A System of Systems Approach for Improved Autonomy of Unmanned Systems

George J. Vachtsevanos

Professor Emeritus

School of Electrical and Computer Engineering

Georgia Institute of Technology, Atlanta Georgia

and

Kimon P. Valavanis

John Evans Professor

University of Denver

Topical Outline

What is Autonomy? What are Autonomous Systems?

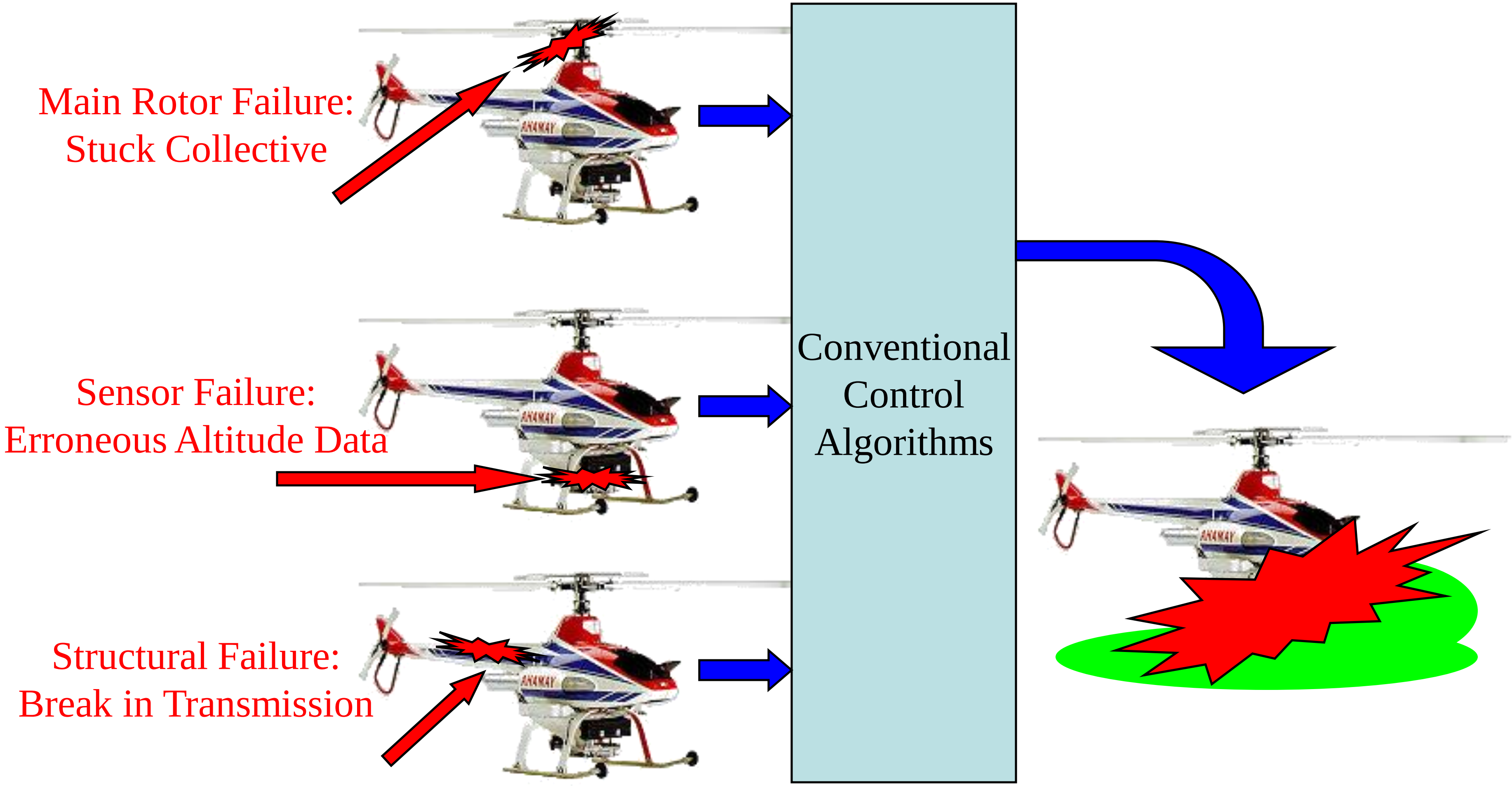
Three Pillars for ***Resilient Design and Operation of Autonomous Systems***:

Prognostics and Health Management

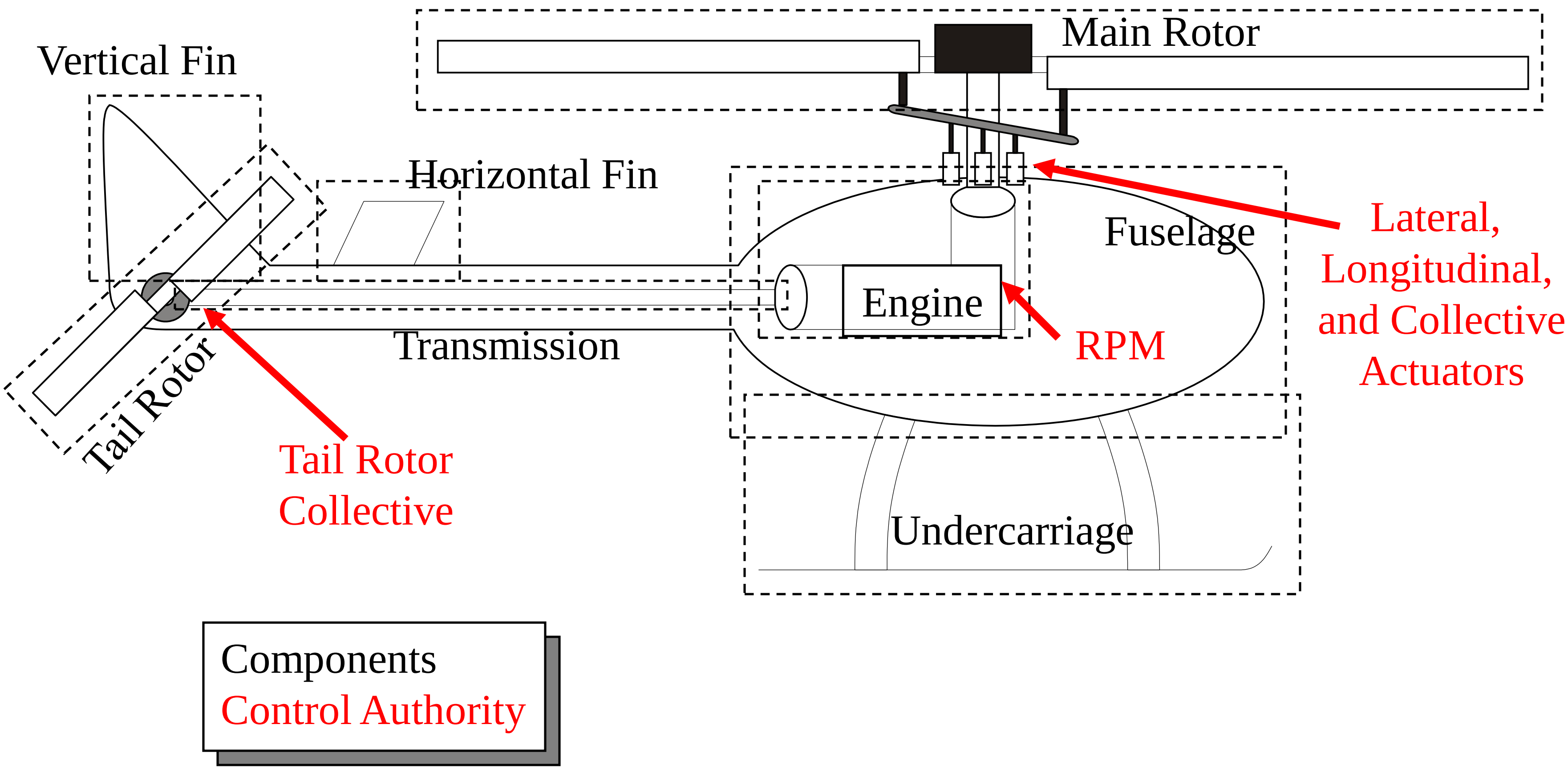
Resilient Design and Operation

Risk Analysis and Risk Control

Fault Tolerant Control: The Problem



Fault Tolerant Control: The Approach



Objectives of Fault Tolerant Control

To provide answers to the following:

- ❖ Did a failure occur?
- ❖ What failure(s) occurred?
- ❖ What is the impact of this failure on other (healthy) system components?
- ❖ How can we <u>restructure</u> the system so that it remains operational (even at a degraded mode)
- ❖ How can we <u>reconfigure</u> the controls so that the system remains stable and maintains some level of acceptable performance during the emergency?

Autonomous Control Level (ACL) Chart

Note: As ACL increases, capability includes, or replaces, items from lower levels

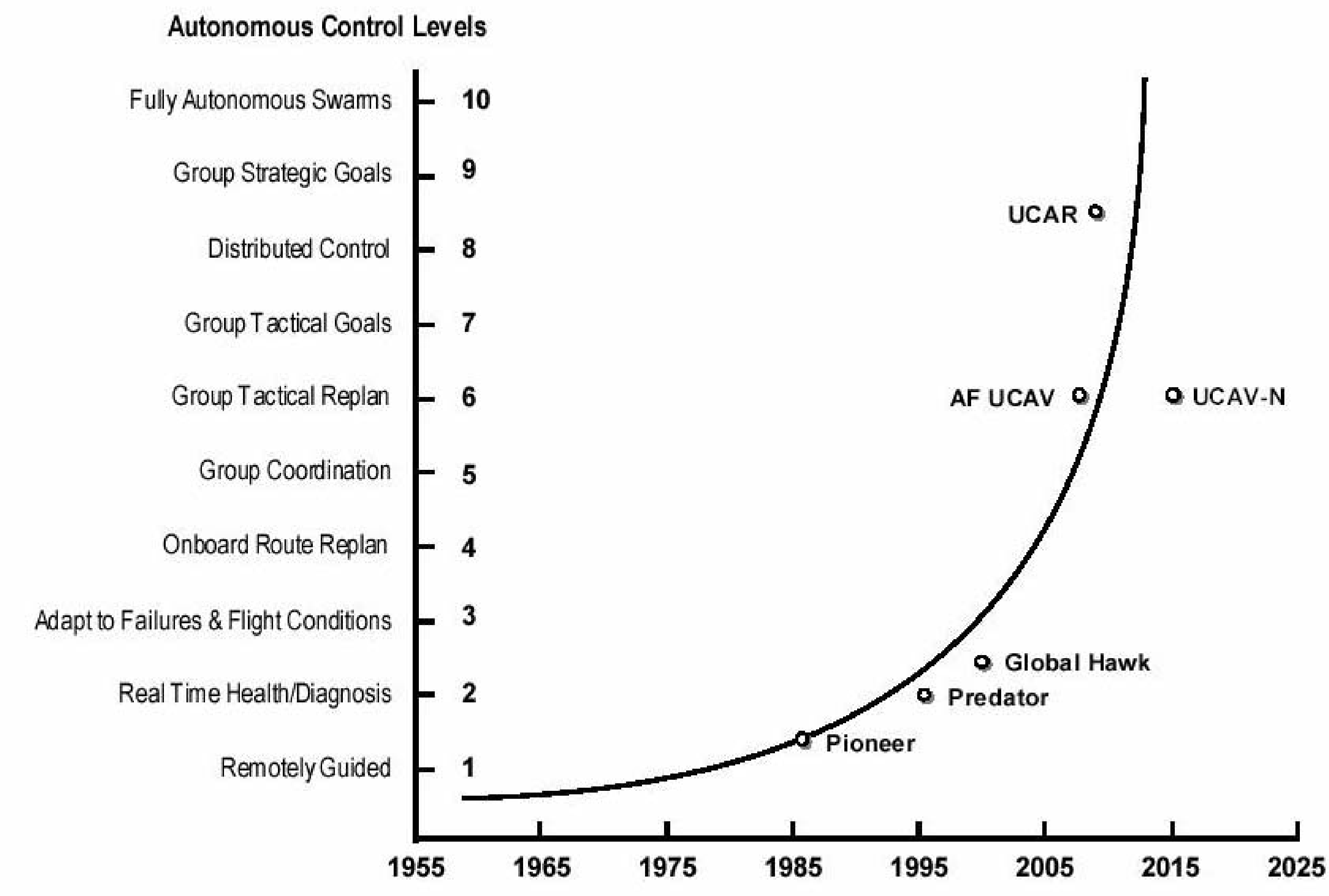
Level	Level Descriptor	Observe Perception / Situational Awareness	Orient Analysis / Coordination	Decide Decision Making	Act Capability
10	Fully Autonomous	Cognizant of all within Battlespace	Coordinates as necessary	Capable of total independence	Requires little guidance to do job
9	Battlespace Swarm Cognizance	Battlespace inference – Intent of self and others (allies and foes). Complex Intense environment – on-board tracking	Strategic group goals assigned. Enemy Strategy inferred.	Distributed tactical group planning. Individual determination of tactical goal. Individual task planning/execution. Choose tactical targets	Group accomplishment of strategic goal with no supervisory assistance
8	Battlespace Cognizance	Proximity inference – Intent of self and others (allies and foes). Reduced dependence upon off-board data.	Strategic group goals assigned Enemy tactics inferred ATR	Coordinated tactical group planning Individual task planning / execution Choose targets of opportunity	Group accomplishment of strategic goal with minimal supervisory assistance
7	Battlespace Knowledge	Short track awareness – History and predictive battlespace data in limited range, timeframe and numbers.	Tactical group goals assigned. Enemy Trajectory estimated.	Individual task planning / execution to meet goals	Group accomplishment of tactical goal with minimal supervisory assistance
6	Real-Time Multi-Vehicle Cooperation	Ranged awareness – Onboard sensing for long rang, supplemented by off-board data	Tactical group goals assigned Enemy location sensed / estimated	Coordinated trajectory planning and execution to meet goals – Group optimization	Group accomplishment of tactical goal with minimal supervisory assistance
5	Real-Time Multi-Vehicle Coordination	Sensed awareness – Local sensor to detect others, fused with off-board data	Tactical group plan assigned RT Health Diagnosis; Ability to compensate for most failures and flight conditions; Ability to predict onset of failures (e.g. Prognostic Health Mgmt) Group diagnosis and resource management	On-board trajectory replanning – Optimize for current and predictive conditions Collision avoidance	Group accomplishment of tactical plan as externally assigned Air collision avoidance Possible close air space separation for AAR, formation in non-thread conditions

Autonomous Control Level (ACL) Chart

Note: As ACL increases, capability includes, or replaces, items from lower levels

Level	Level Descriptor	Observe Perception / Situational Awareness	Orient Analysis / Coordination	Decide Decision Making	Act Capability
4	Fault / Event Adaptive Vehicle	Deliberate Awareness – Allies communicate data	Tactical plan assigned Assigned Rules of Engagement RT Health Diagnosis. - Ability to compensate for most failures and flight conditions – inner loop changes reflected in outer loop performance	On-board trajectory replanning – Event driven, Self resource management, Deconfliction	Self accomplishment of tactical plan as externally assigned Medium vehicle airspace separation
3	Robust Response to Real-Time Faults / Events	Health / Status history and models	Tactical plan assigned RT Health Diagnostics Ability to compensate for most control failures and flight conditions	Evaluate status vs. required mission capabilities Abort / RTB if insufficient	Self accomplishment of tactical plan as externally assigned
2	Changeable Mission	Health / Status sensors	RT Health diagnosis Off-board replan	Execute preprogrammed or uploaded plans in response to mission and health conditions	Self accomplishment of tactical plan as externally assigned
1	Execute Preplanned Mission	Preloaded mission data Flight Control and Navigation Sensing	Pre / Post Flight BIT Report status	Preprogrammed mission and abort plans	Wide airspace separation requirements
0	Remotely Piloted Vehicle	Flight Control sensing Nose camera	Telemetered data Remote pilot commands	N/A	Control by remote pilot

Autonomous Control Level Trend



Autonomy-An Engineering Perspective

A system is called “autonomous” if:

- It can monitor its own performance.
- It can detect, isolate and identify incipient failures of its critical components.
- It can predict the remaining useful life of failing components.
- It can take appropriate corrective action to safeguard its integrity for the duration of the emergency.

Design for Autonomy

Design for autonomy requires game changing technologies that synergistically contribute to an **integrated integrity management architecture** that may reduce significantly the operator engagement, while improving attributes of vehicle safety, durability and reliability.

Basic Ingredients for “Design for Autonomy”

- Advanced System Design Concepts—Design for fault Detection/Prediction, Fault Tolerance
- Sensing Strategies
- Modern Control Technologies
- A Hybrid Hardware/Software Framework
- “Smart” Cognitive Concepts-Learning and Adaptation

Autonomy-The Constituent Technologies

Brief Remarks on:

- Risk
- Confidence
- Uncertainty Management
- Fault-Tolerant Control

Autonomous Systems: Problems, Challenges, Enablers

❖ An Autonomous Vehicle Operator (AVO) at times, "he's been more overcome by the torrent of information pouring in during a drone flight than he was in the cockpit".

❖ Need: Improve Reliability, Availability, Safety

❖ The Defense Science Board: "issues including the need to build trust in autonomous systems while also improving the trustworthiness of autonomous capabilities"

❖ Enablers:

-Integrity Management/Prognostics and Health Management

-Resilient design and operation

-Safety assurance/risk management

Assured and Trusted Autonomy

Assurance and Trustworthiness for Unmanned Autonomous Systems

Assured autonomy *enabled via:*

- *Integrity Management/Prognostics and Health Management strategies*
- *Design for resilience and reliability, i.e. endowing unmanned systems with properties to withstand/accommodate severe external/internal disturbances*
- *Safety assurance and risk management*
- ***Trusted autonomy*** - *achieved via quantifiable metrics of* ***confidence, risk and trust consensus***.

Design for Autonomy

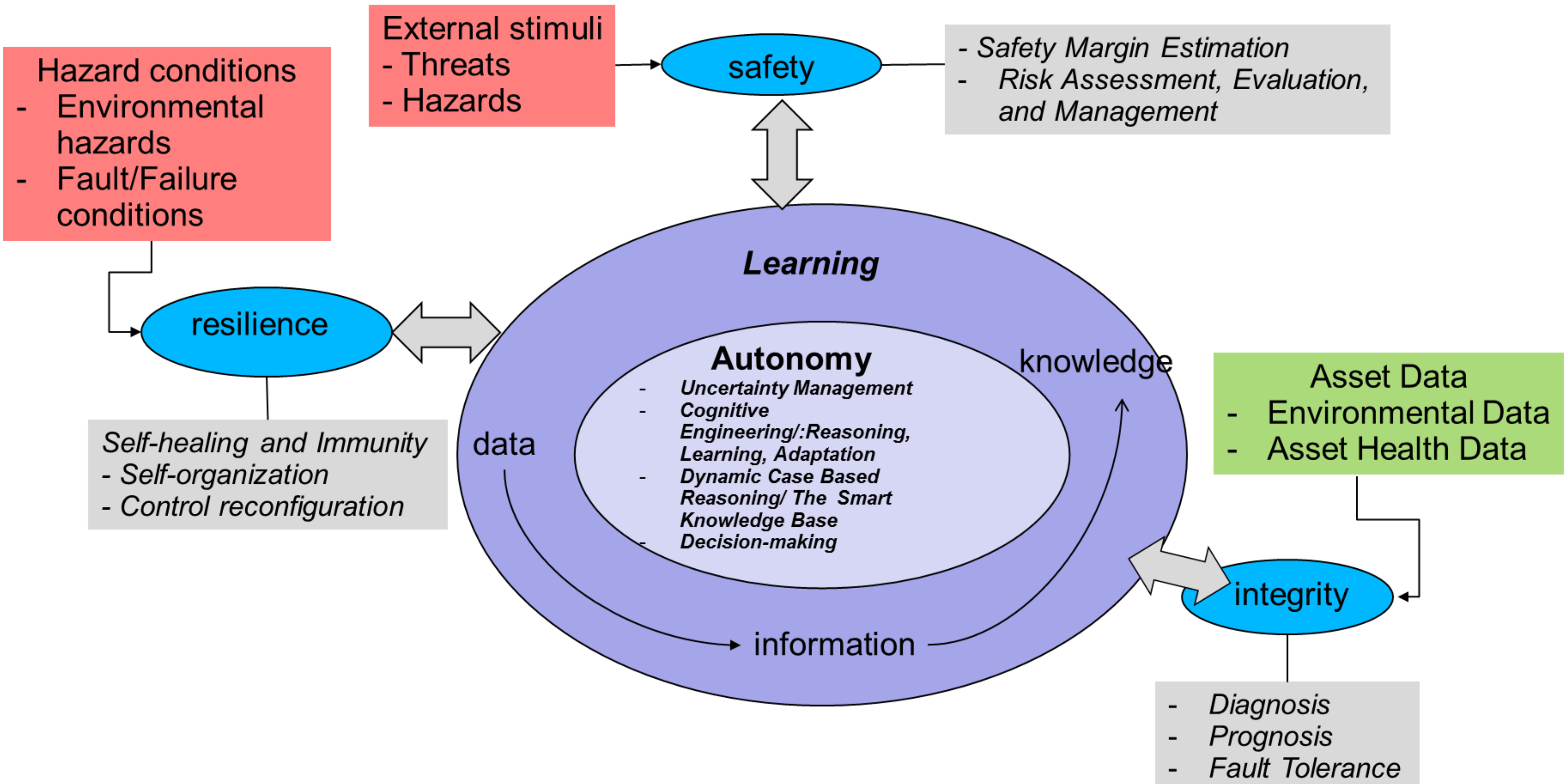
- Design for autonomy requires game changing technologies that synergistically contribute to an integrated integrity management architecture that may reduce significantly the operator engagement, a necessary requisite for space/aerospace vehicles executing long-term missions, while improving attributes of vehicle safety, durability and reliability.

Fundamental ingredients for autonomy

- Risk
- Confidence
- Uncertainty Management
- Fault-Tolerant Control

Learning-Enabled Assured Autonomy

- **Assurance:** Process of providing confidence in operation of autonomous vehicles in nominal and off-nominal/hazardous conditions.
- Adaptation and Learning for Realizing Assured Reasoning
- Learning-enabled conversion of Data into Information and Knowledge
- Utilize new evidence to adapt system behavior leading to desired optimal performance



DATA

- Time series data
- Images
- Text/Observations

INFORMATION

- Features
- Patterns
- Motifs

KNOWLEDGE

- Learning-enabled knowledge
- Discovery
- Acquisition

- Knowledge about assured / trusted autonomy
- Feedback for learning-enabled knowledge enhancements

Autonomous Systems: Problems, Challenges, Enablers

- An Autonomous Vehicle Operator (AVO) at times, “he’s been more overcome by the torrent of information pouring in during a drone flight than he was in the cockpit”.
- 40% of Class Air Mishaps Attributed to UAVs
- Need: Improve Reliability, Availability, Safety

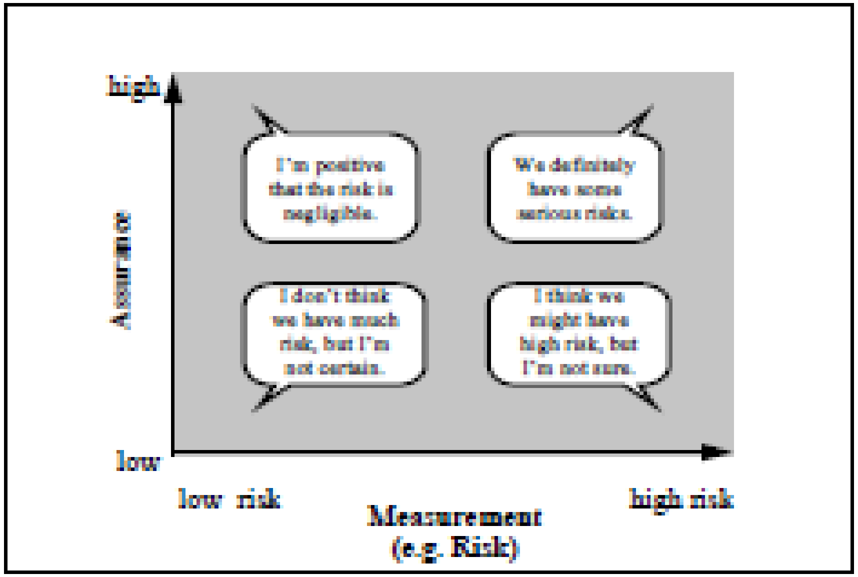
- Enablers:
  - -Integrity Management
  - -Resilience
  - -Safety

The Need

- Develop new and innovative technologies to establish *"assured and trusted autonomy"* even in the presence of extreme hazards/dangers (internal/external).
- ***Assured autonomy*** is enabled via verifiable means to detect, identify and predict the evolution of incipient vehicle failure modes, and take action to mitigate the contingency; design for resilience and reliability,
- ***Trusted autonomy*** is achieved via quantifiable metrics of confidence, risk and trust consensus. The Defense Science Board study focused on "issues including the need to build trust in autonomous systems while also improving the trustworthiness of autonomous capabilities"

Assurance and Trustworthiness

Assurance: the degree of confidence that the system performs its assigned tasks with acceptable risk. Assurance and risk are orthogonal, i.e. as assurance increases, risk decreases.



The Enabling Technologies

Achieving significant gains in **assured and trusted autonomy** and autonomous operation of critical unmanned assets will require developing new and innovative technologies to establish "assured and trusted autonomy" even in the presence of extreme hazards/dangers. Through integrated system health management, resilient design and operation of UAVs and swarms of vehicles, adaptive vehicle control, and safety/risk assessment and management technologies enabling complex systems to operate across a range of functional capabilities.

Design of Self-Organizing, High Confidence Systems

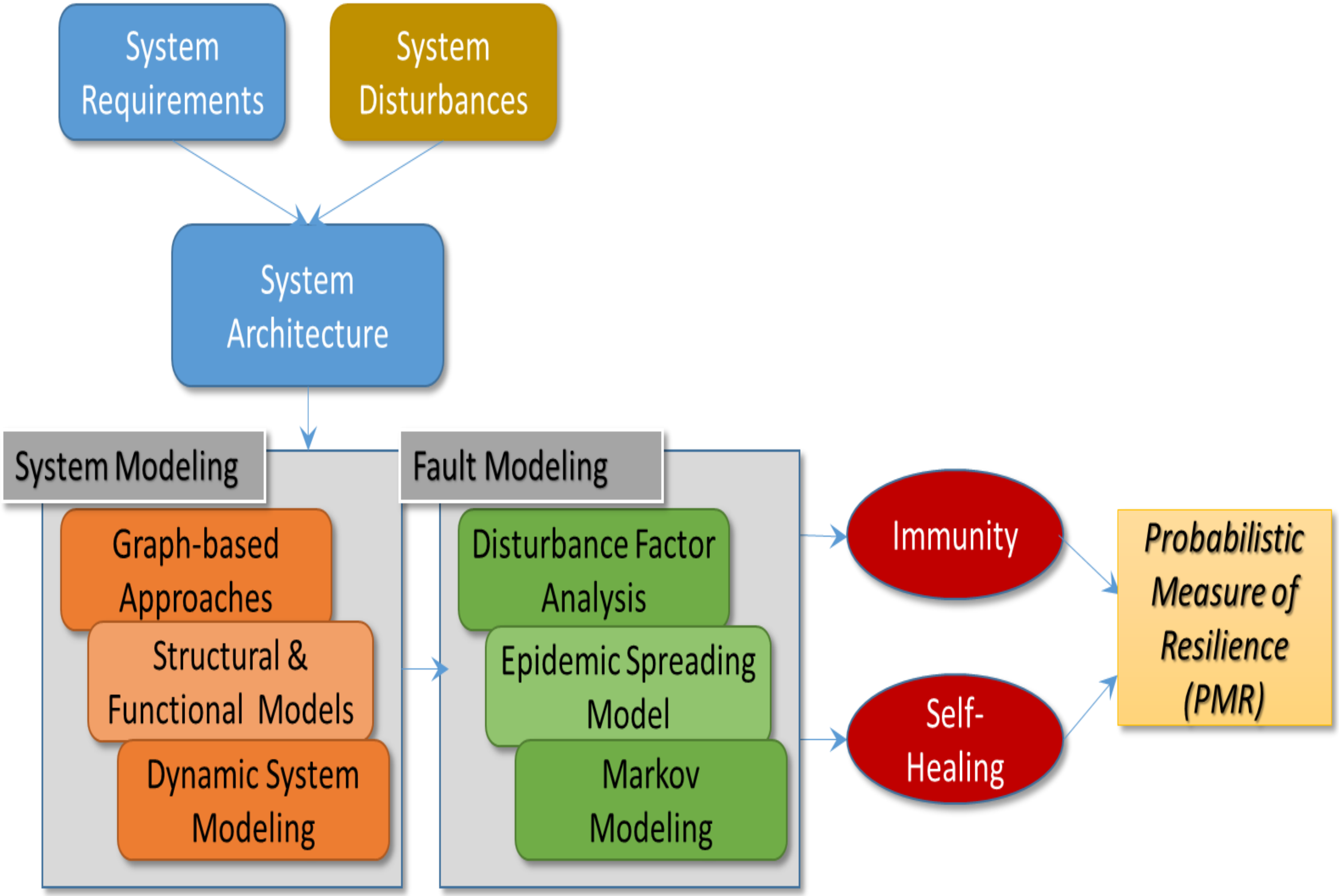
Resilience

- **Resilience:** Ability of a system to absorb disturbances (faults, failures, shocks, wind gusts, etc.) without an adverse effect on the system's operational integrity; ability to predict failures and proactively adapt (or recover) from possibly detrimental events.

- **Objective:** Determine optimum design parameters that lead to the best system performance in terms of achieving maximum resilience

➢ *Develop novel and fundamentally sound technologies into the design, control and operation of complex aerospace systems.*

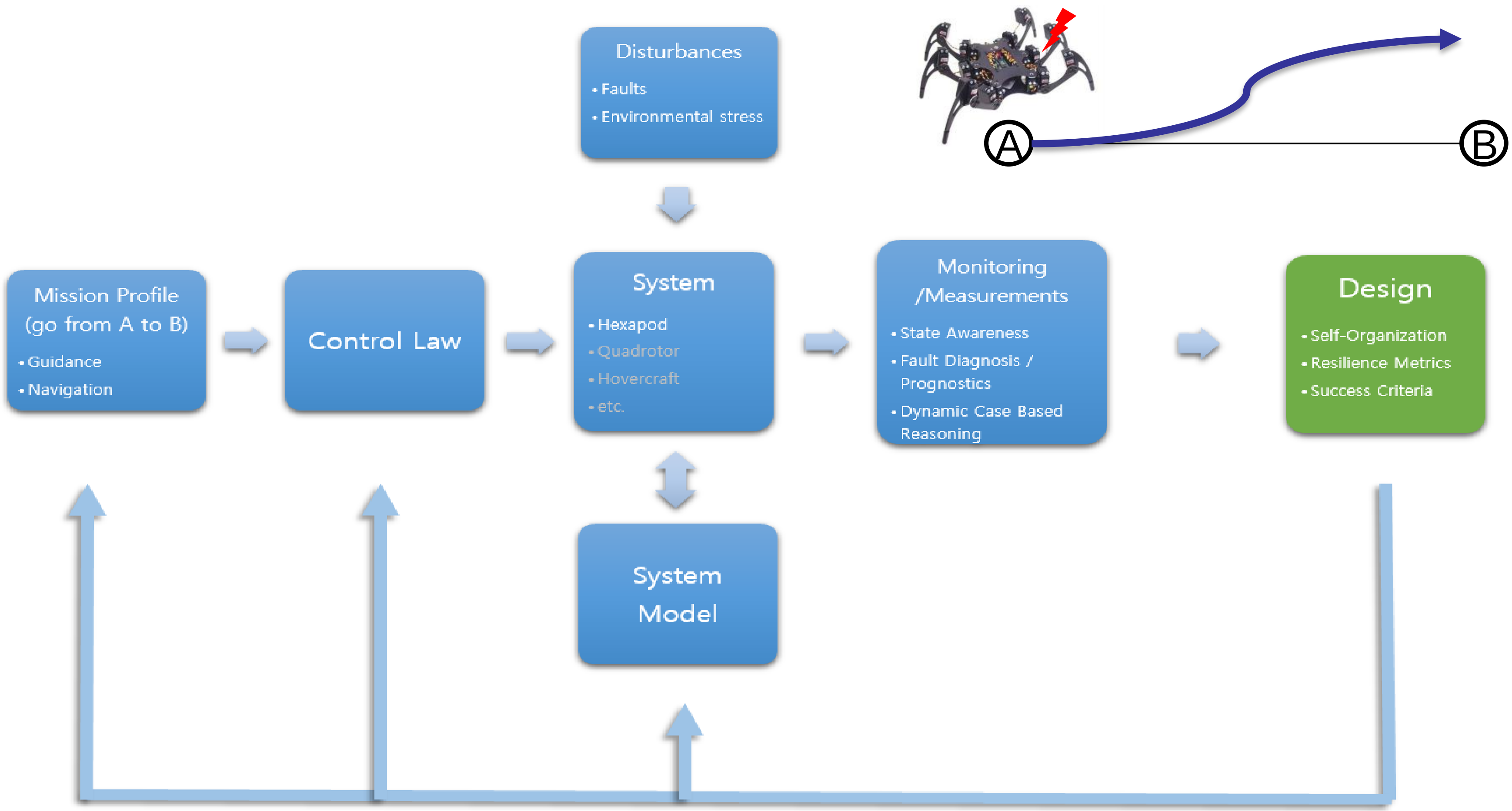
Design for Resilience



Design of Self-Organizing, High Confidence Systems

- Current State and Goal:
  - Complex systems are vulnerable to severe faults/failures and external hazards/dangers
  - Expedient adjustment of system functionality in response to sudden (expected/unexpected) changes in performance requirements
  - Flexible adaptation to new mission profiles and/or incipient failure modes
- Self-Organization:
  - Overall system order arises from local interactions between parts of an initially disordered system
  - Spontaneous (no external control agent required)
  - Decentralized, distributed over all components
- Enablers:
  - Data acquisition: MATLAB
  - Modeling: physical, functional, nonlinear dynamic, graph-theoretic
  - Policy design tools: self-organization strategy, requirements (constraints) definition, Markov Decision Process, optimization methods (dynamic programming)
  - Success criteria: performance function (time, energy, position), system stability

Overall Self-Organization Methodology



Control Strategies

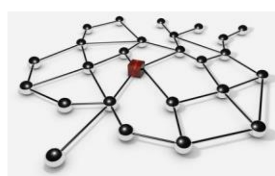
Hierarch of Controls

1. Eliminate the hazard
2. Reduce the hazard level
3. Provide safety devices
4. Provide safety warnings
5. Provide safety procedures

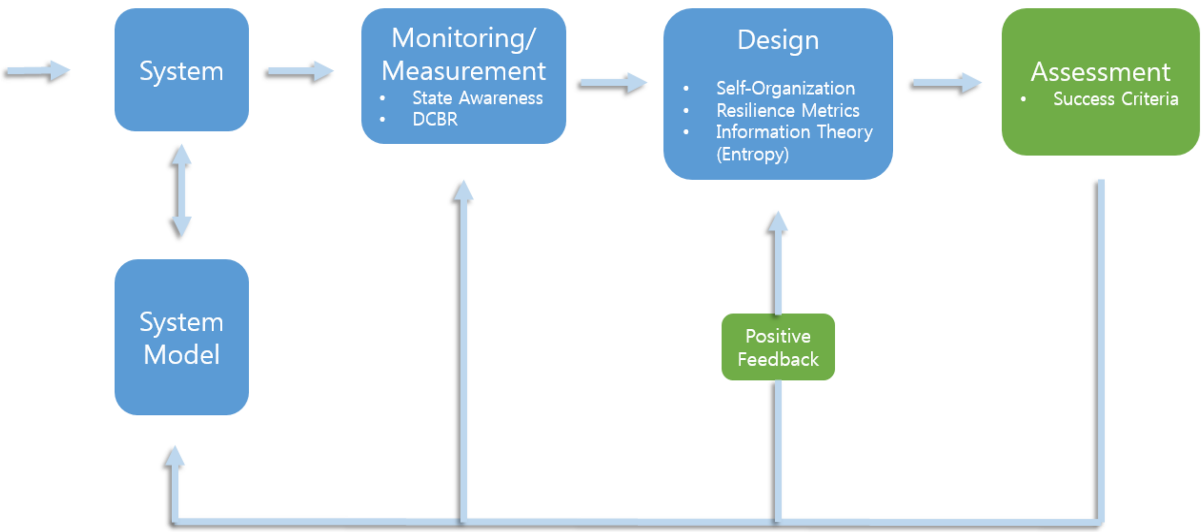
Novel Modeling Tools

We propose a ***graph spectral*** *approach* to calculate the resilience of complex cyber physical systems based upon the system topology, using eigenvalues of the system adjacency and Laplacian matrices.

Non-linear dynamical system *(NLDS) and* ***epidemic spreading*** models used to quantify both the immunity as well as the self-healing properties of the system



Self-Organization Strategy (Performance Assessment & Feedback)

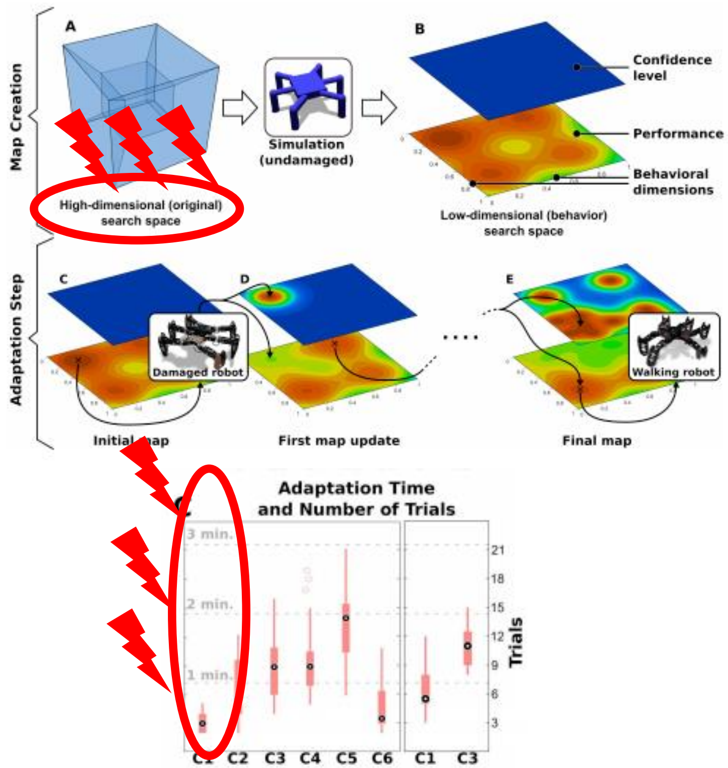


- Compensating action is internally generated and applied
- Positive feedback in hexapod joint rotational forces to be explained later

Safety Risk Management

- Safety requires effective practices in managing risk from
  - The Aircraft
  - The Environment
  - The Human
- Process
  - Identify hazards
  - Risk Analysis and Assessment
  - Controls for Risk
  - Residual Risk
  - System Operation

Innovation



- Reliable, autonomous approach to designing a resilient system
- No additional components needed (cost-effective)
- Compared to trial & error methods:
  - Minimal search space for action through optimization routine (compared to high-dimensional search space)
  - MDP map of policy computes the optimal compensating action instantly (compared to minutes of adaptation time)

State of the Art

- ❖ Improved system resilience has been addressed in the recent past via a number of methodologies (including extensive work by this research team and others) on fault accommodation or fault tolerant control and adaptation, robust control, intelligent control, among others.
- ❖ Electronics/Software – FPGAs, patterning, other

UAS Safety Management System

Potential Impact

The design for resilience methodology will have a potential impact on many critical areas such as unmanned autonomous cyber physical systems, Unmanned aerospace platforms, communication networks and the emerging field of cyber security.

Software Resilience

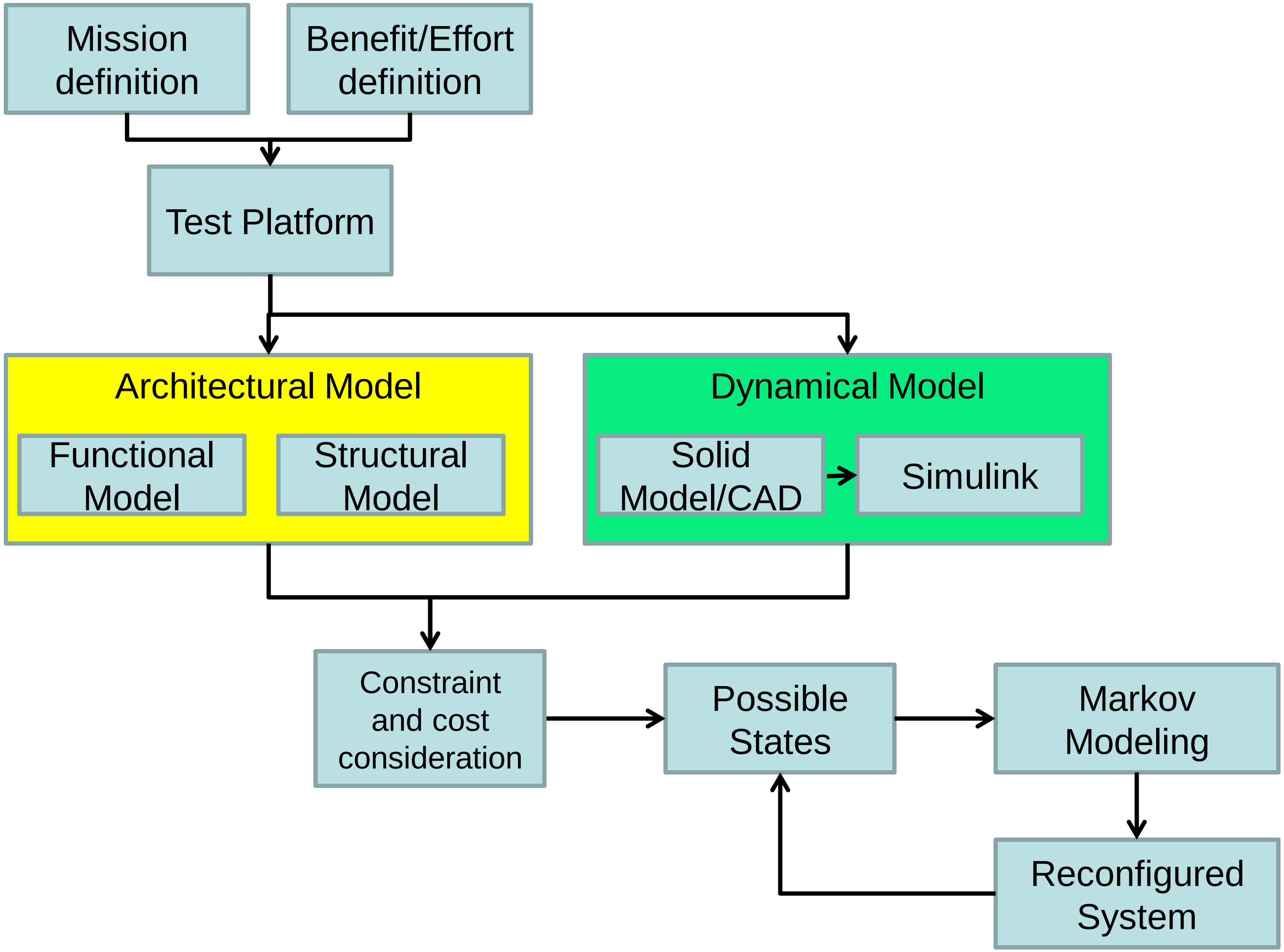
- A Cyber Physical System should be resilient to both hardware and software faults/extreme disturbances.

- Software strategies are required to make the system resilient to: (a) errors in the code, (b) inaccurate or unpredictable context, (c) attacks from malicious entities/sources – the security issue.

- The first ones are addressed at the design stage via formal verification tools/methods but malicious attacks may be severely endangering the integrity of the vehicle.

- We introduce an intelligent approach to the detection and identification of malicious attacks based on a reasoning paradigm and detection, isolation and identification tools/methods developed for fault/failure modes.

Reconfiguration Methodology - Overview



The Control Architecture

Health vs. Performance

Assumptions

RUL Related States vs. Tracking Error

Adaptation parameter ρ adjusts cost

- The cost function:

$$J = \min_{\Delta \mathbf{u}} \int_{t_0}^{t_0+T} [(\mathbf{x}-\mathbf{x}^{\star})^{\top} (\;\mathbf{Q})(\mathbf{x}-\mathbf{x}^{\star}) + \Delta\mathbf{u}^{\top}\mathbf{R}\Delta\mathbf{u}]\,dt$$

- Subject to the constraints,

$$\begin{cases} \Delta\mathbf{u}_{\min} \leq \Delta\mathbf{u}(t) \leq \Delta\mathbf{u}_{\max} \\ \quad \mathbf{u}_{\min} \leq \quad \mathbf{u}(t) \leq \quad \mathbf{u}_{\max} \end{cases}$$

Concluding Comments-Future Challenges

- Reconfigurable software- programming architectures
- Combine hardware, control and software reconfiguration strategies
- Design for reconfigurability: the complexity paradigm
- A confluence of tools/methods from large-scale system theory, complexity theory, coordination and control, etc.

Self-Organization - Epidemic Analysis

- The epidemic threshold τ is a measure showing if the impact/disturbance will survive or die out over time and is defined as β/δ where β is the infection rate, and δ is the recovery rate

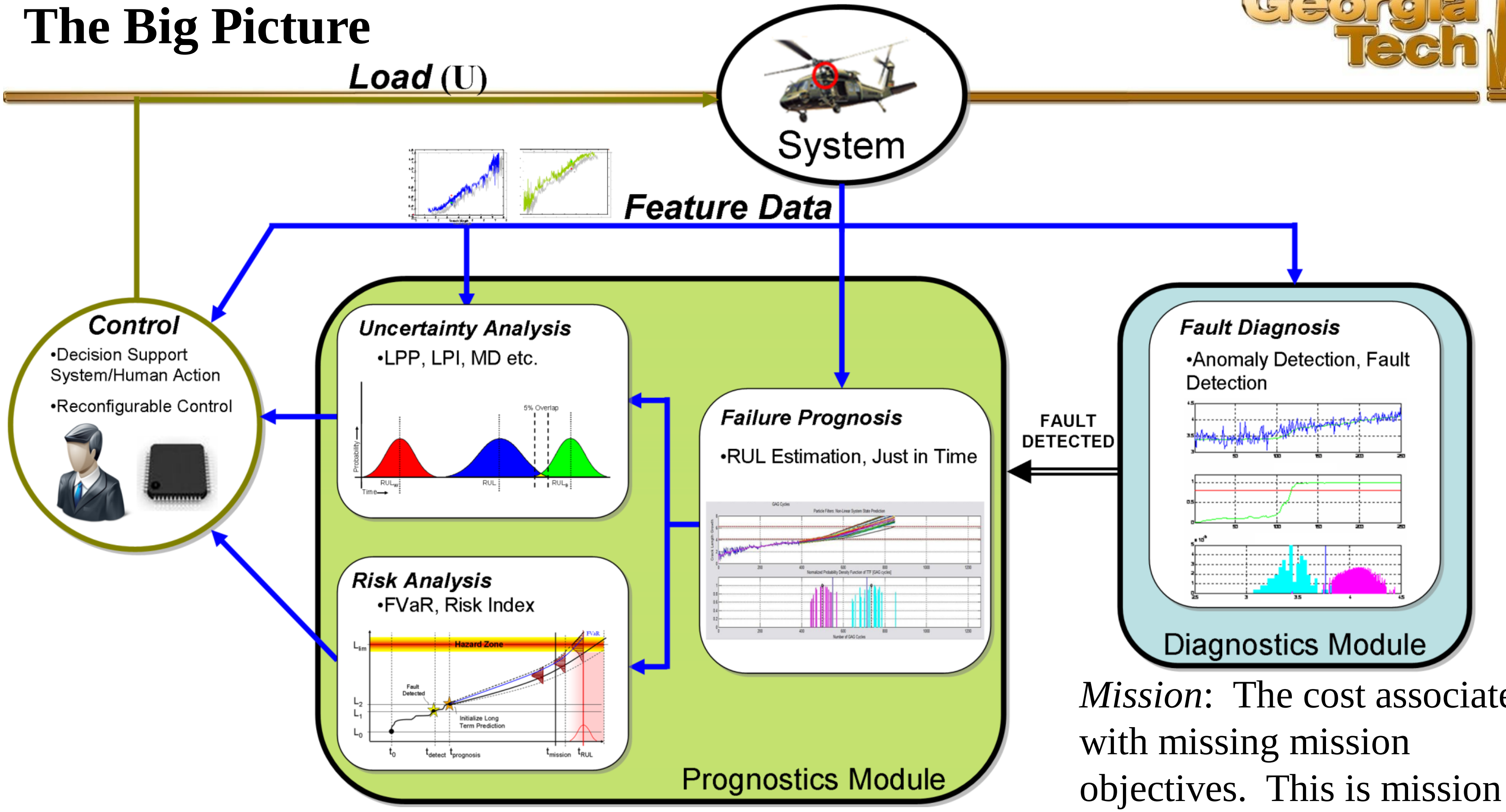
Question: How do we predict the epidemic threshold?

- The epidemic threshold of a star-topology (hexapod) network can be expressed as $\frac{1}{\sqrt{d}}$ where d is the degree (# of connections) of the central (core) node.

A strategic approach to effective self-organization for fault tolerance would be to predict the most crucial impact areas and start from there.

$$\Theta_k(t) = \sum_{k'} P(k'|k)\rho_{k'}(t).$$

The Big Picture

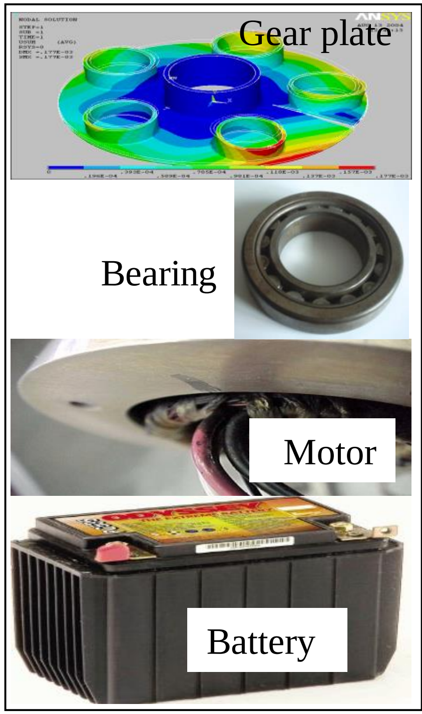


Mission: The cost associated with missing mission objectives. This is mission dependent and possibly subjective.

