a set of inputs you must always get the same result ...

Aerospace engineers love *deterministic* models providing *deterministic* designs

**Real world is not that deterministic:**
An aircraft is a highly complex interconnected hybrid system combining physical dynamics with computational processes. It has multiple behavioral modes interacting with each other that can change according to the operational conditions, external environment and pilot input...

Main issue: how to reconcile deterministic models / designs with a nondeterministic world and to get an as deterministic (predictable) as possible behavior...
The **sizing element** is the achievable performance and robustness not in nominal situations, but in “off-nominal” flight regimes: extreme, unusual and non-standard flight conditions.

Good “average” performance is necessary, but not sufficient at all!

**Unexpected**: High Angle of Attack

**Unexpected**: Bad sensor data
Rendez-vous phase, MSR Mission
(NASA | ESA)
A success story:
Model-based fault management design for new generation Airbus A350 aircraft
A successful example

Two adjacent dissimilar actuators for moving a single control surface on the **A380**

On the left: an EHA, and on the right a conventional hydraulic actuator.
The story began with basic research & lab investigations

Evaluations on Aircraft Airbus models; ground testbed platforms; in-flight tests…

Certification to fly, Entry into service and commercial flight (15-01-2015)
## A successful example (A350)

<table>
<thead>
<tr>
<th>Event</th>
<th>Details</th>
<th>TRL</th>
</tr>
</thead>
<tbody>
<tr>
<td>January 15, 2015</td>
<td>First commercial flight of A350 XWB (Doha–Frankfort, Qatar Airways)</td>
<td></td>
</tr>
<tr>
<td>September 30, 2014</td>
<td>Certification, European authorities (EASA)</td>
<td>9</td>
</tr>
<tr>
<td>November 12, 2014</td>
<td>Certification, US authorities (FAA)</td>
<td></td>
</tr>
<tr>
<td>Flight V&amp;V: A380</td>
<td>Flight tests, adaptation to A350, tuning / initializations, extreme flight conditions,</td>
<td>6–8</td>
</tr>
<tr>
<td>Ground V&amp;V: Evaluation</td>
<td>final implementation in FCC ...</td>
<td></td>
</tr>
<tr>
<td></td>
<td>on Airbus testbed platforms, actuator bench, System Integration Bench...</td>
<td>4–5</td>
</tr>
<tr>
<td>Ideas and concepts</td>
<td>Ideas and concepts, design and theoretical developments, publications/patents, high</td>
<td>1–3</td>
</tr>
<tr>
<td></td>
<td>fidelity simulations, tuning, proof of concept ...</td>
<td></td>
</tr>
</tbody>
</table>

≈ 8 years

These are \bibitem{A350} successful examples of aircraft development and implementation.
Another example

An adaptive model-based system for early and robust detection of abnormal positions of aircraft control surfaces


Example: Civil aviation operations
A modern aircraft as A350 offers on-board cockpit technologies a pilot 2-3 decades ago could only dream about.

But, what will next-generation aircraft and air transport look like in the next decades?
Starting in the 1950s, commercial aviation has been experiencing de-crewed operations.

Today, the vector is pointed towards more **autonomy** and **intelligence** in the cockpit.
Challenges and barriers for moving to SPO

SPO are associated with some issues that make incremental evolution of existing systems to SPO a tall order:

- Safety barriers
- Social acceptability
- Pilot incapacitation
- Desirable attributes for selection of the single pilot
- Ground-based support and air-ground collaboration,
- Cyberattacks or malevolent intents.
- …
Safety barriers for moving to SPO

- **Root problem:** The coordinated crew will not be available as a resource. Thus, during a profusion of complex situations, the single pilot in command will be ultimately responsible for the aircraft safety.

- Enabling technologies for automation augmentation, with automation replacing the second pilot? However, some issues in putting too much faith in automation:
  
  - How to manage mode confusion and automation surprises in SPO?
  - How to manage developing (or already developed) LOC-I situations in SPO?
  - From a regulatory and certification perspective: it may be difficult to design “automated systems” to achieve the level of safety in SPO as in today dual-pilot configuration.

Advanced fault management methods will be required to accommodate emerging functional requirements to solve safety puzzle in SPO.
Final words

- Today, with the current airworthiness regulatory framework, there is not really much space left for novelty model-based FDIR designs, both in aerospace and aviation… Yet, they may be still useful to complement specific local solutions to support (not to rule out) the current state-of-practice (incremental evolution)

- Tomorrow: the challenges in aerospace are far greater as individual systems evolve and operate with greater autonomy and intelligence within a connected and distributed flight environment

Research in model-based fault management for future aerospace programs has still some beautiful days ahead, provided that traditional questions are reopened and reinvestigated from this perspective

End of the old story and beginning of a new story!
That’s all

Thank You

Some more details in: