Model-Based Fault Management for Aerospace Systems: A Look Backwards and Forwards

Ali Zolghadri

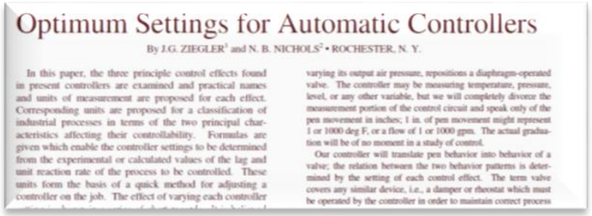
Bordeaux University - CNRS | IMS LAB

France

Do we provide fault management methodologies for practicing aerospace engineers ?

Do we provide a control system methodology for practicing engineers ?

(Ziegler and Nichols, 1942)



Optimum Settings for Automatic Controllers

By J.G. ZIEGLER[1] and N. B. NICHOLS[2] • ROCHESTER, N. Y.

In this paper, the three principle control effects found in present controllers are examined and practical names and units of measurement are proposed for each effect. Corresponding units are proposed for a classification of industrial processes in terms of the two principal characteristics affecting their controllability. Formulas are given which enable the controller settings to be determined from the experimental or calculated values of the lag and unit reaction rate of the process to be controlled. These units form the basis of a quick method for adjusting a controller on the job. The effect of varying each controller

varying its output air pressure, repositions a diaphragm-operated valve. The controller may be measuring temperature, pressure, level, or any other variable, but we will completely divorce the measurement portion of the control circuit and speak only of the pen movement in inches; 1 in. of pen movement might represent 1 or 1000 deg F, or a flow of 1 or 1000 gpm. The actual graduation will be of no moment in a study of control.

Our controller will translate pen behavior into behavior of a valve; the relation between the two behavior patterns is determined by the setting of each control effect. The term valve covers any similar device, i.e., a damper or rheostat which must be operated by the controller in order to maintain correct process

Outline

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 $0 < \alpha < 1; \alpha \in R$

What not

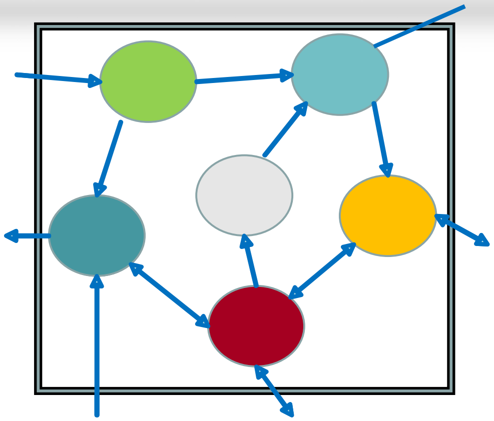
- New “Results” and “Theorems”

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…

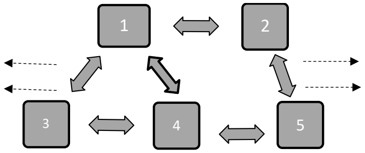
My current research interests (1)

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

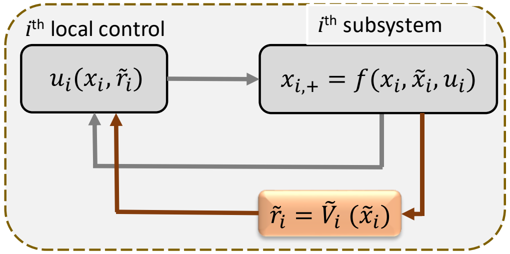


$$
\begin{aligned}
x_1^+ &\in f_1(x_1, x_2, x_3, \ldots x_n, u_1)\\
x_2^+ &\in f_2(x_2, x_1, x_3, \ldots x_n, u_2)\\
&\vdots\\
x_n^+ &\in f_n(x_n, x_1, x_2, \ldots x_{n-1}, u_n)
\end{aligned}
$$

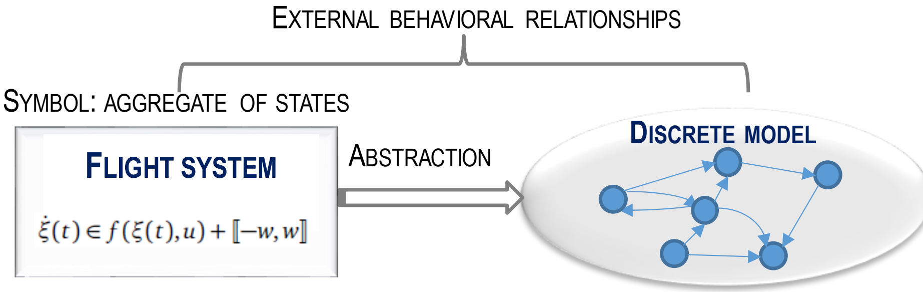
$x_i \in X_i,\ u_i \in U_i,\ i \in I$



i^{th} controlled flight system


i^{th} local control
i^{th} subsystem
$u_i(x_i, \tilde{r}_i)$
$x_{i,+} = f(x_i, \tilde{x}_i, u_i)$
$\tilde{r}_i = \tilde{V}_i\,(\tilde{x}_i)$

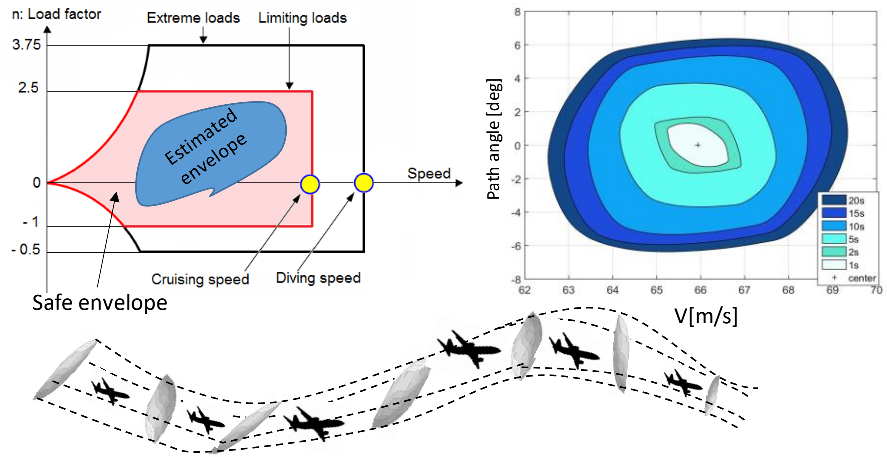
Partial information about the states of neighboring systems through ranking functions $\tilde{V}_i$.

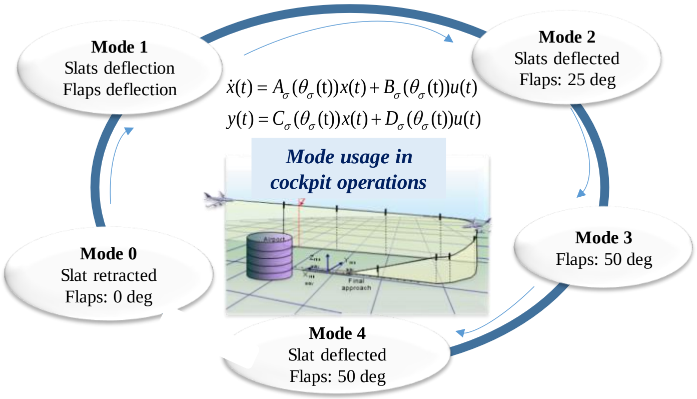

EXTERNAL BEHAVIORAL RELATIONSHIPS
SYMBOL: AGGREGATE OF STATES
FLIGHT SYSTEM
ABSTRACTION
DISCRETE MODEL

$\dot{\xi}(t) \in f(\xi(t), u) + [\![-w, w]\!]$

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…**





Mode 1
Slats deflection
Flaps deflection

Mode 2
Slats deflected
Flaps: 25 deg

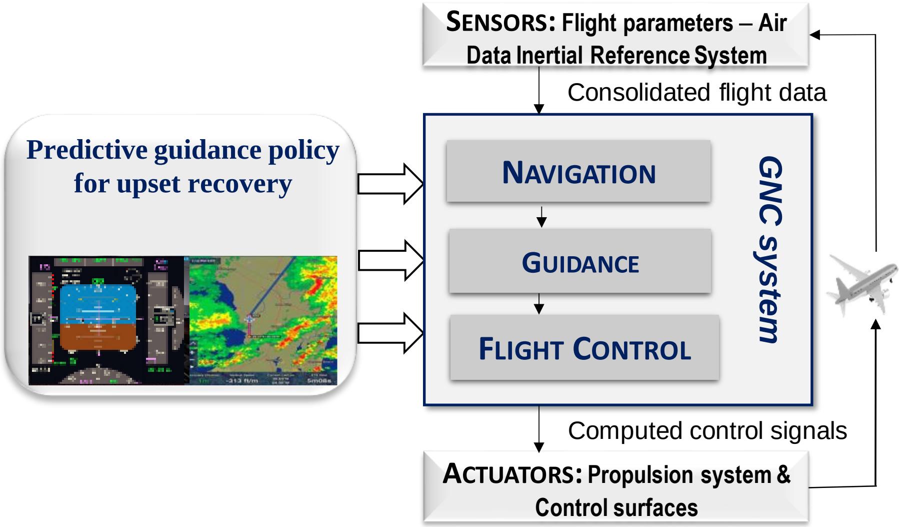
Mode 3
Flaps: 50 deg

Mode 4
Slat deflected
Flaps: 50 deg

Mode 0
Slat retracted
Flaps: 0 deg

$$\dot{x}(t) = A_\sigma(\theta_\sigma(t))x(t) + B_\sigma(\theta_\sigma(t))u(t)$$
$$y(t) = C_\sigma(\theta_\sigma(t))x(t) + D_\sigma(\theta_\sigma(t))u(t)$$

Mode usage in cockpit operations



SENSORS: Flight parameters – Air Data Inertial Reference System

Consolidated flight data

Predictive guidance policy for upset recovery

GNC system

NAVIGATION

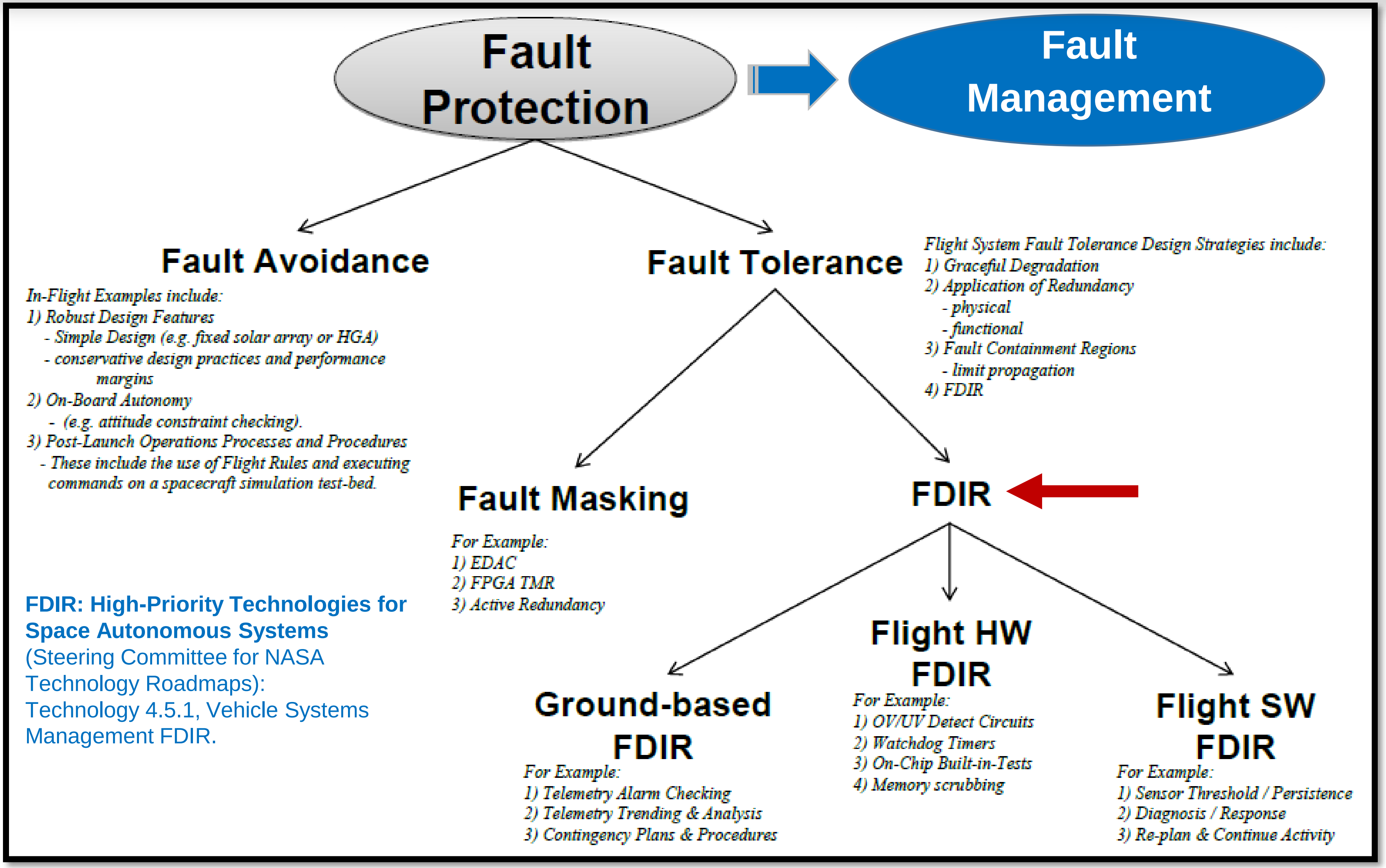
GUIDANCE

FLIGHT CONTROL

Computed control signals

ACTUATORS: Propulsion system & Control surfaces

Basic terminology (source: NASA - JPL)



Fault Protection

Fault Management

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

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

Fault Masking

For Example:
1) EDAC
2) FPGA TMR
3) Active Redundancy

FDIR

Ground-based FDIR

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

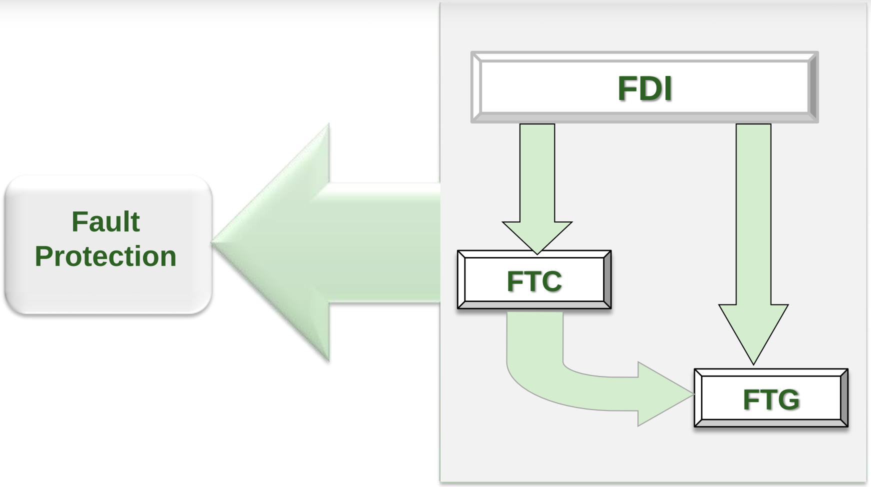
Flight HW FDIR

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

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

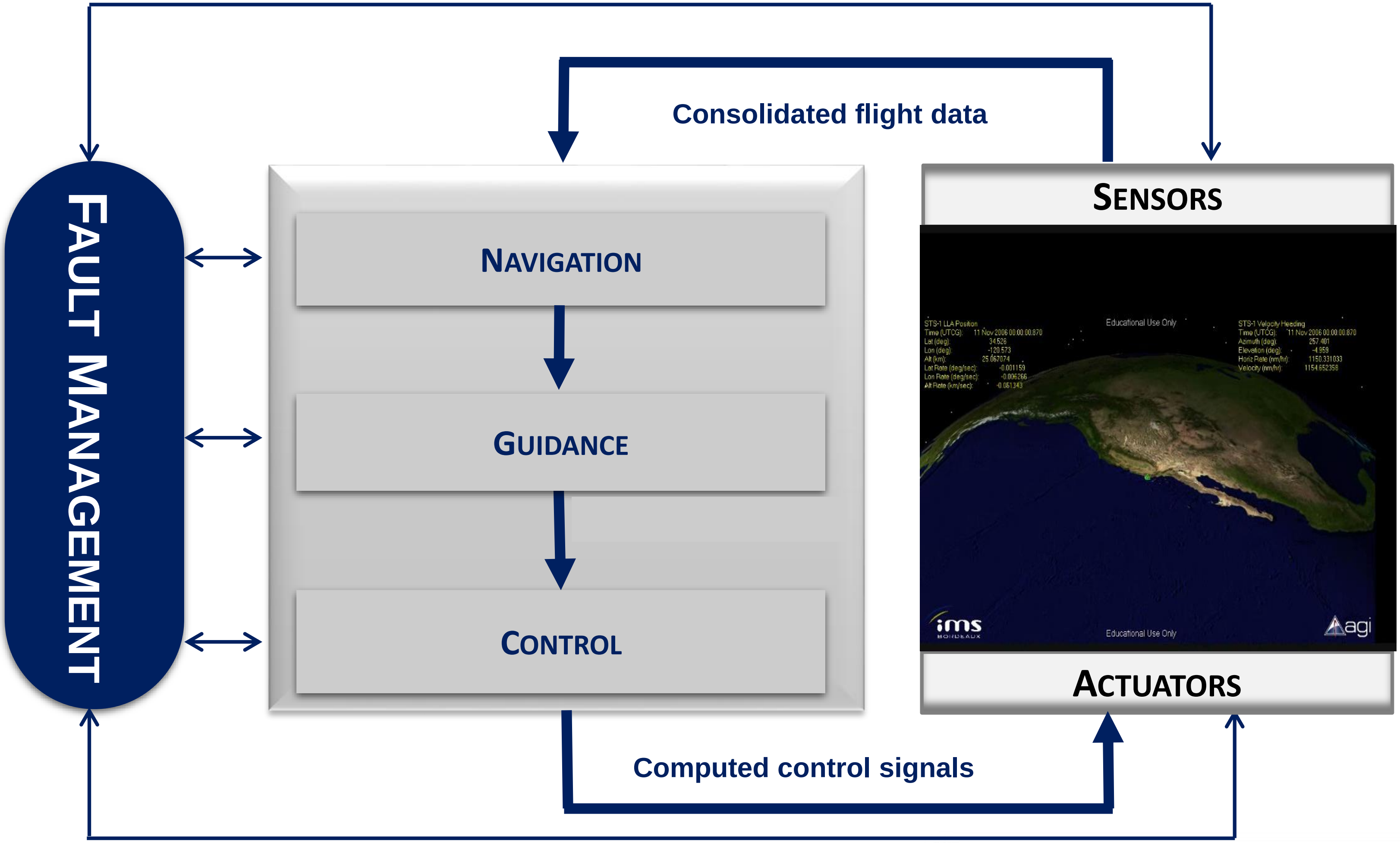


$$\mathrm{FDIR} = \mathrm{FDI} \,\&\, \underbrace{\mathrm{FTC} / \mathrm{FTG}}_{\mathrm{Recovery}}$$

- **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

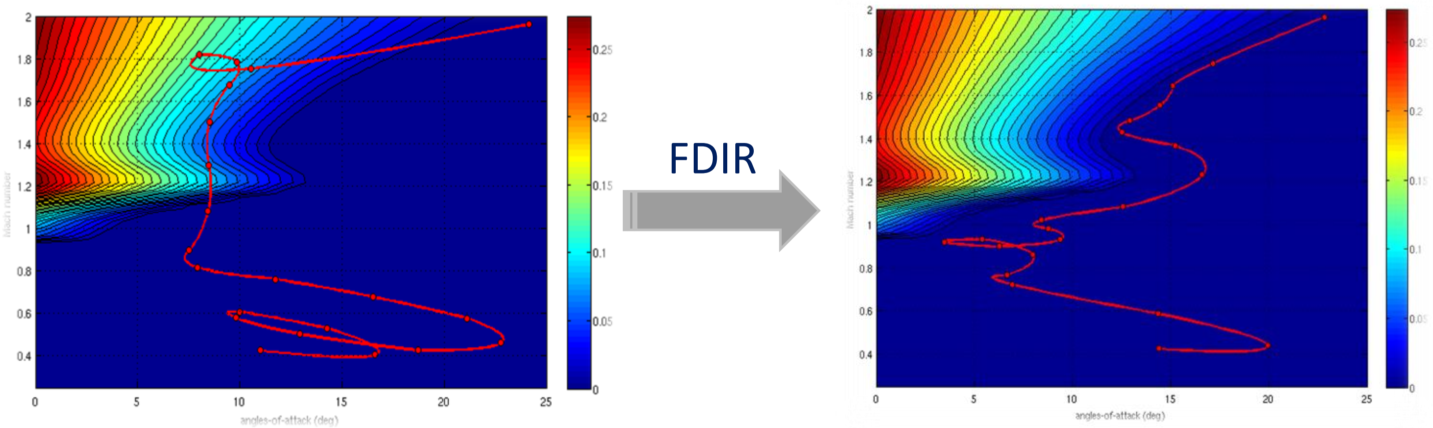


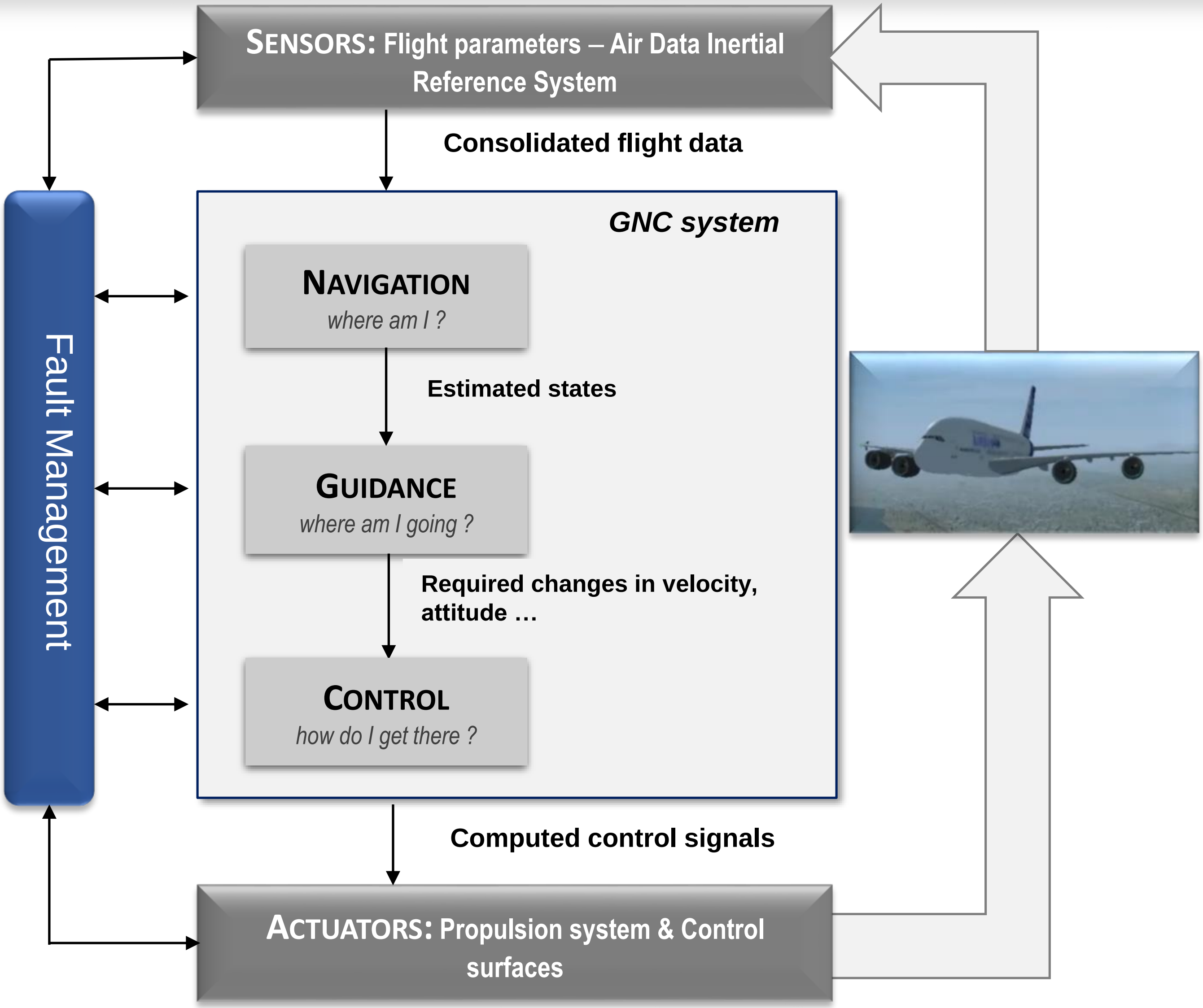
STS-1 LLA Position
Time (UTCG): 11 Nov 2006 00:00:00.870
Lat (deg): 34.526
Lon (deg): -120.573
Alt (km): 25.067074
Lat Rate (deg/sec): -0.001159
Lon Rate (deg/sec): -0.006266
Alt Rate (km/sec): -0.051343

STS-1 Velocity Heading
Time (UTCG): 11 Nov 2006 00:00:00.870
Azimuth (deg): 257.401
Elevation (deg): -4.959
Horiz Rate (nm/hr): 1150.331033
Velocity (nm/hr): 1154.652358

Shuttle mission STS-86

J.L. Chrétien
Astronaut
NASA | ESA





SENSORS: Flight parameters – Air Data Inertial Reference System

Consolidated flight data

GNC system

NAVIGATION
where am I ?

Estimated states

GUIDANCE
where am I going ?

Required changes in velocity, attitude …

CONTROL
how do I get there ?

Computed control signals

ACTUATORS: Propulsion system & Control surfaces

Fault Management

“Application”

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 !

Model vs. Reality

Solomon Golomb (1932-2016)
“Mathematical models – Uses and limitations”.
Aeronautical Journal, 1968.

- **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 !**

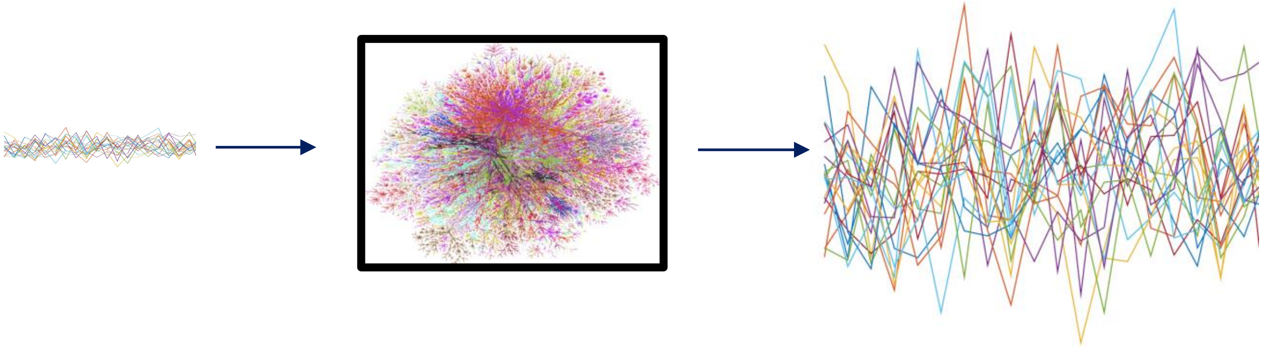
A **sophisticated** model is often not needed for fault management design, what is needed is, say, a **“sophistically simple”** model in terms of useful informational content to satisfy specifications.

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

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

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 …

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…

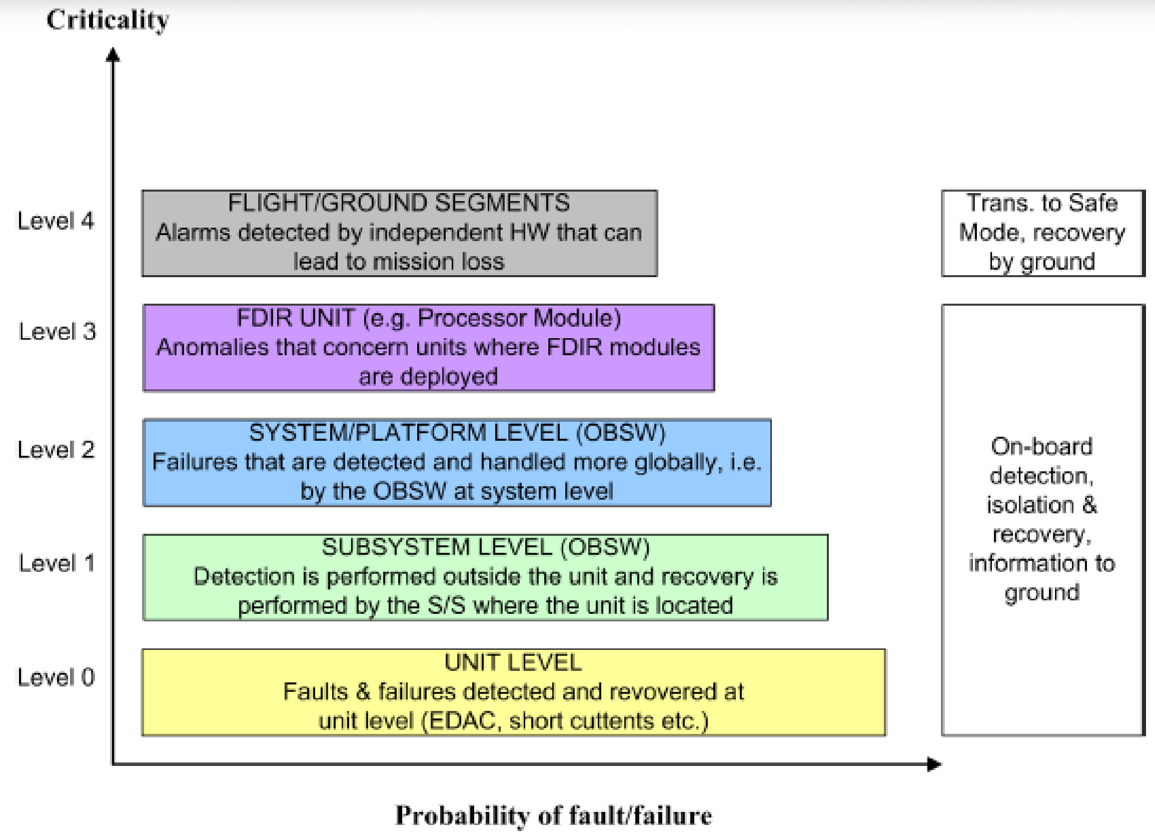
Conventional (in service) FDIR for aerospace systems

- ✓ 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.

Today State-of-Practice



Criticality

Level 4

FLIGHT/GROUND SEGMENTS
Alarms detected by independent HW that can lead to mission loss

Trans. to Safe Mode, recovery by ground

Level 3

FDIR UNIT (e.g. Processor Module)
Anomalies that concern units where FDIR modules are deployed

Level 2

SYSTEM/PLATFORM LEVEL (OBSW)
Failures that are detected and handled more globally, i.e. by the OBSW at system level

Level 1

SUBSYSTEM LEVEL (OBSW)
Detection is performed outside the unit and recovery is performed by the S/S where the unit is located

Level 0

UNIT LEVEL
Faults & failures detected and revovered at unit level (EDAC, short cuttents etc.)

On-board detection, isolation & recovery, information to ground

Probability of fault/failure

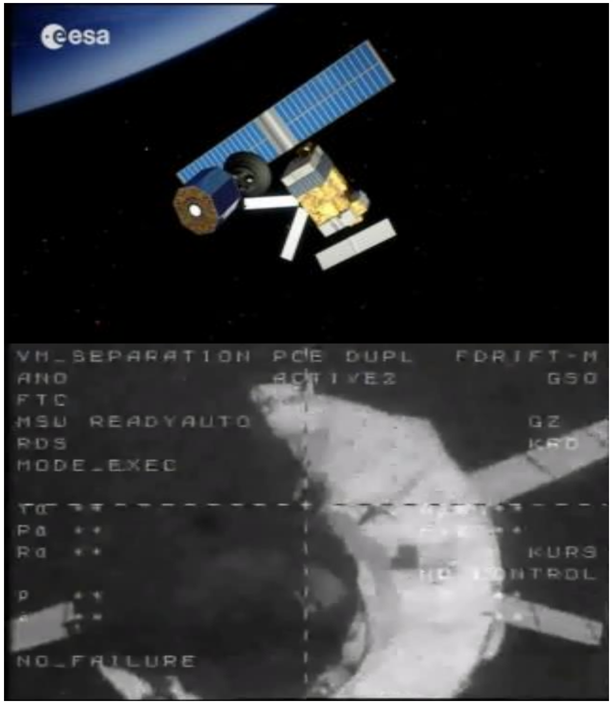
Fault monitoring: cross checks, consistency checks, voting mechanisms, and Built-In Test techniques of varying sophistication