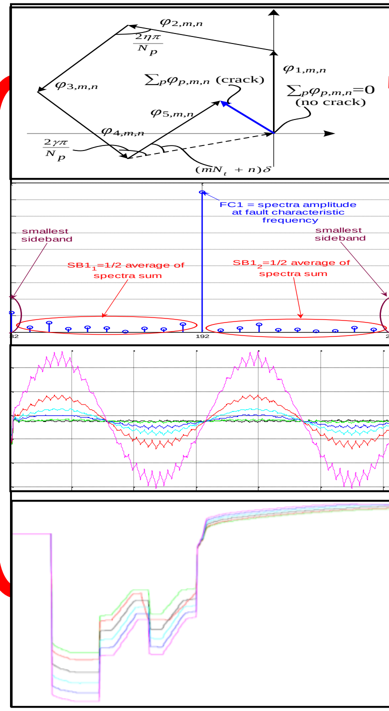
1) Choose U to Minimize a Cost Function: $J = \alpha FVaR + \beta Mission$

2) This U is called U_{opt}. The Uncertainty metrics provide a bound around U_{opt} where the system operator may adjust U and still ensure system reliability and sub-optimal operating conditions.

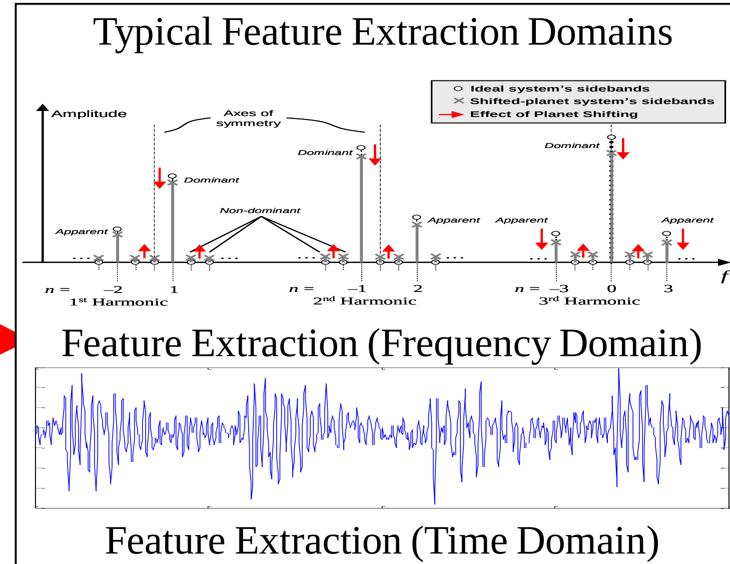
Understanding the Physics of Failure Mechanisms



Various Systems

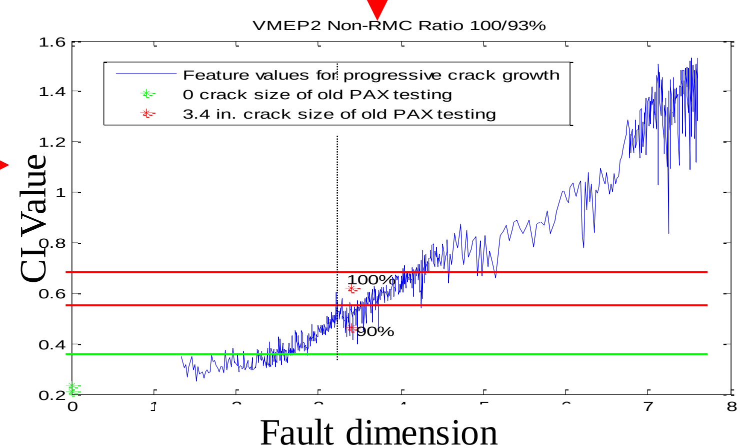


Failure Mechanism



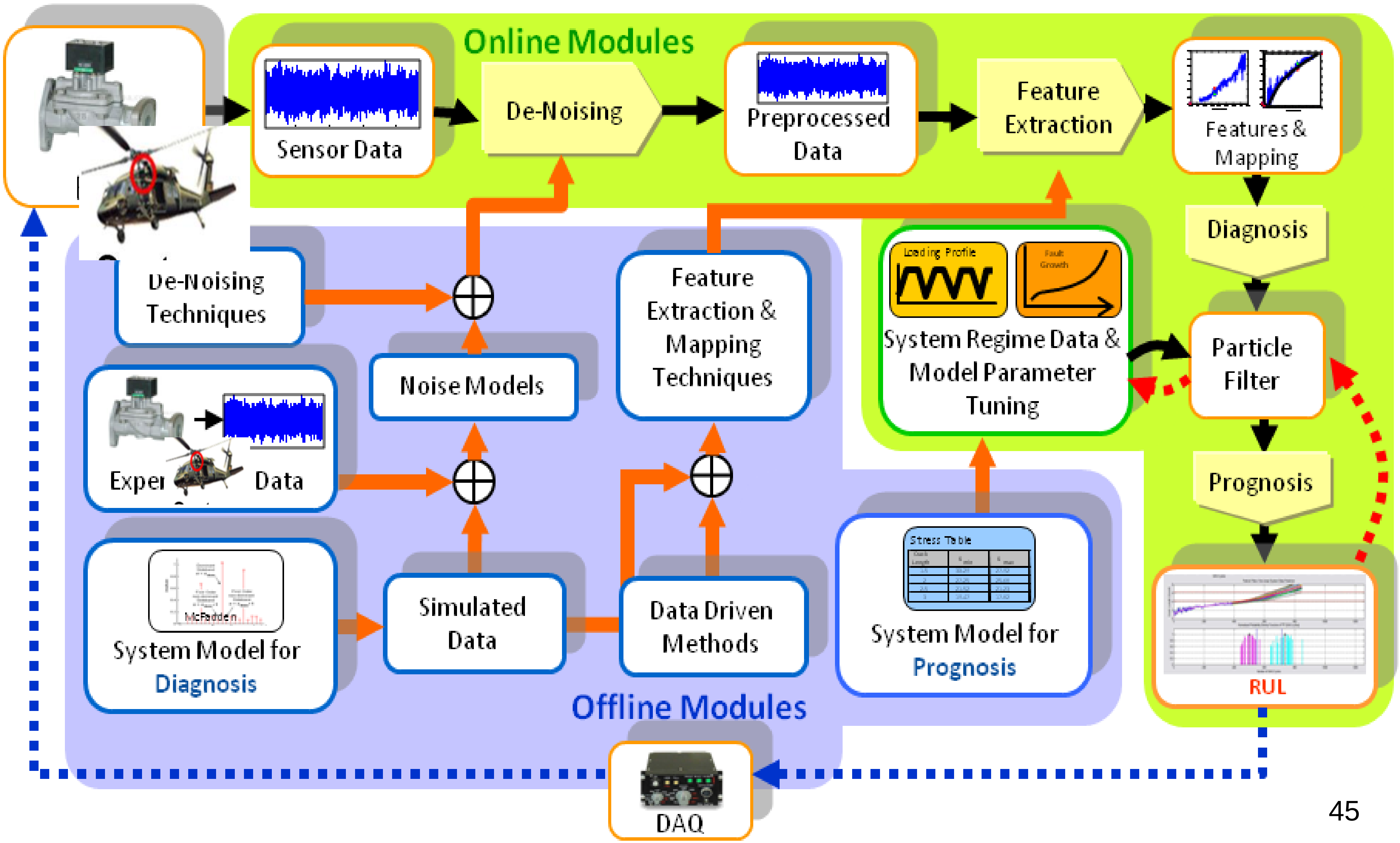
CI Value

Ground Truth Fault Dimension



- Optimum Feature Selection
- Mapping of Features vs. Fault Dimension
- Utility in Diagnosis / Prognosis

Background
PHM Architecture

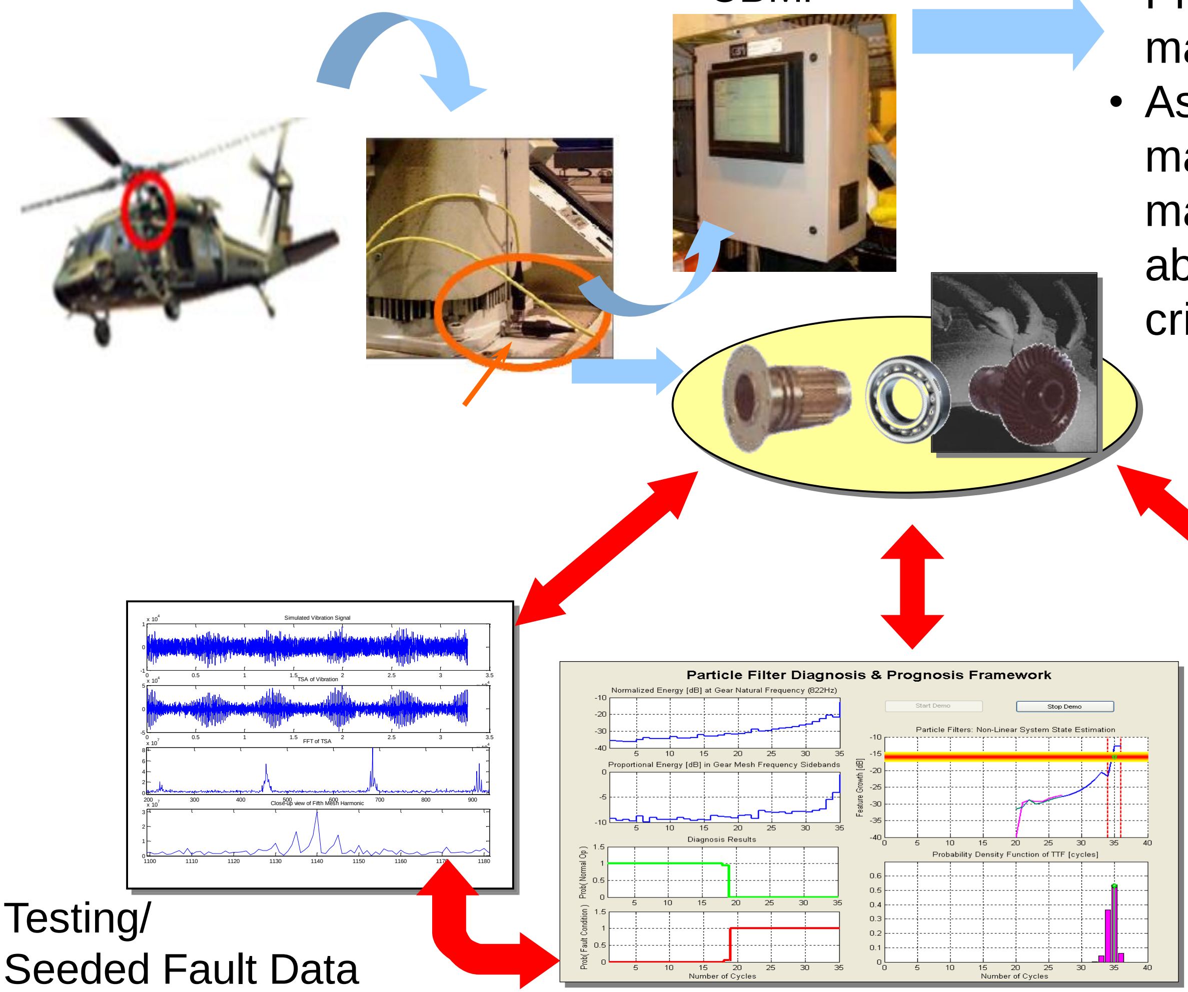


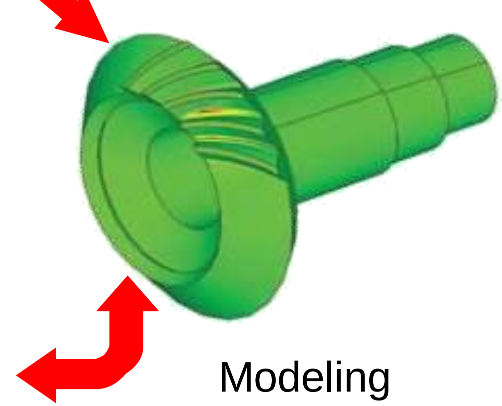
A Systems Engineering Approach to Integrity Management

Testing, Modeling, and Reasoning Architecture – The Enabling Technologies for CBM

CBMi

- Prevent unscheduled maintenance
- Assist the maintainer/manager in making intelligent decisions about the health status of critical equipment/facilities





Testing/
Seeded Fault Data

Modeling

Reasoning Architecture for Diagnosis-Prognosis

Particle Filtering Anomaly Detection Framework

➢ The implementation Philosophy:

- Initially, noisy accelerometer measurements suggest that the fault hypothesis (crack, for example) is rejected. Confidence in fault being detected ~ 0-5%.
- A fault (crack) is initiated and its evolution is tracked via a model.

$$L(k+1) = L(k) + C \cdot (\Delta K)^x + w(k)$$

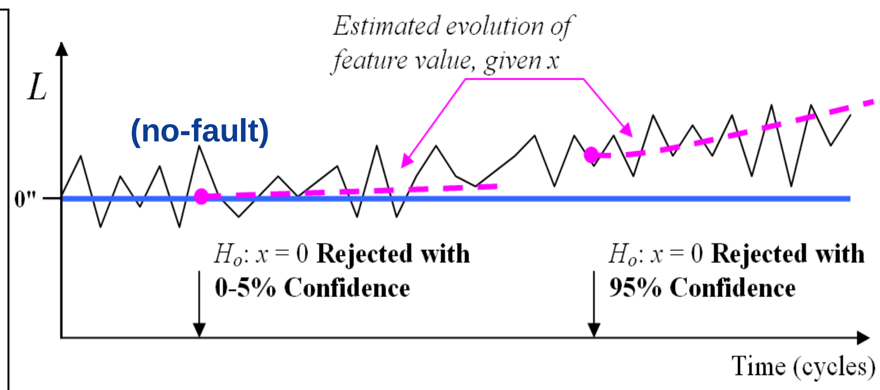
where:

$L(k)$: Crack length at time instant k

C: Material related coefficient

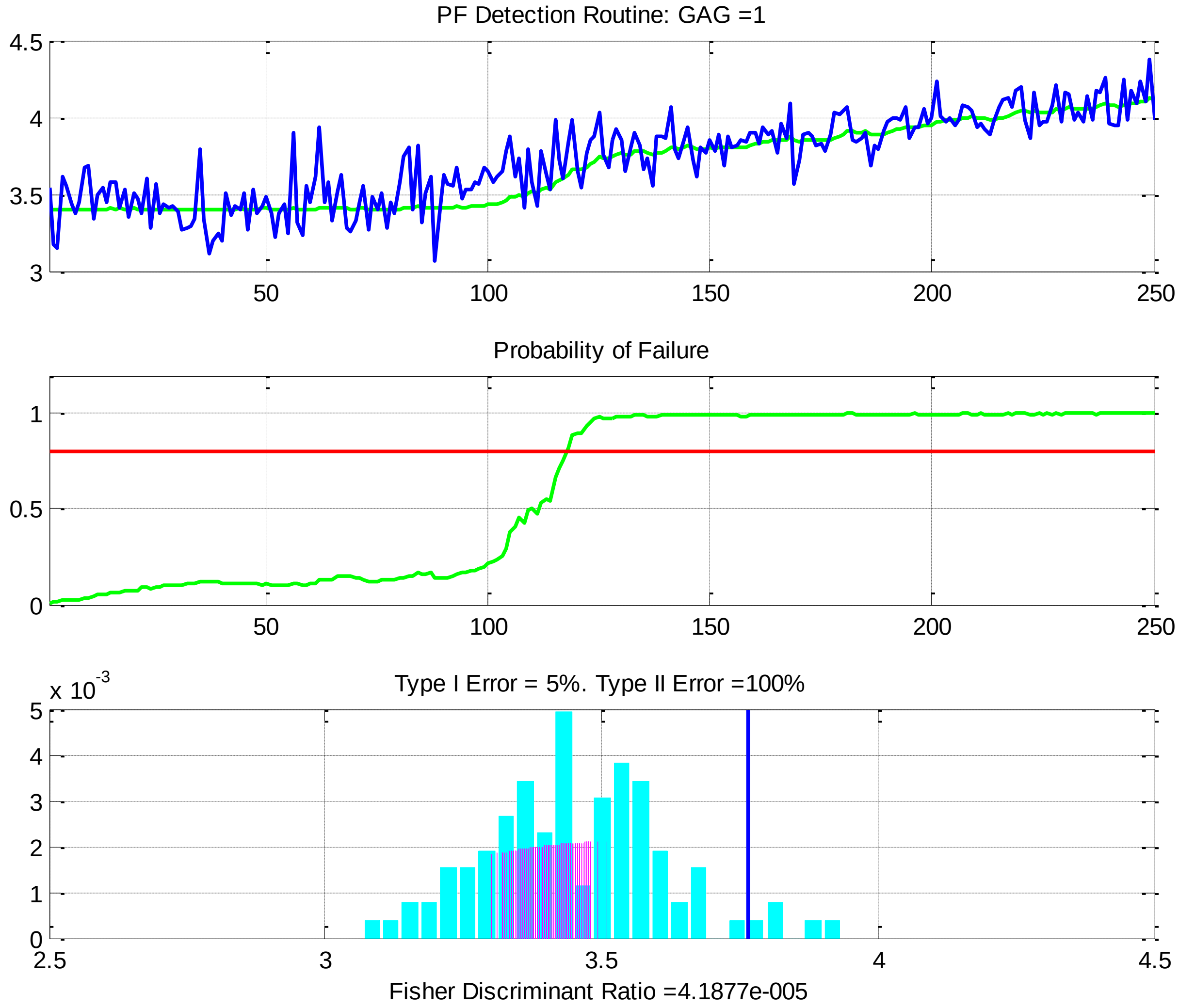
ΔK: Stress variation due to load profile

$w(k)$: white noise signal



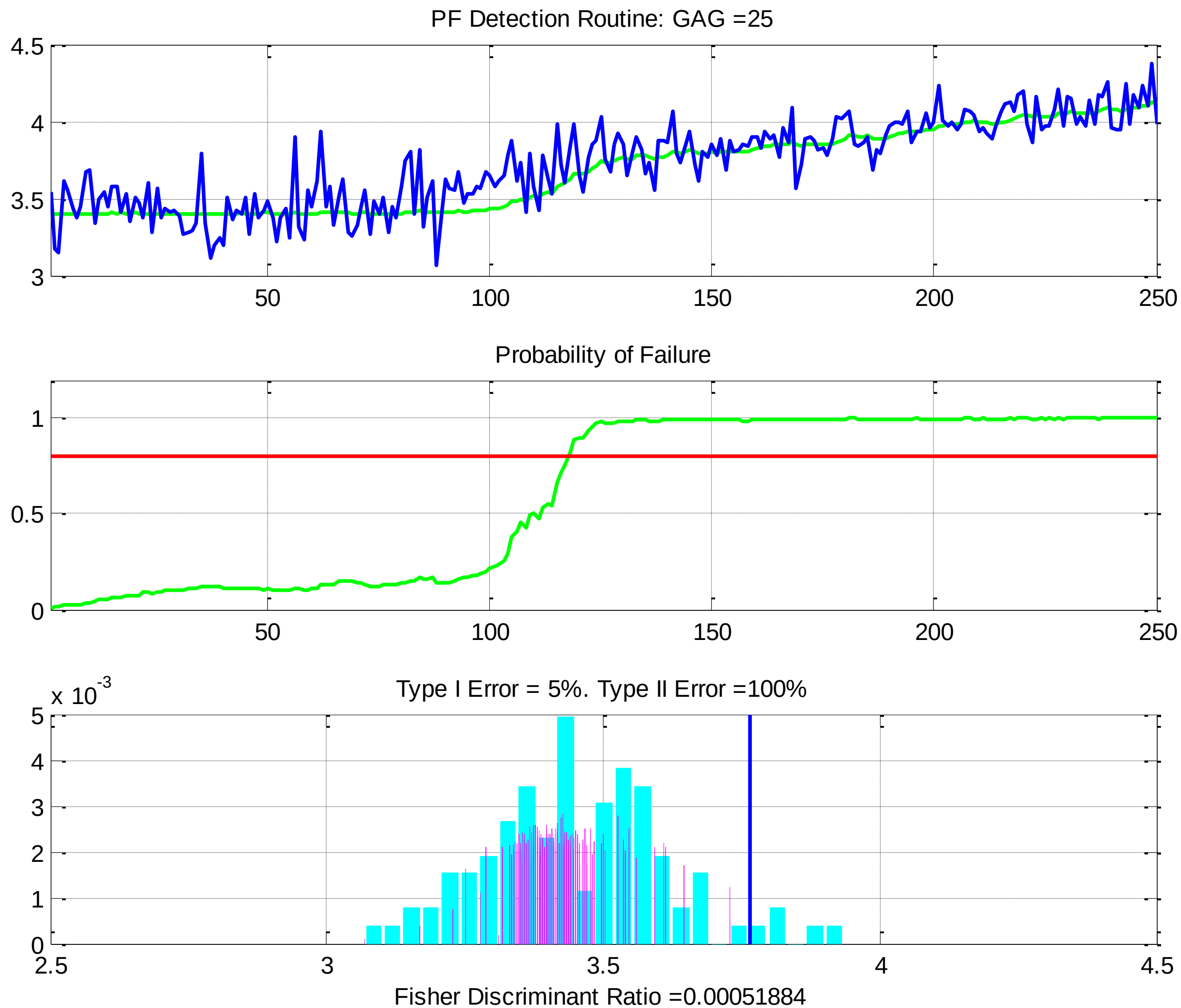
Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%



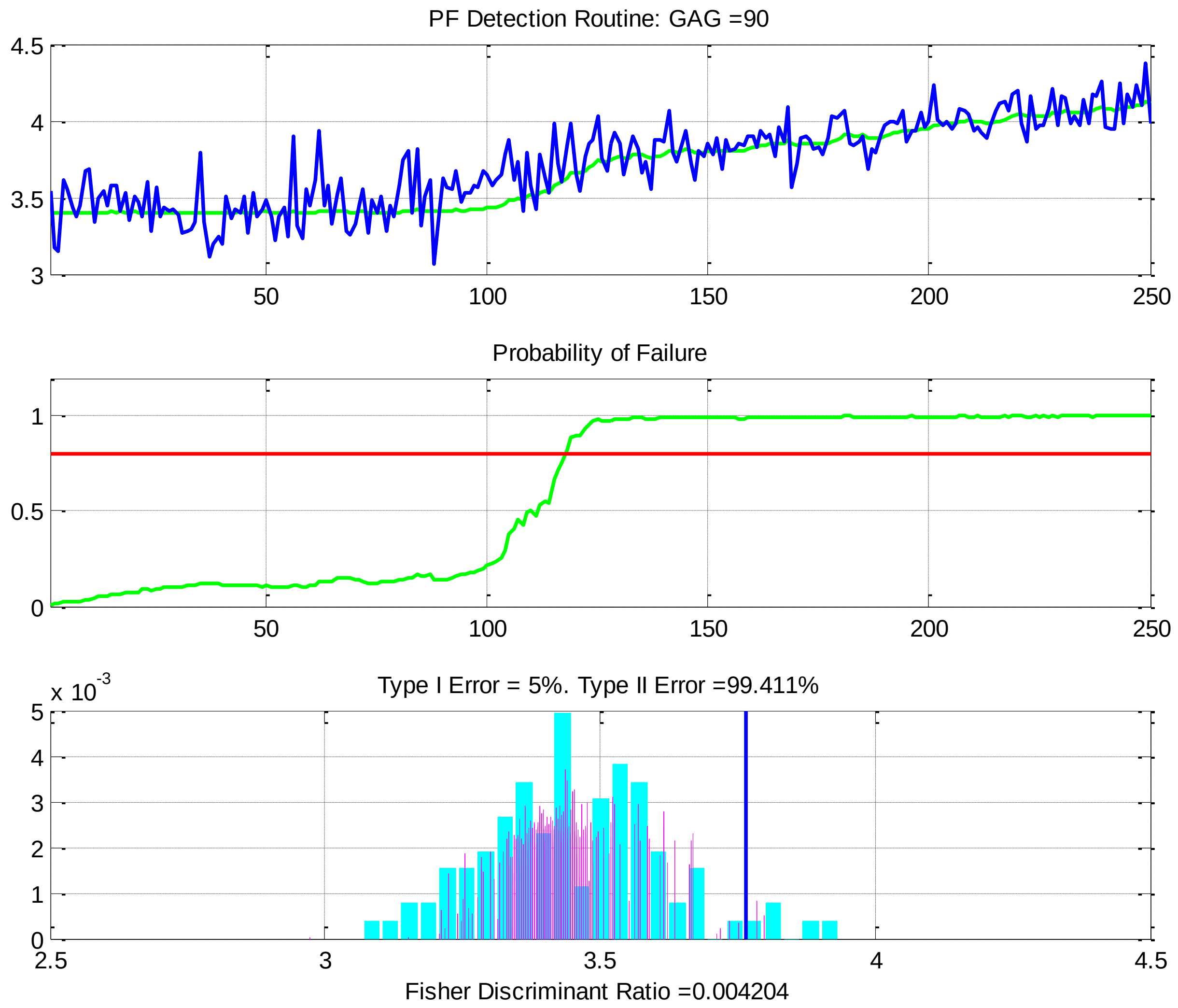
Particle Filtering FDI Framework

Detection Results: Type I Error = 5%



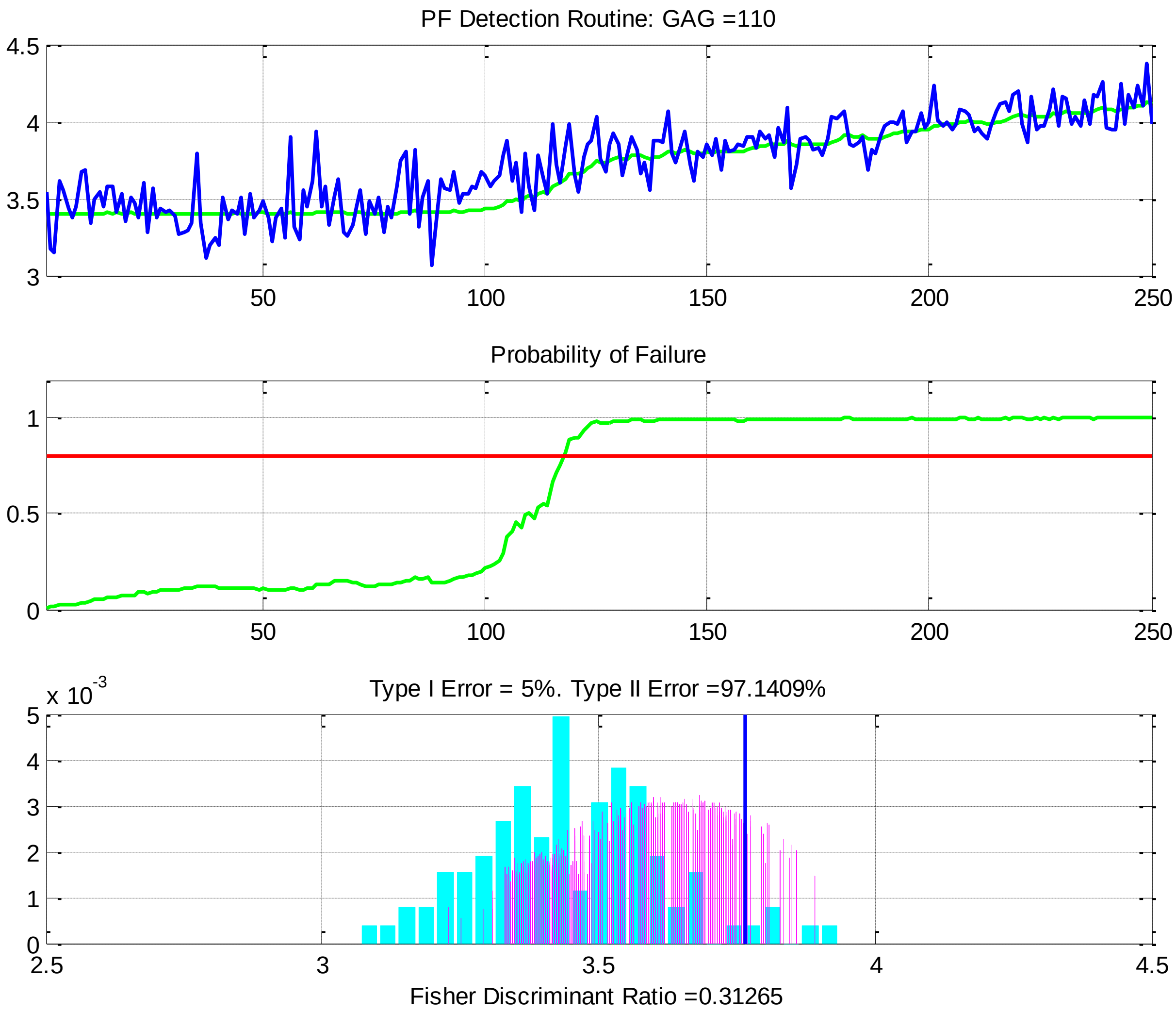
Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%



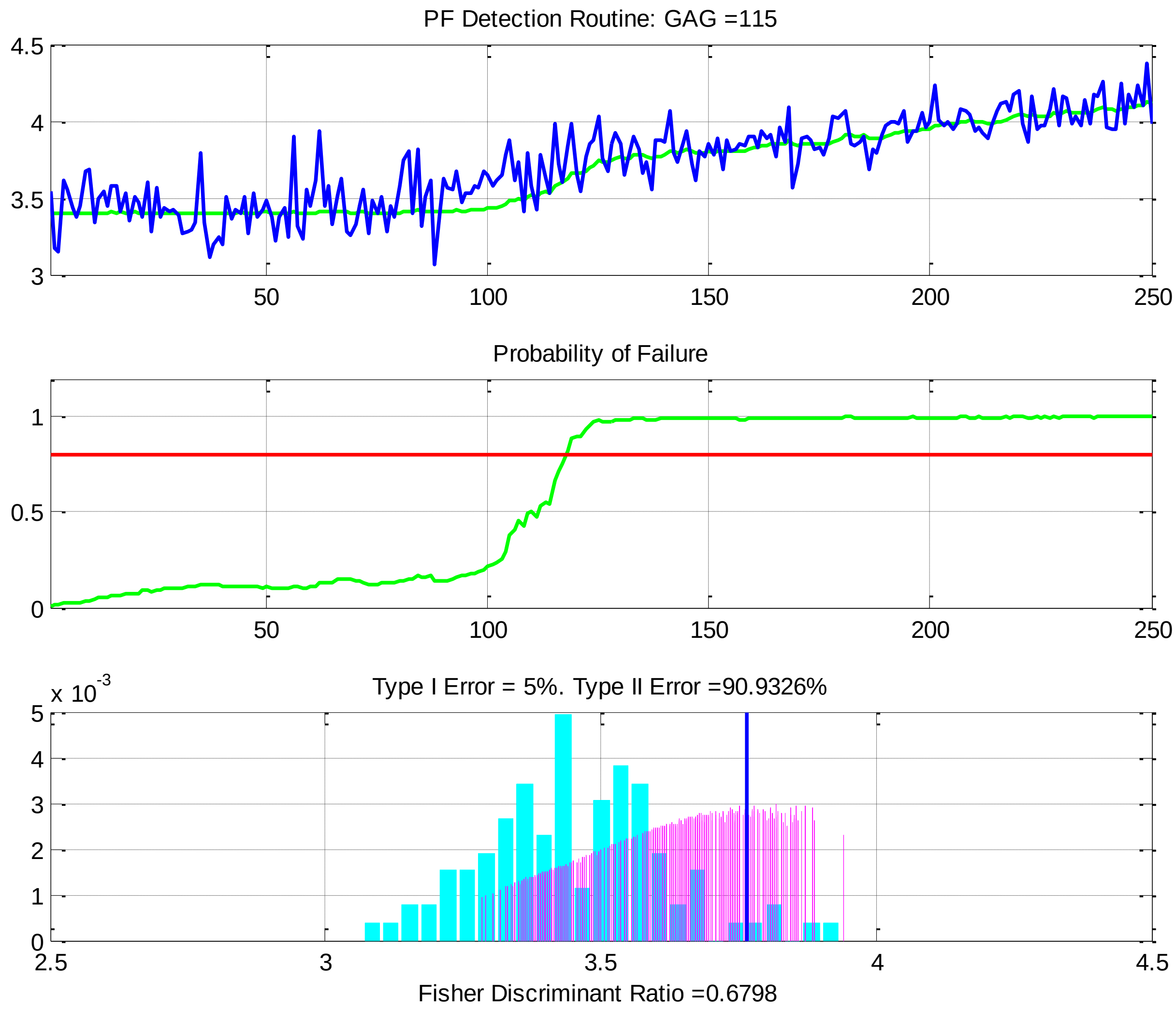
Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%