Recovery: hardware/software reconfiguration: hot or cold redundancy

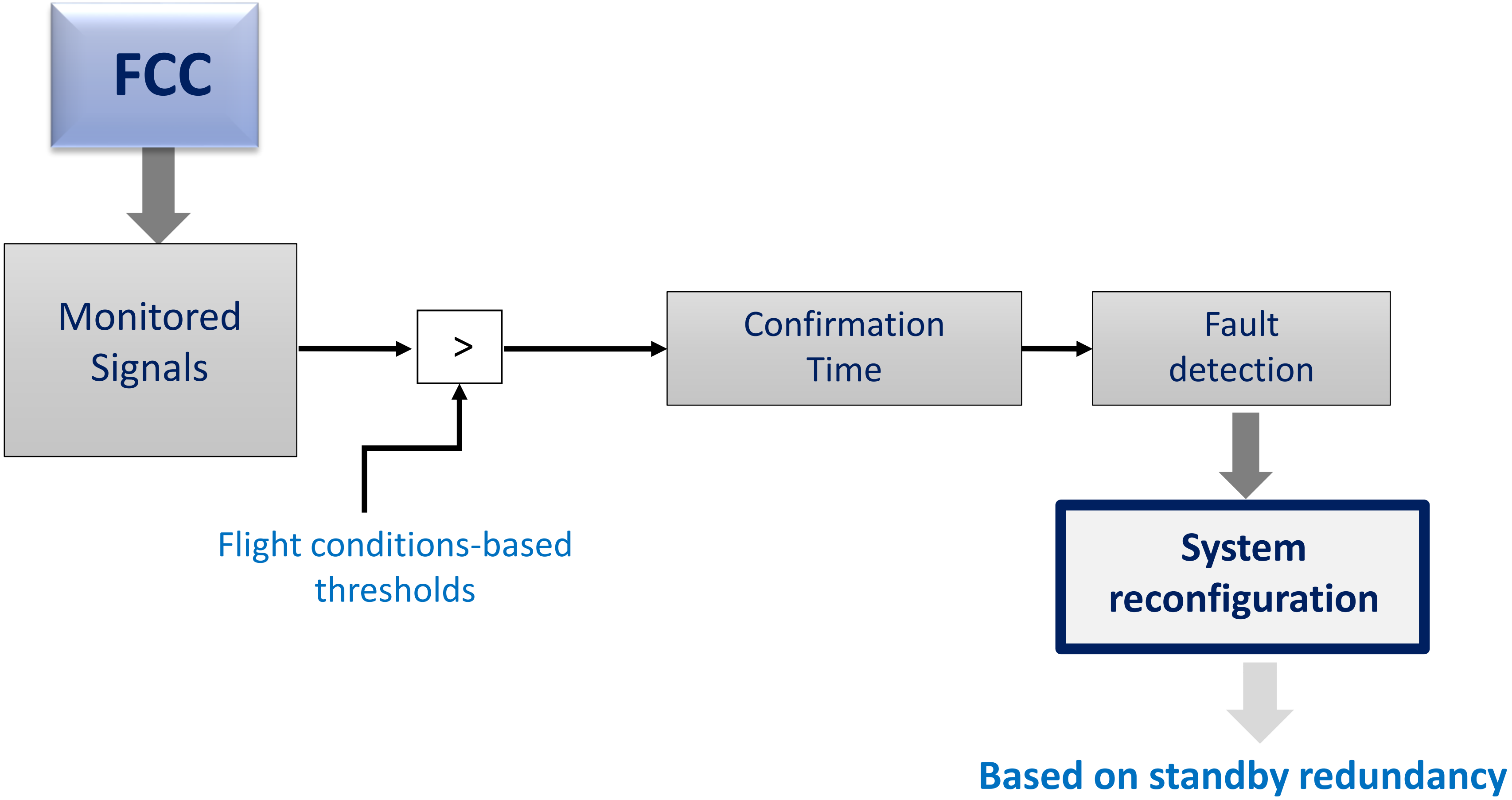
Against unexpected …

Many examples where fault protection responded appropriately to transit behavior that was <u>unexpected</u>:

- *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



FCC

Monitored Signals

>

Flight conditions-based thresholds

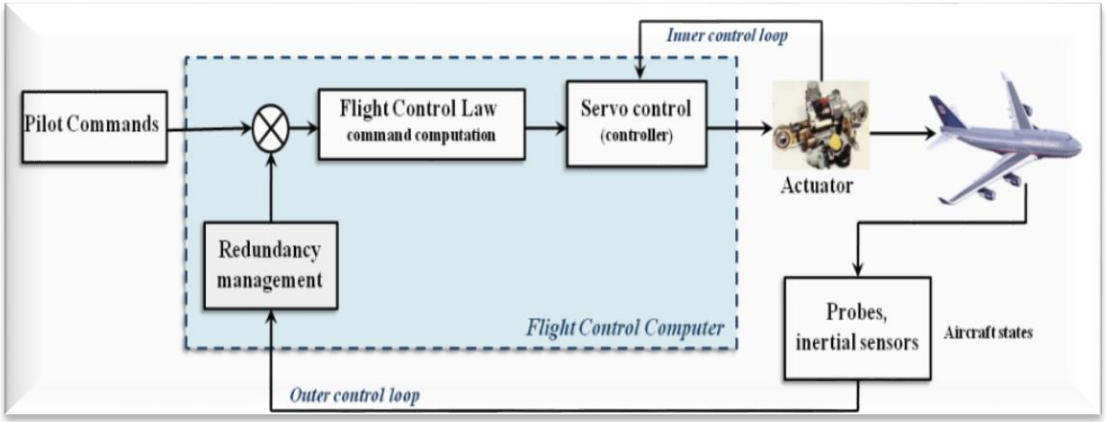
Confirmation Time

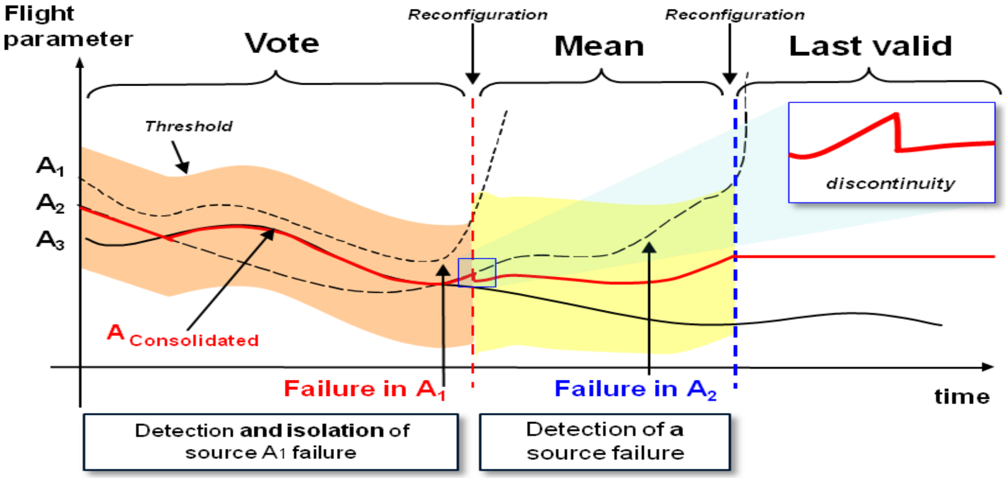
Fault detection

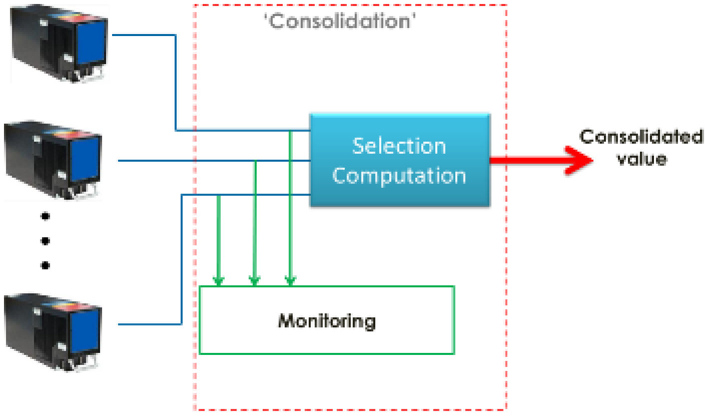
System reconfiguration

Based on standby redundancy

Example: Sensor Fault Tolerance and Air Data and Inertial Reference System (ADIRS)



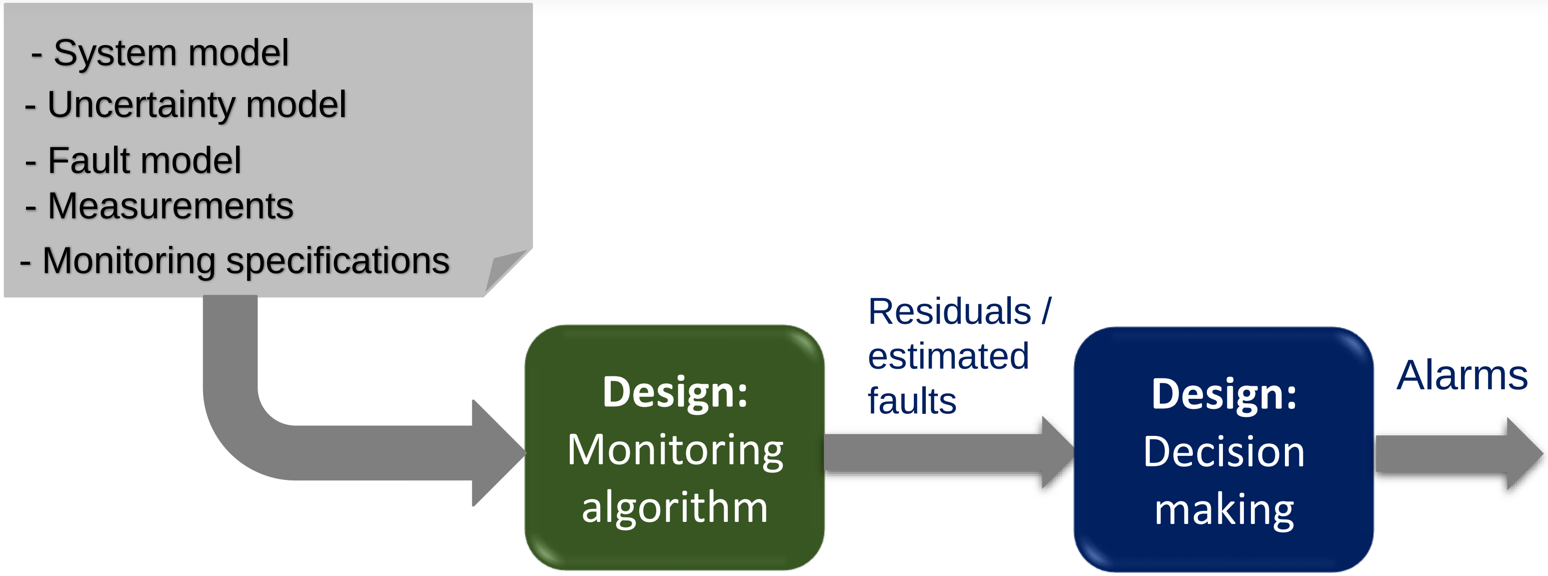




Advanced academic model-based FDIR

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

Residuals / estimated faults

Design: Decision making

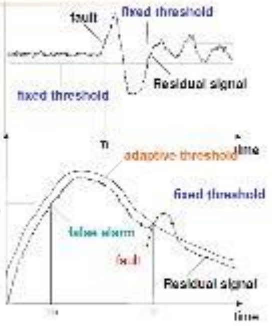
Alarms

- 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_∞ /H_-, 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

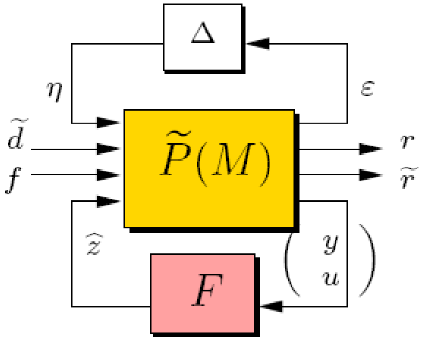
Today

Fixed/Adaptive thresholds; statistical tests…

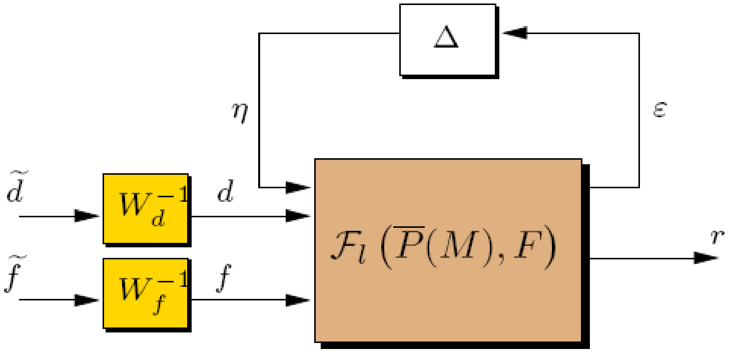


Example: H_∞ / H_- design and Generalized μ

Design (mini-max / muti-objectives)



Analysis (diagnosis-oriented, generalized μ)



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/

- **Europe** is behind the schedule …

And projects …

- 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!)

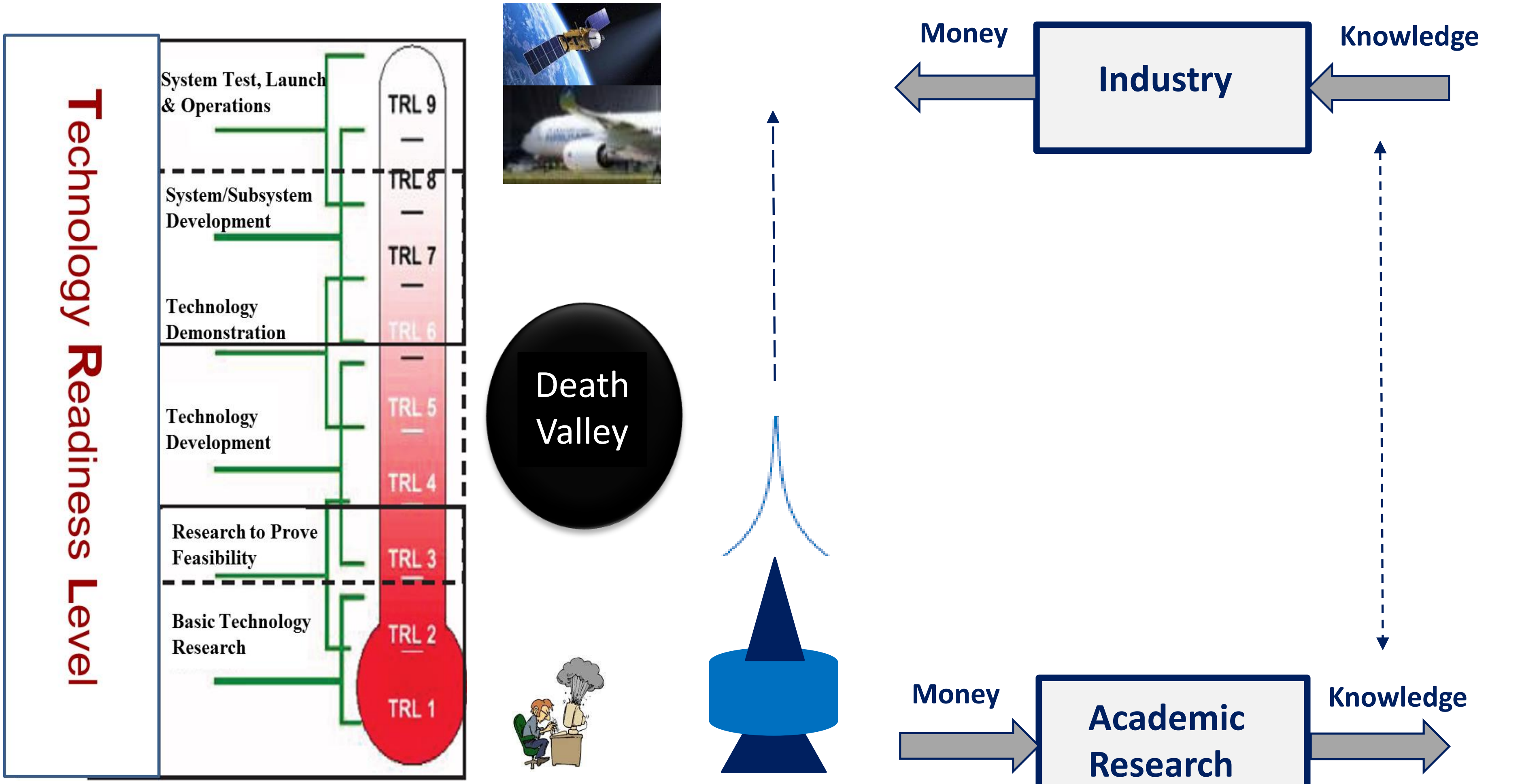
Moreover, many successful aerospace demonstrations exist.