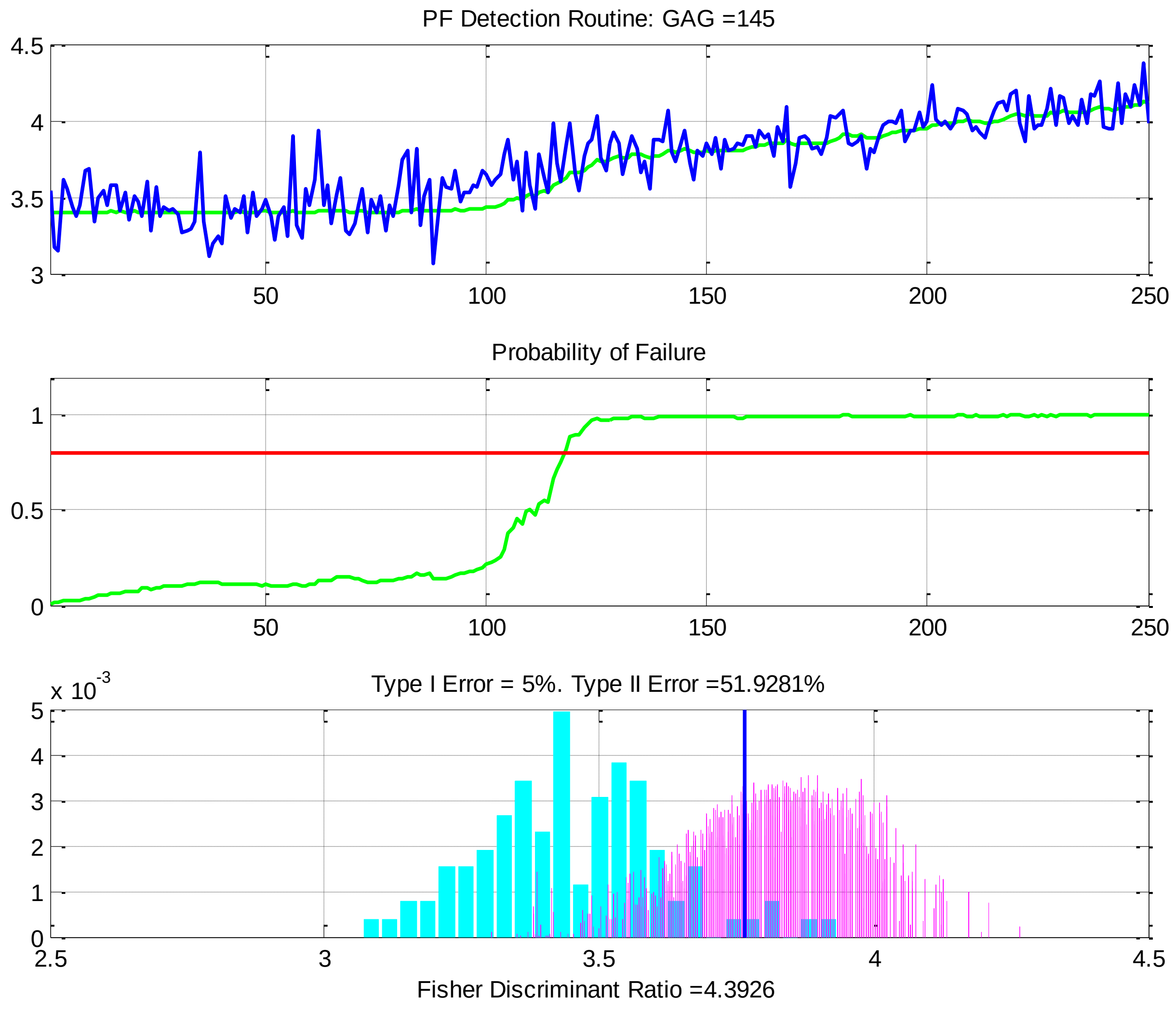
Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%



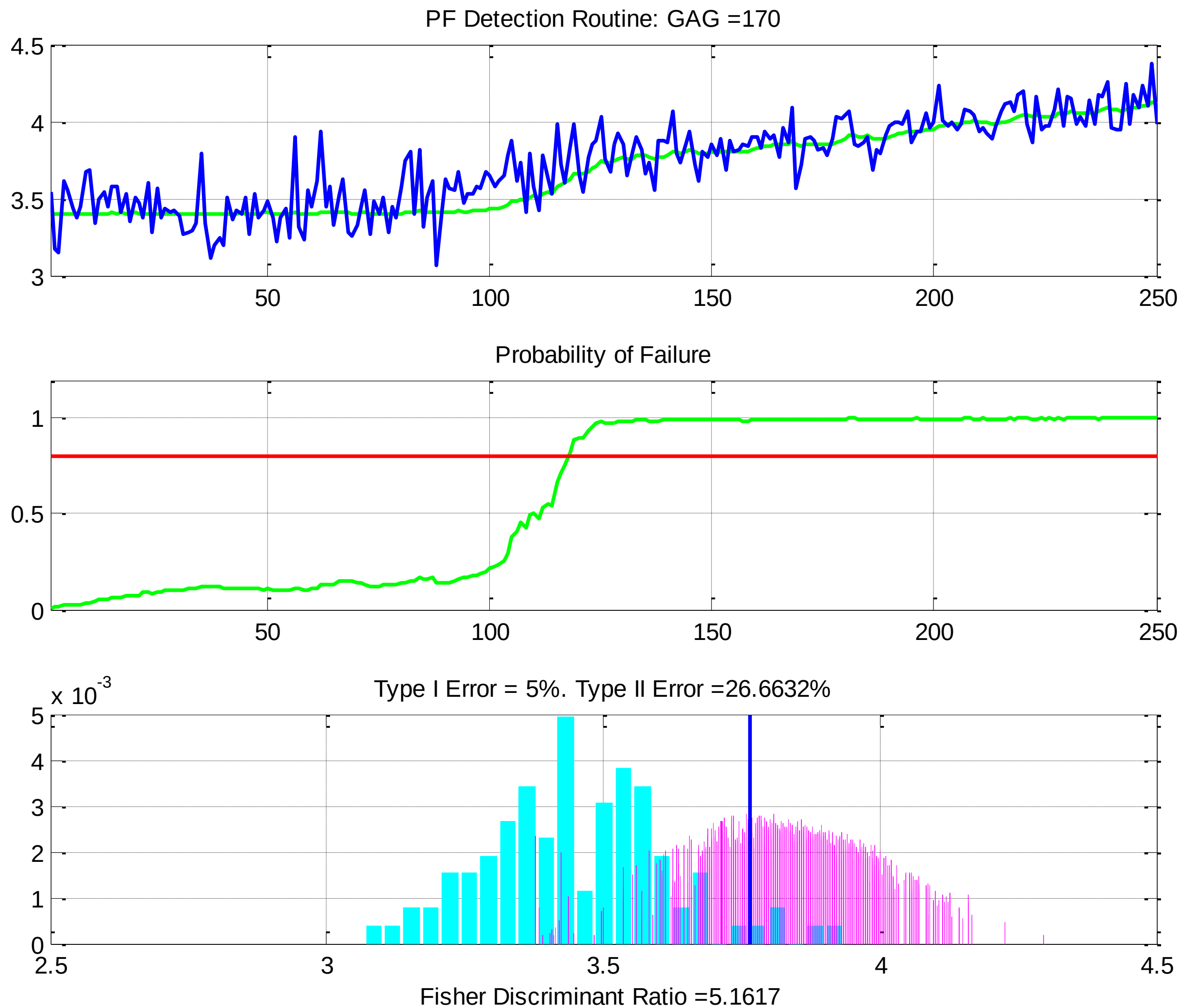
Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%



Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%

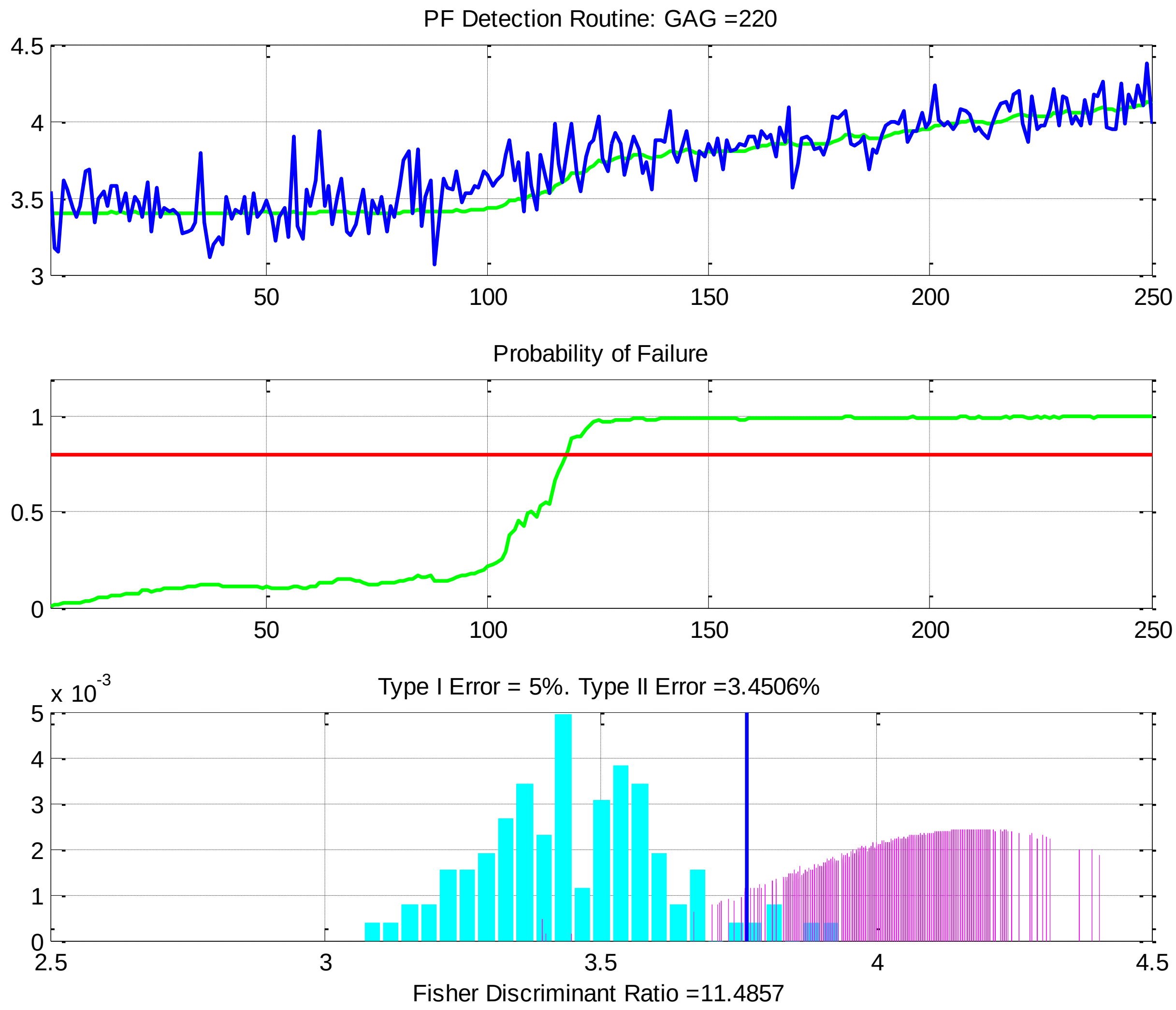


Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%

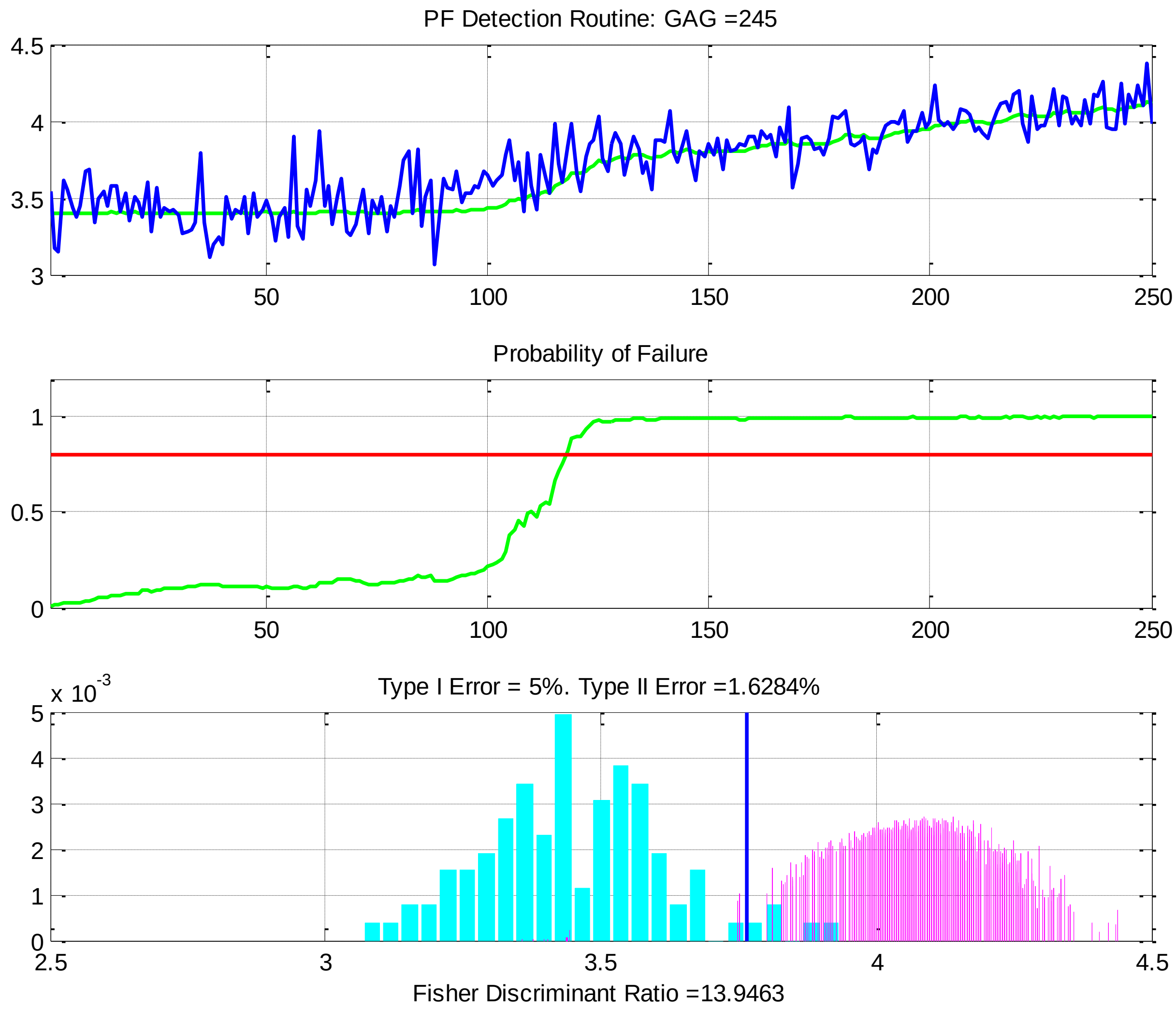
Particle Filtering-FDI Framework

Detection Results: Type I Error = 5%



Particle Filtering-FDI Framework

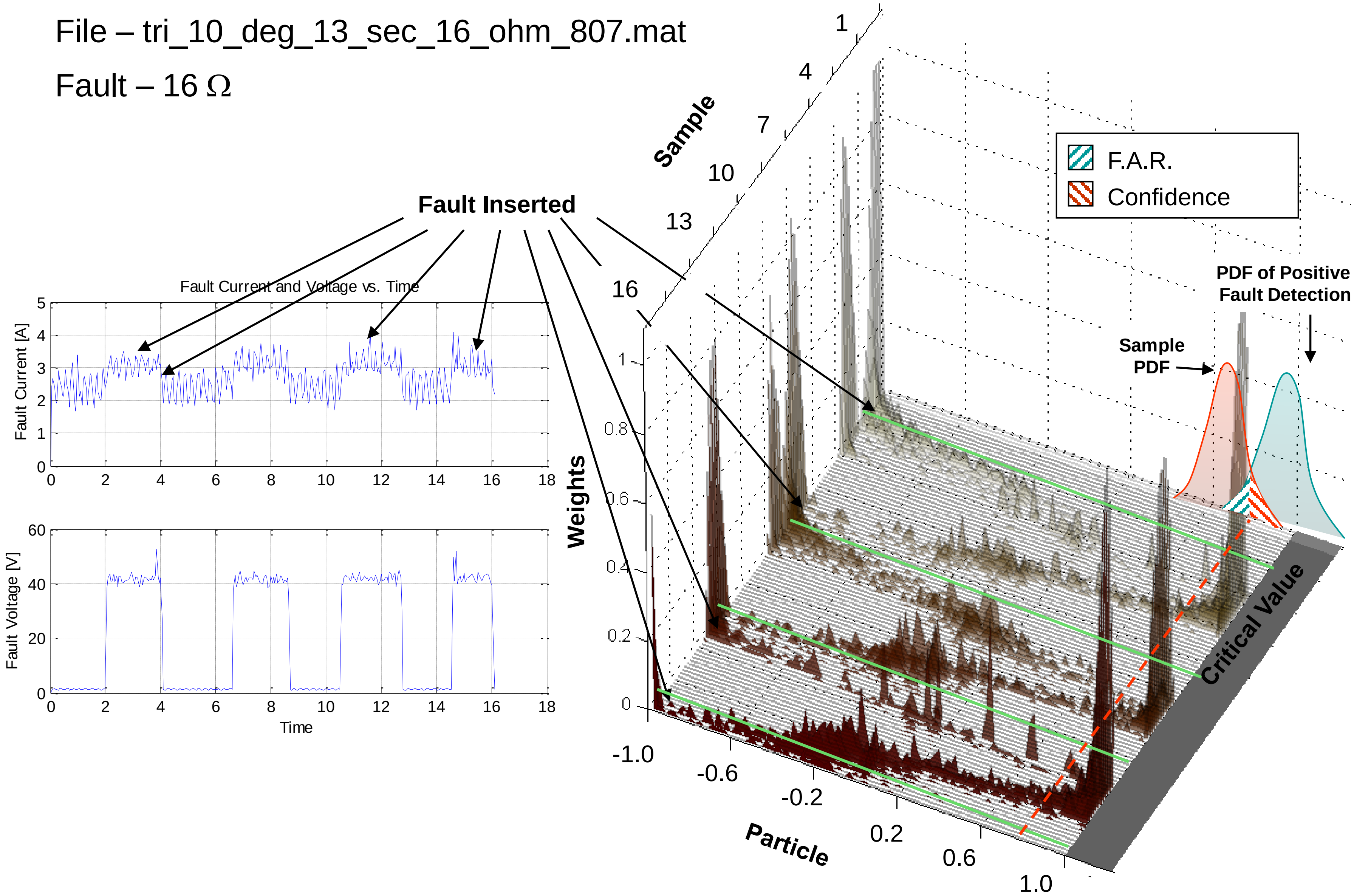
Detection Results: Type I Error = 5%



Particle Filter Results – Fault

File – tri_10_deg_13_sec_16_ohm_807.mat

Fault – 16 Ω



Failure Prognosis

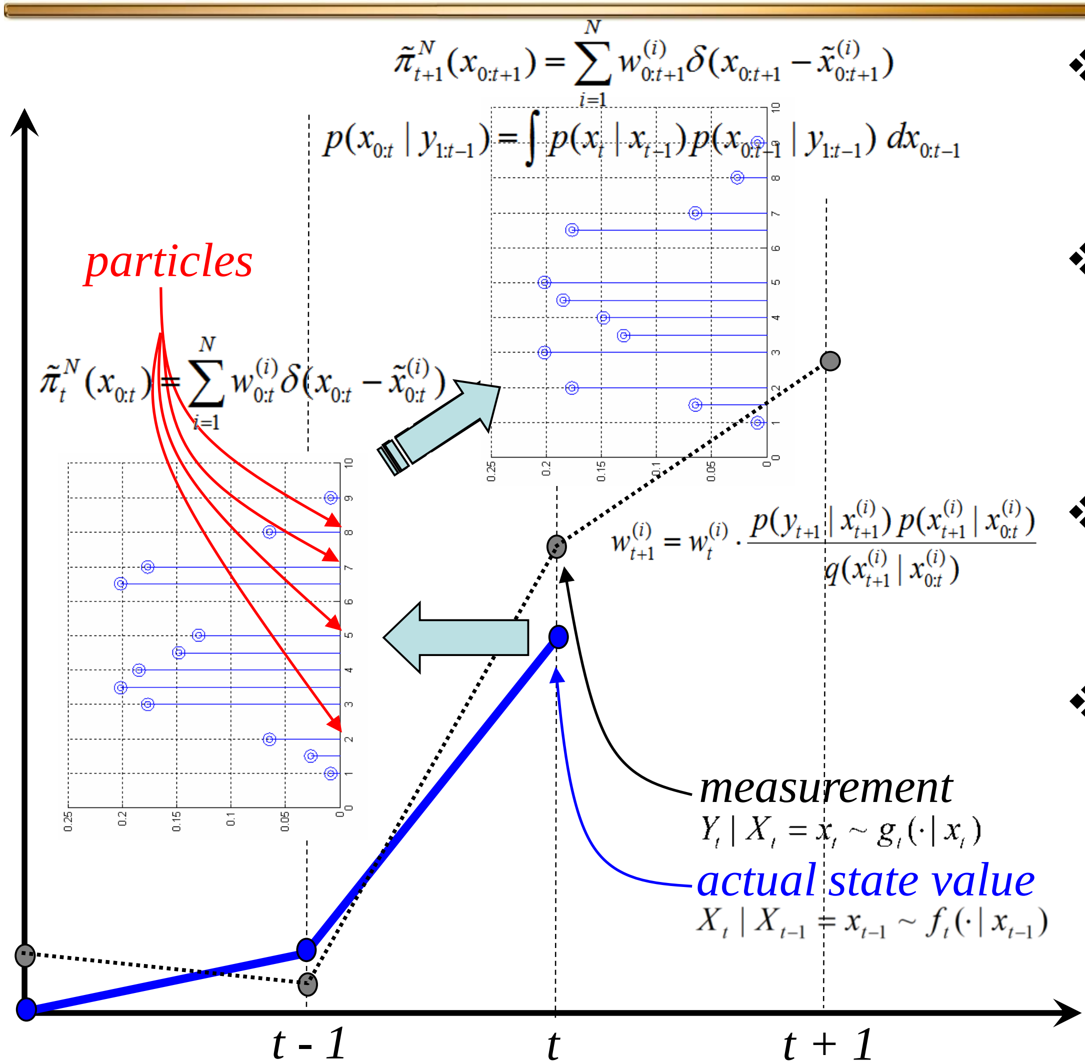
- Objective
  - Estimation of Remaining Useful Life of a failing component/system
  - Determine time window over which optimal maintenance or corrective action must be performed without compromising the system's operational integrity
- Prognosis vs. Trending
- Prediction in the presence of uncertainty
- Prognosis from "birth" or "usage-based" vs. "heatlt-based" or, real-time prognosis
- The customer base:
  - The maintainer
  - The fleet commander/process manager
  - The designer

Prognosis: A Model-based and Measurements Approach

Proposed Approach:

Utility of a fault model, a feature vs. fault dimension mapping and a particle filtering framework (Bayesian estimation) for long-term prediction

Particle Filtering Algorithm



❖ **Particle**: Duple $\{w_t^{(i)}, x_{0:t}^{(i)}\}$, being $x_{0:t}^{(i)}$ a realization of process state *pdf*.

❖ Every particle is associated with an scalar $w_t^{(i)}$, namely the weight

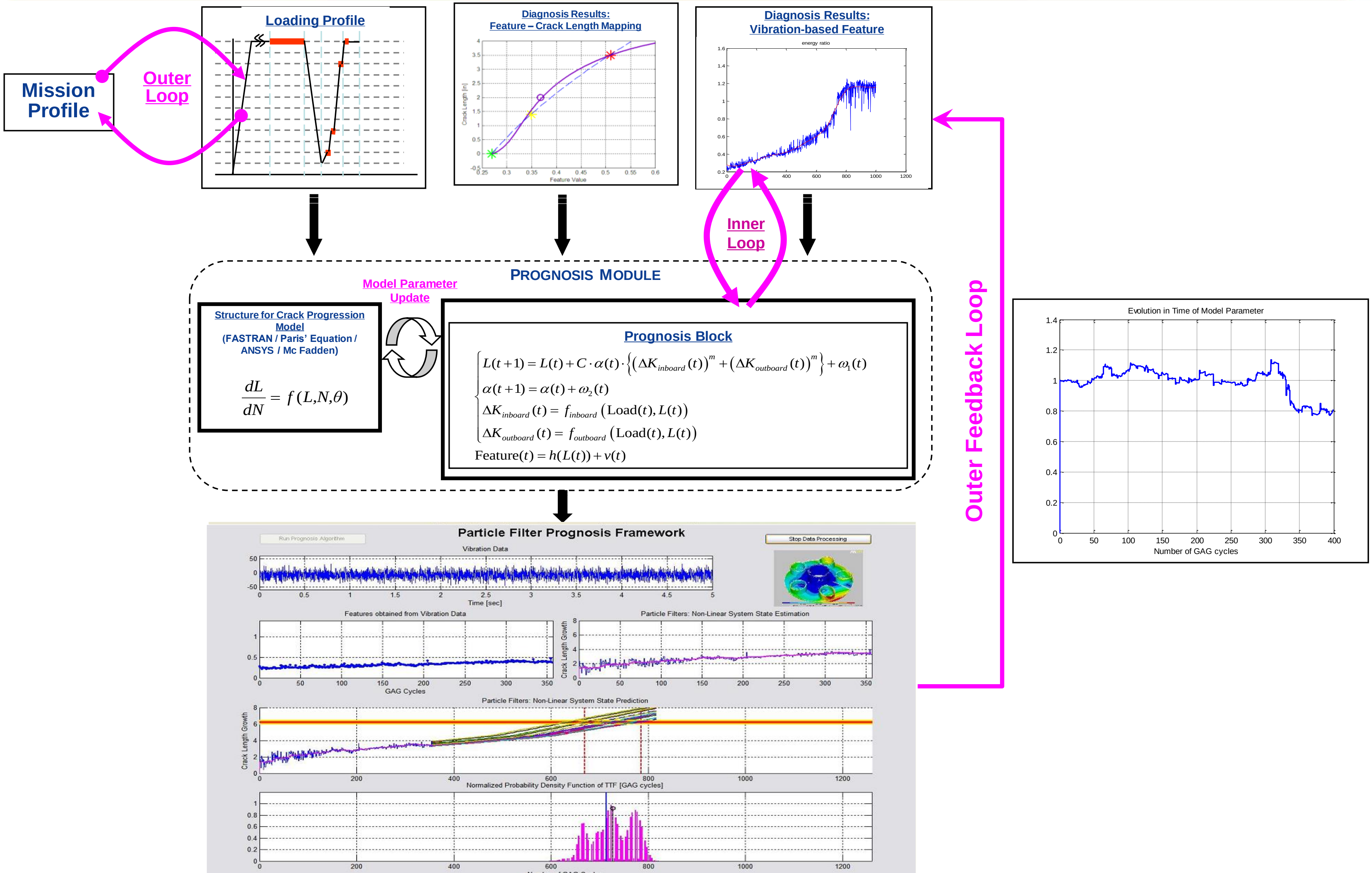
- Sampled version of the PDF

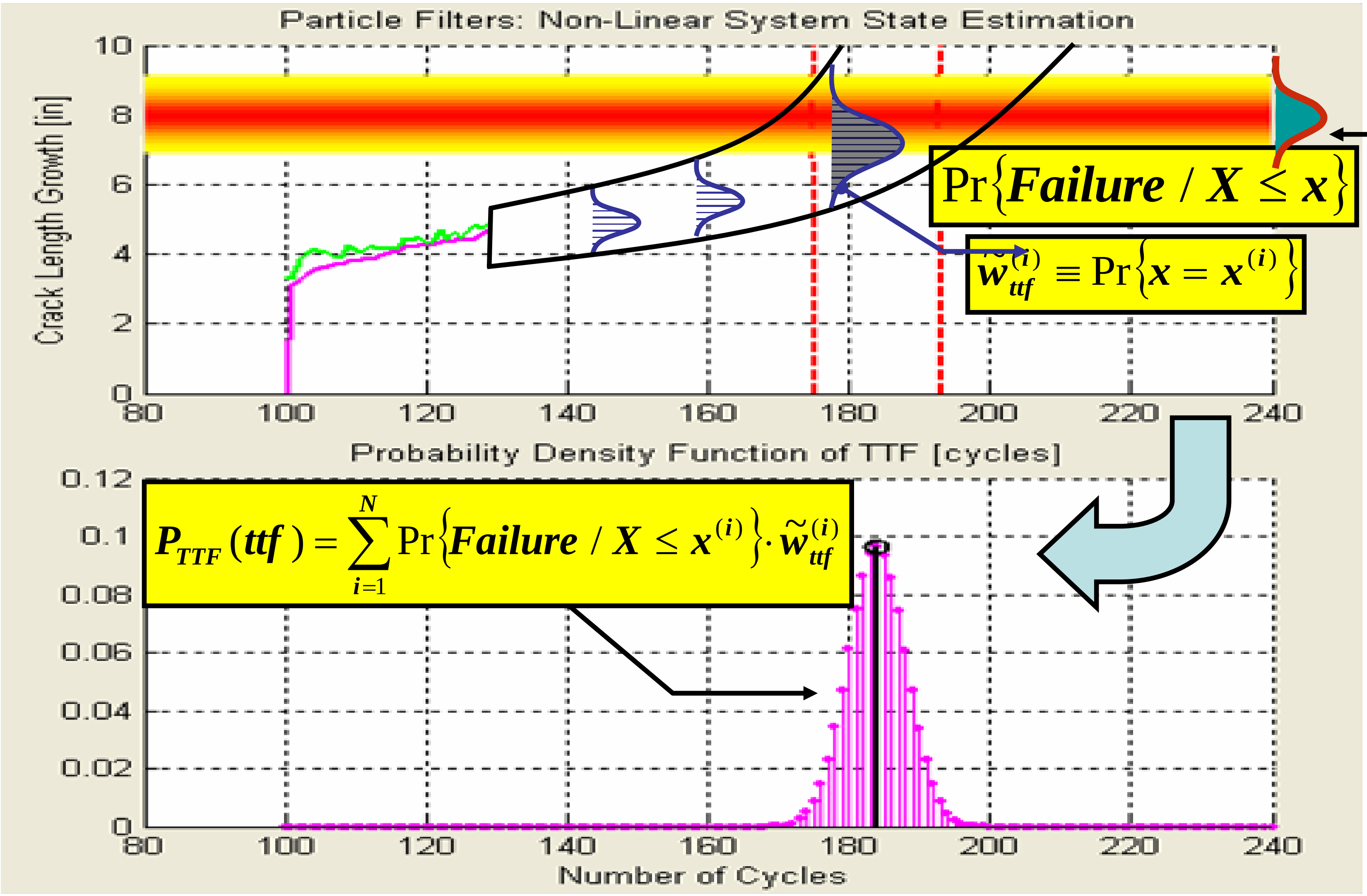
❖ We only need to study the propagation of particles in time!

❖ **Steps**:

- Predict the "*a priori*" PDF, using the model
- Update parameters, given the new measurement

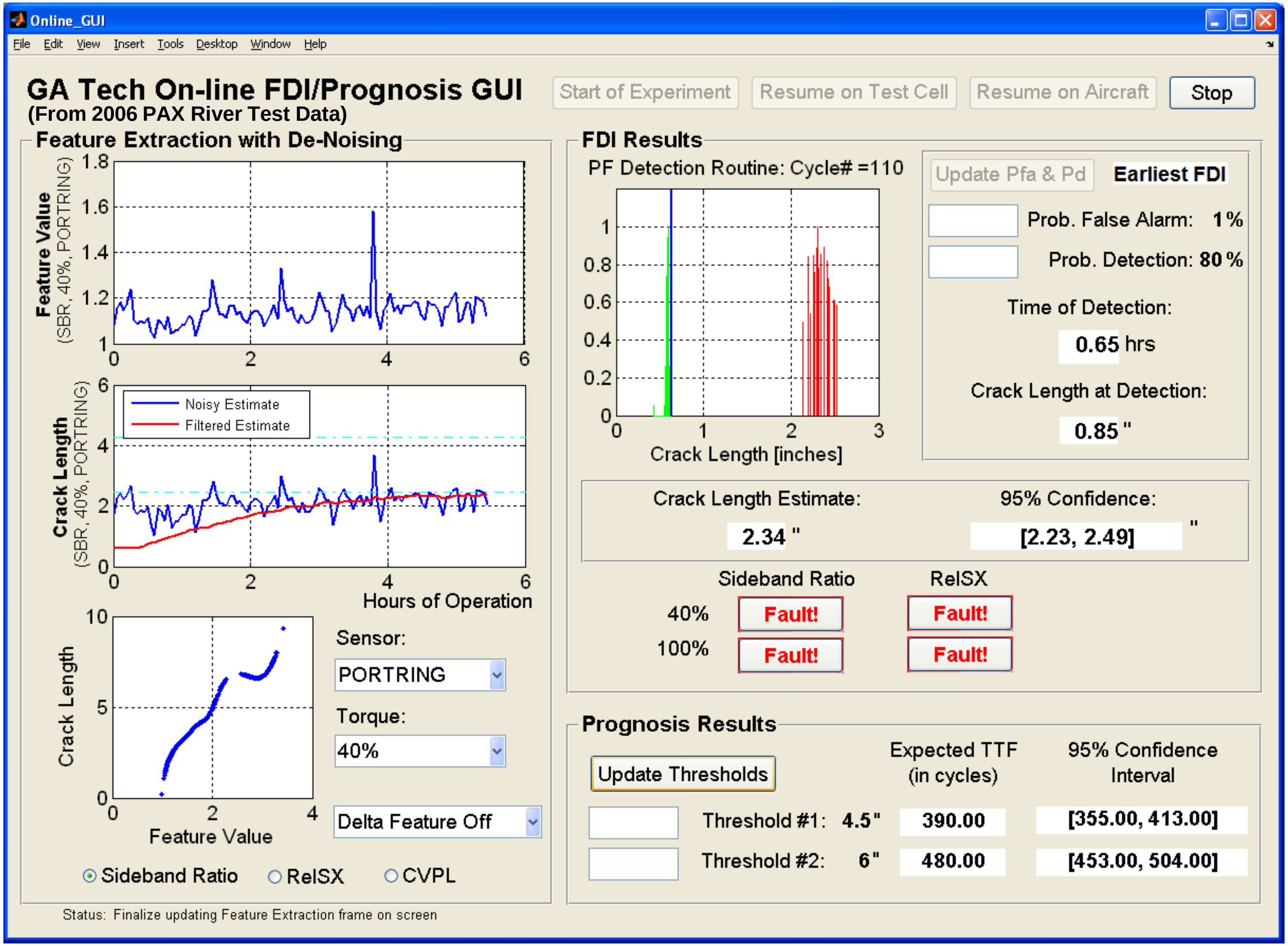
Prognosis Architecture





Particle Filters: Non-Linear System State Estimation
Crack Length Growth [in]
$\Pr\{Failure / X \le x\}$
$\tilde{w}_{ttf}^{(i)} \equiv \Pr\{x = x^{(i)}\}$
Probability Density Function of TTF [cycles]
$P_{TTF}(ttf) = \sum_{i=1}^{N} \Pr\{Failure / X \le x^{(i)}\} \cdot \tilde{w}_{ttf}^{(i)}$
Number of Cycles