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

System Test, Launch & Operations

System/Subsystem Development

Technology Demonstration

Technology Development

Research to Prove Feasibility

Basic Technology Research

TRL 9

TRL 8

TRL 7

TRL 6

TRL 5

TRL 4

TRL 3

TRL 2

TRL 1

Death Valley

Money

Industry

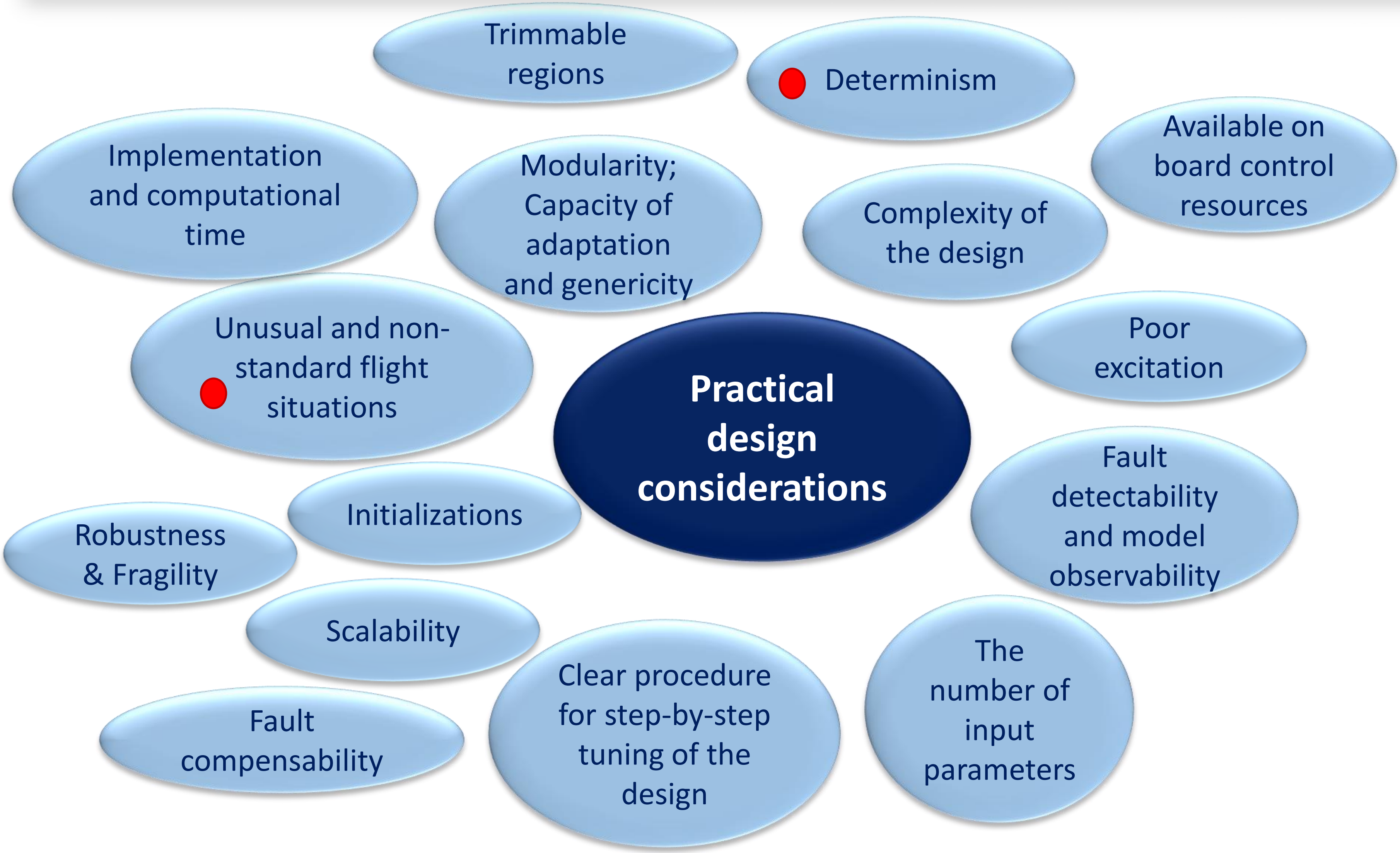
Knowledge

Money

Academic Research

Knowledge

Practical design considerations (output W_1)



Practical design considerations

Trimmable regions

● Determinism

Implementation and computational time

Modularity; Capacity of adaptation and genericity

Complexity of the design

Available on board control resources

● Unusual and non-standard flight situations

Poor excitation

Robustness & Fragility

Initializations

Fault detectability and model observability

Scalability

Fault compensability

Clear procedure for step-by-step tuning of the design

The number of input parameters

Determinism

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…

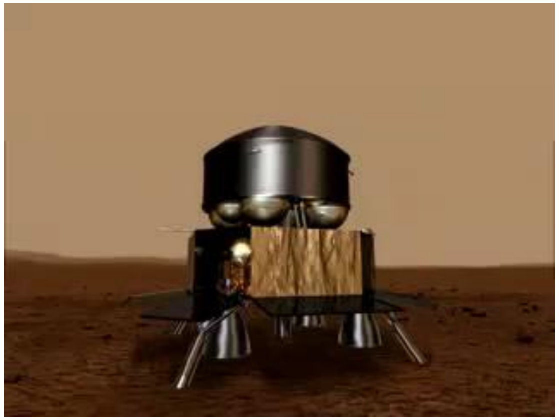
Main issue: how to reconcile deterministic models / designs with a nondeterministic world and to get an as deterministic (predictable) as possible behavior…

Unusual and extreme flight situations

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

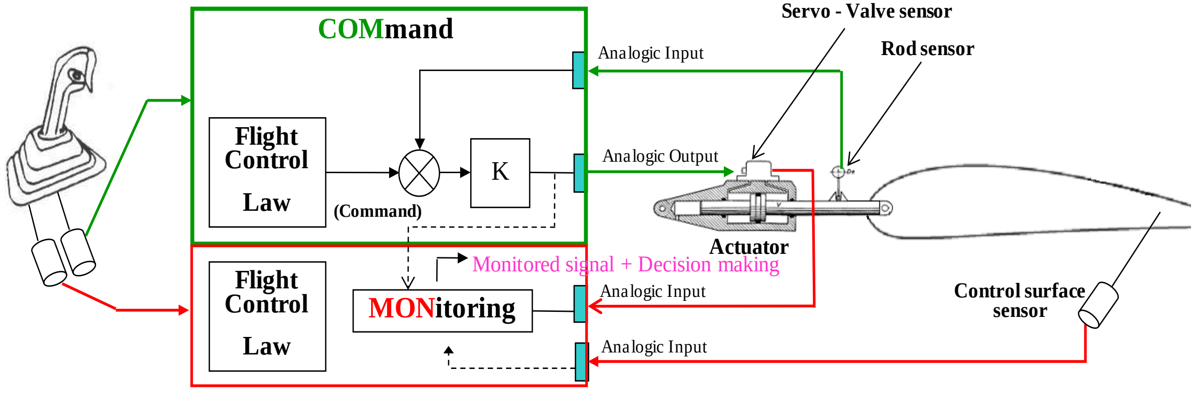
(NASA | ESA)

A successful example A350

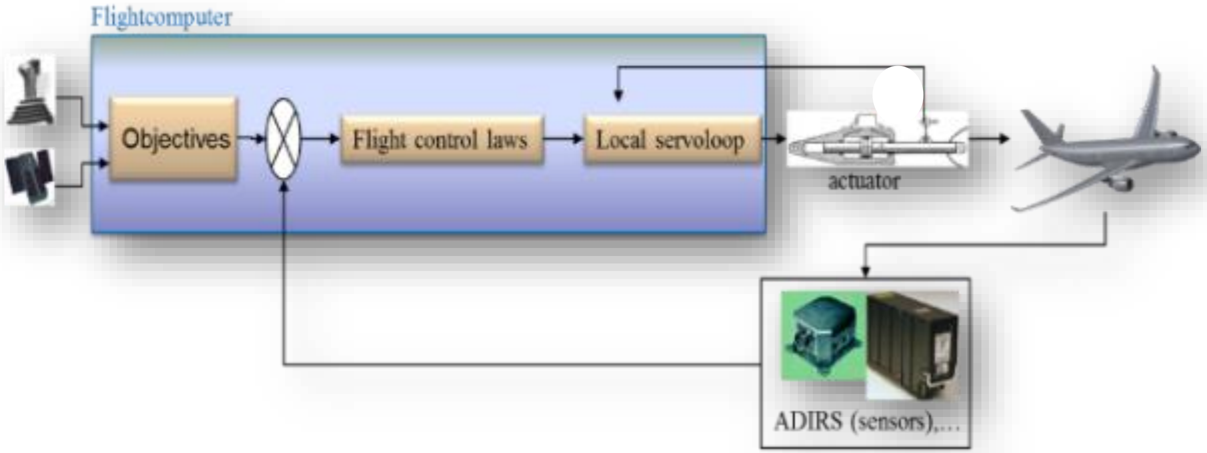
A success story:

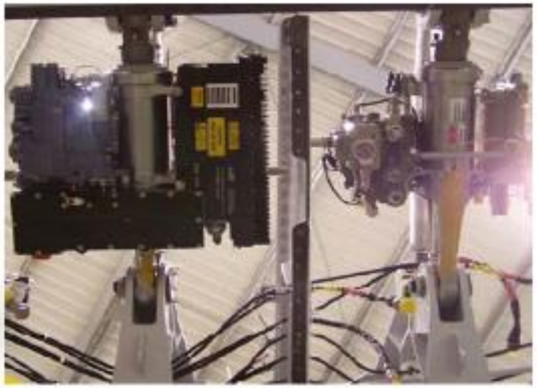
Model-based fault management design for new generation Airbus A350 aircraft

A successful example



Flight Control Computer (FCC)

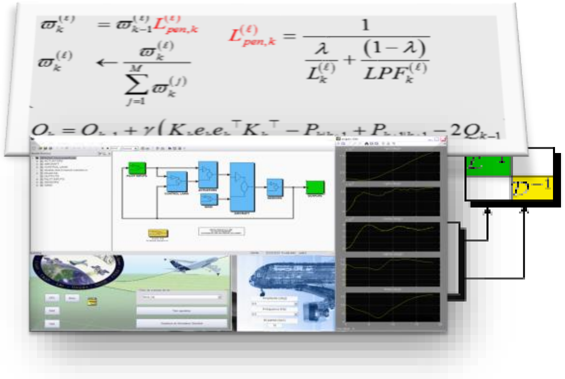




Two adjacent dissimilar actuators for moving a single control surface on the **A380**
On the left: an EHA, and on the right a conventional hydraulic actuator.

A successful example

The story began with basic research & lab investigations



$$\varpi_k^{(\ell)} = \varpi_{k-1}^{(\ell)} L_{pen,k}^{(\ell)} \qquad L_{pen,k}^{(\ell)} = \frac{1}{\frac{\lambda}{L_k^{(\ell)}} + \frac{(1-\lambda)}{LPF_k^{(\ell)}}}$$

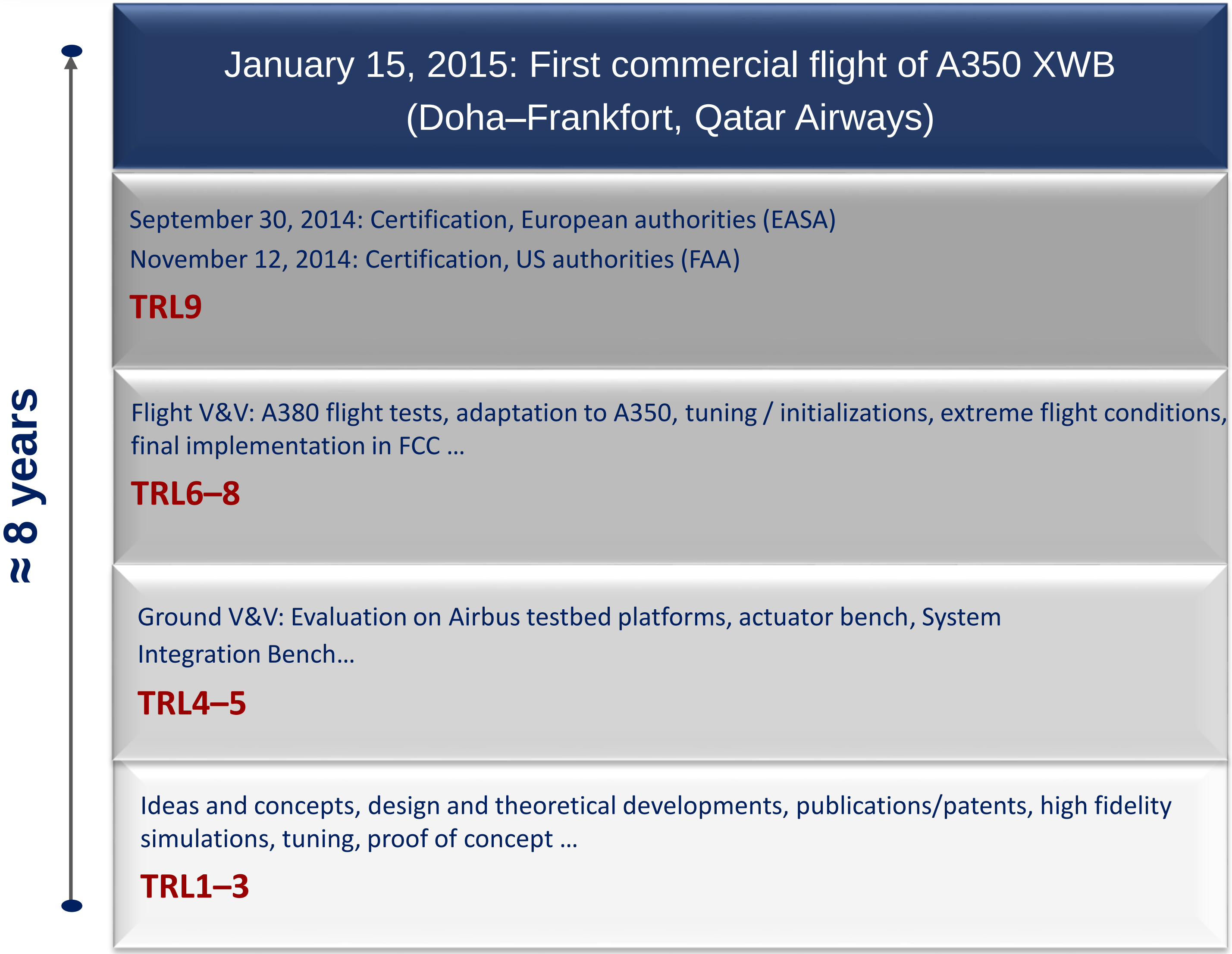
$$\varpi_k^{(\ell)} \leftarrow \frac{\varpi_k^{(\ell)}}{\sum_{j=1}^{M} \varpi_k^{(j)}}$$

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)



January 15, 2015: First commercial flight of A350 XWB (Doha–Frankfort, Qatar Airways)

September 30, 2014: Certification, European authorities (EASA)

November 12, 2014: Certification, US authorities (FAA)

TRL9

Flight V&V: A380 flight tests, adaptation to A350, tuning / initializations, extreme flight conditions, final implementation in FCC …

TRL6–8

Ground V&V: Evaluation on Airbus testbed platforms, actuator bench, System Integration Bench…

TRL4–5

Ideas and concepts, design and theoretical developments, publications/patents, high fidelity simulations, tuning, proof of concept …

TRL1–3

≈ 8 years

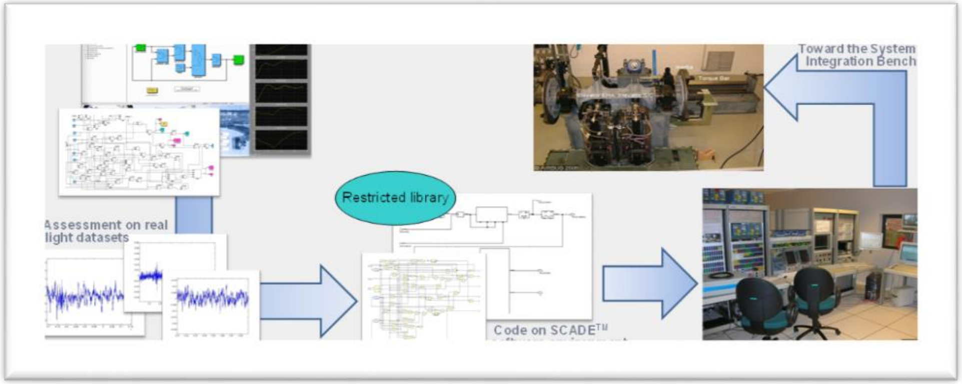
Another example

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

Initial theory: Zolghadri A. (1996). An algorithm for failure detection in Kalman filters. **IEEE Transactions on Automatic Control**.

Some experimental results: Zolghadri A, Goupil P., J. Cieslak, Dayre R. (2016). **Journal of Aircraft, American Institute of Aeronautics and Astronautics.**

TRL



Toward the System Integration Bench
Restricted library
Assessment on real light datasets
Code on SCADE™

A look forwards

Example: Civil aviation operations

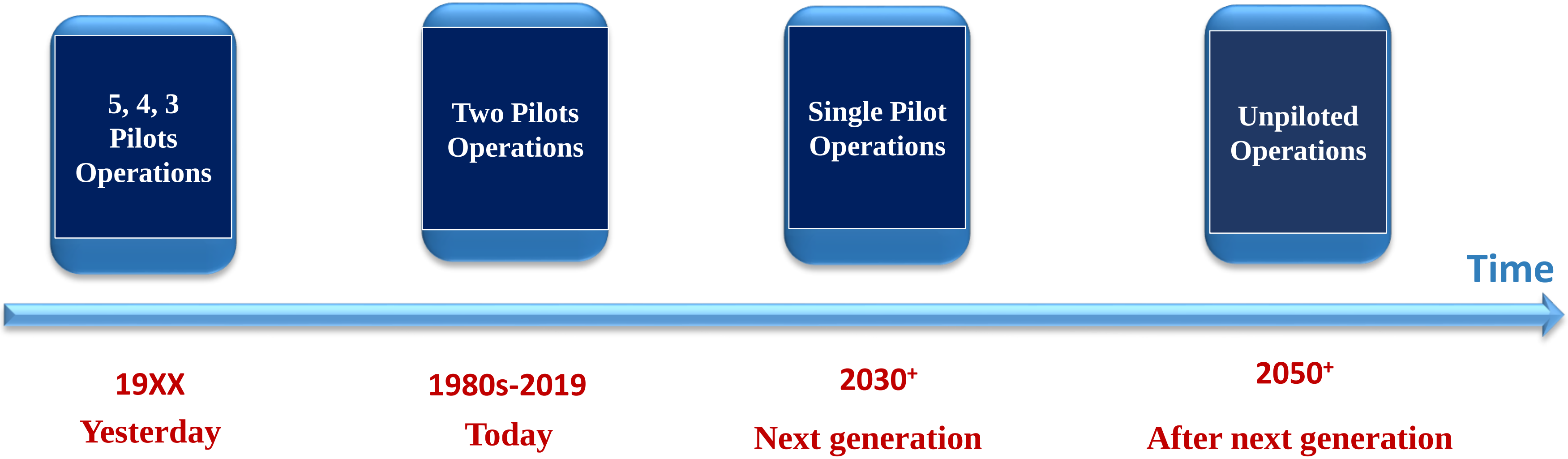
Civil aviation operations

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?

Trend in civil aviation operations

- Starting in the 1950s, commercial aviation has been experiencing de-crewing
- Today, the vector is pointed towards more autonomy and intelligence in the cockpit



5, 4, 3 Pilots Operations

Two Pilots Operations

Single Pilot Operations

Unpiloted Operations

Time

19XX

Yesterday

1980s-2019

Today

2030^+

Next generation

2050^+

After next generation

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:

A. Zolghadri (2018). On flight operational issues management: past, present and future. Annual reviews in control.