On-board Experiment



.NET Implementation in Diagnosis/Prognosis

.NET Component

Diagnosis.dll

.NET Component

Prognosis.dll

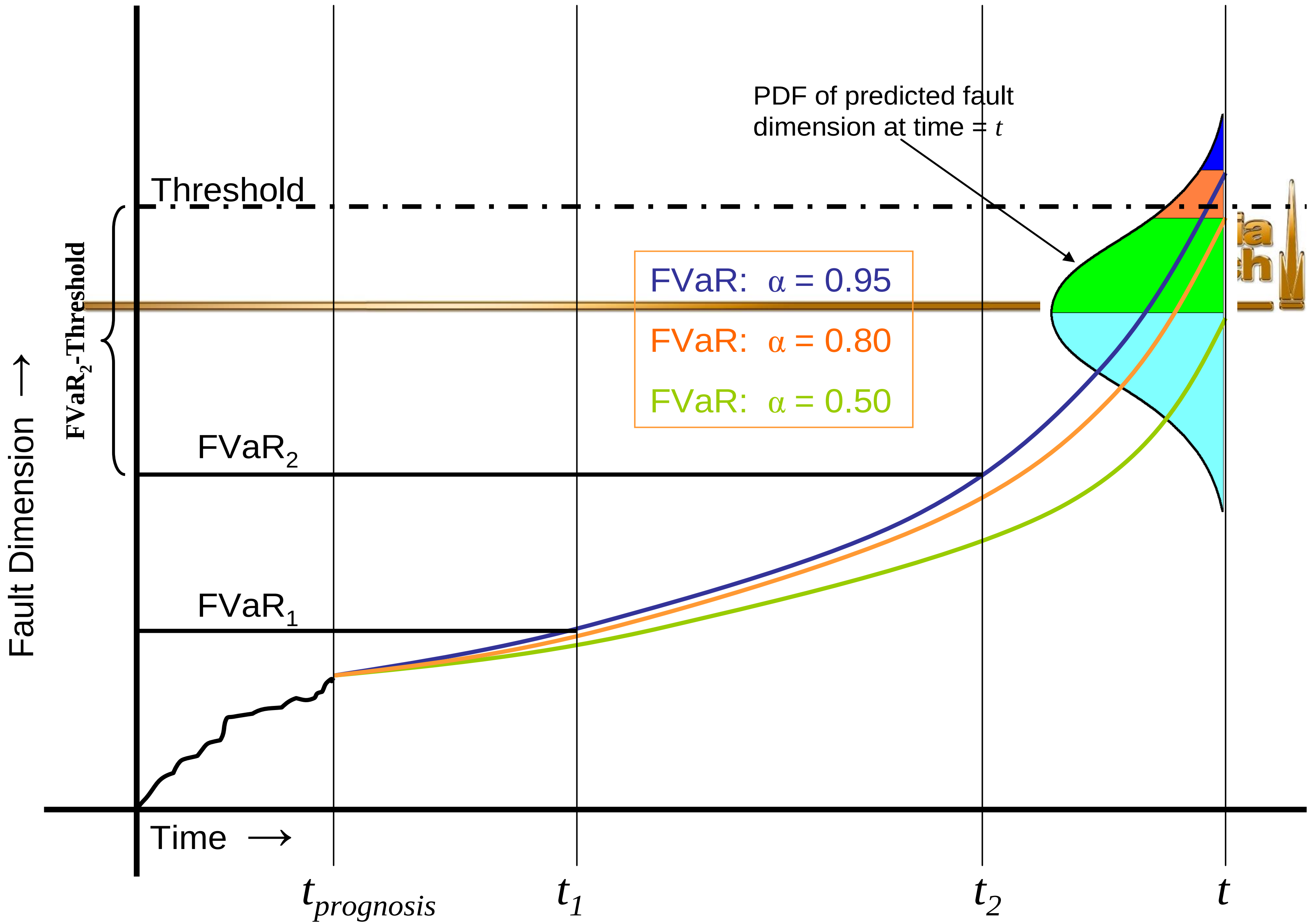
RUN

Normal condition given by a Normal distribution

Real time condition obtained by fault detection

Risk and Confidence

Fault Value at Risk (FVaR)



An Example:

Consider an a/c component fault

The system has 12 hr FVaR of fault dimension at 95% confidence level means:

> We are 95% confident that a change in the fault dimension (damage) in 12 hrs will not result in an increase of 10 units in the fault dimension.

Or:

> There is a 5% confidence level that damage will increase by 10 units or more in 12 hrs.

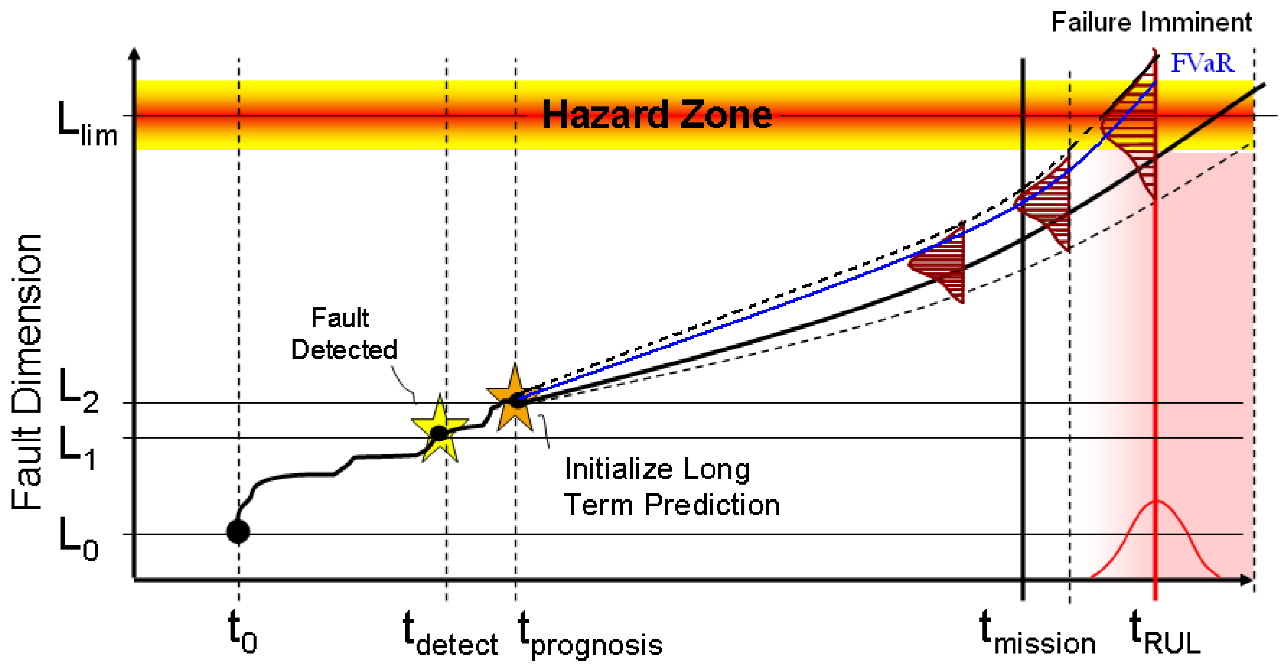
Risk Control

- Change risk profile through proactive maintenance and upgrade
- Take corrective action with acceptable risk
    - Quantify <u>risk</u> and <u>uncertainty</u>
    - Essential link between failure prognosis and reconfigurable control

Confidence: Necessary ingredient for action

$$\alpha = \int_{-\infty}^{FVaR(t_{future},t_{prognosis})} \hat{p}(x_{t_{future}} \mid y_{t_{prognosis}})dx_{t_{future}}$$

α: degree of confidence specified by the user

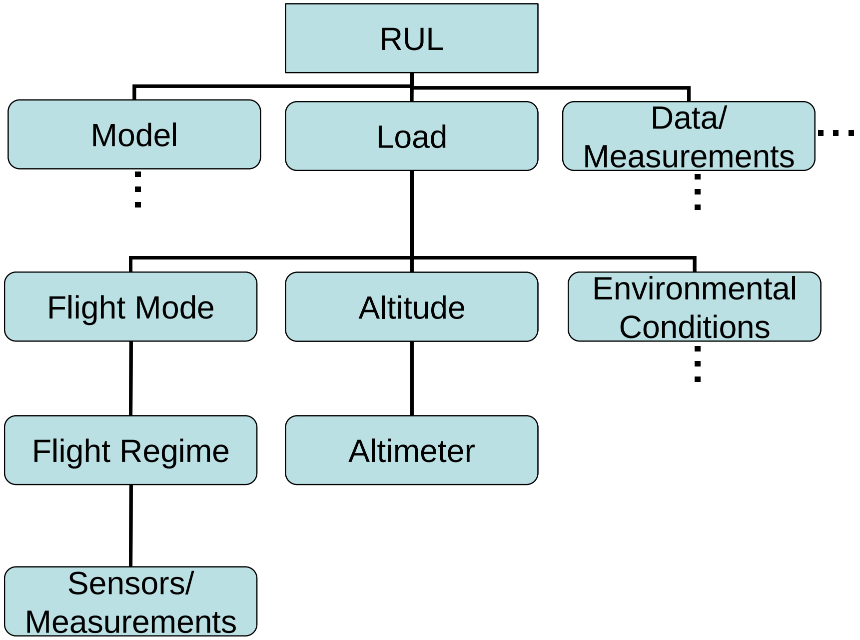


FVaR predicted from time $t_{prognosis}$

Uncertainty Representation and Management

Sources of uncertainty – the uncertainty tree

- A graphical depiction of the variable dependence in uncertainty analysis.
- Technique suitable for combining multiple sources of uncertainty for a single variable.
- Useful also for design of experiments.
- A tool for relating uncertainties: root-sum-square.

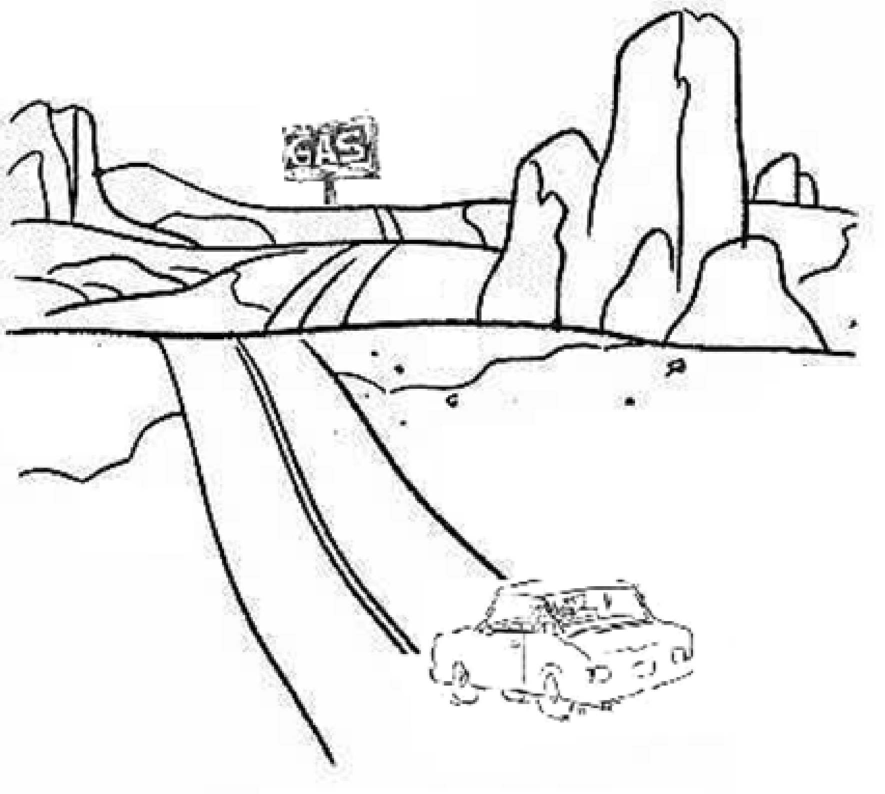


Fault – Tolerant Control

(Fault Mitigation, Fault Accommodation, Reconfigurable Control)

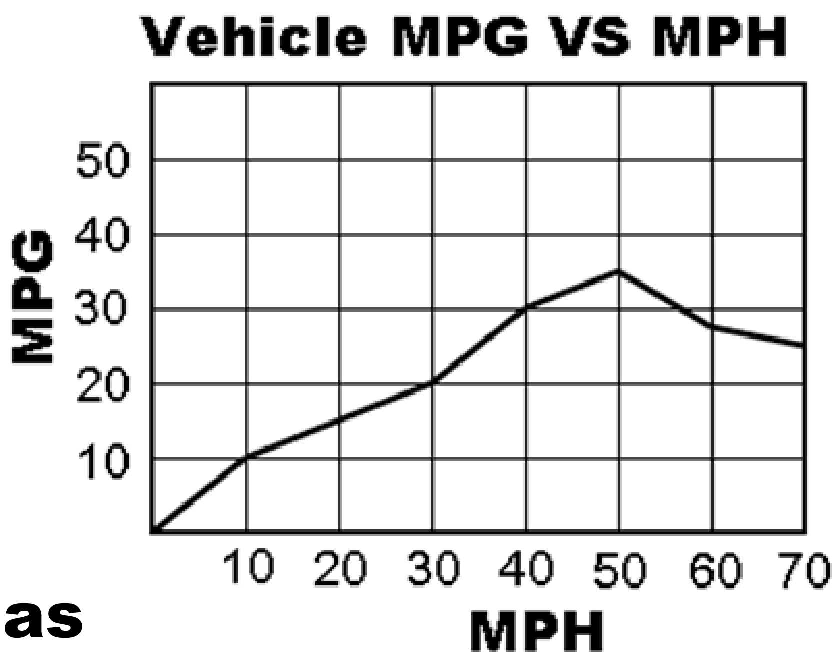
The Caveat: With Prognostic Information

The Link between PHM and Control



We have 1 GAL left in the tank

THE NEAREST STATION IS
30 MI AWAY!!!



Can We Make It To The Gas Station?

Motivation
Previous and Current Initiatives

Timeline

1980's — 1990s — 2000s — Present — Future

Military fixed wing aircraft programs

NASA Aviation Safety Program

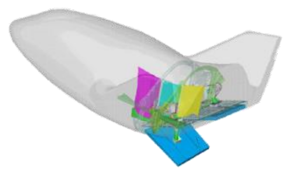
Integrated Resilient Aircraft Control

Integrated Vehicle Health Management

- Prognostic Center of Excellence
- Integrated Flight Deck Project
- Aircraft Aging Durability Project
- Fundamental Aeronautics
- Exploration Systems Mission Directorate
- Joint Army Navy NASA Air Force Initiative

The Control Architecture

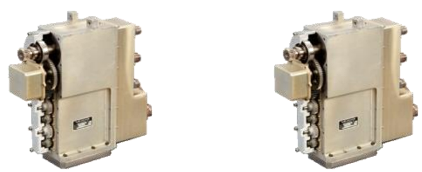
High Level



Vehicle

- System level
- Monitors mission objectives
- Mission adaptation (eg. path replanning)

Mid Level



Control Surface Actuators

- Sub-system level
- Redistributes control authority
- Ensures vehicle stability

Low Level

Brushless-DC Motors

- Component level
- Reconfigures set-points
- Ensures minimum performance

Motivation

Previous and Current Initiatives

Timeline

1980's 1990s 2000s Present Future

Collective goals:

- Improve safety and reliability
  - Early fault detection
  - Failure prediction
  - Fault mitigation
- Reduce maintenance costs
- Reduce system down-time

ter of

Deck

rability

ronautics

ems

ate

NASA

e

Health Management

The Control Architecture
Introduction

– The Big Question –

Can remaining useful life (RUL) be increased by reducing performance?

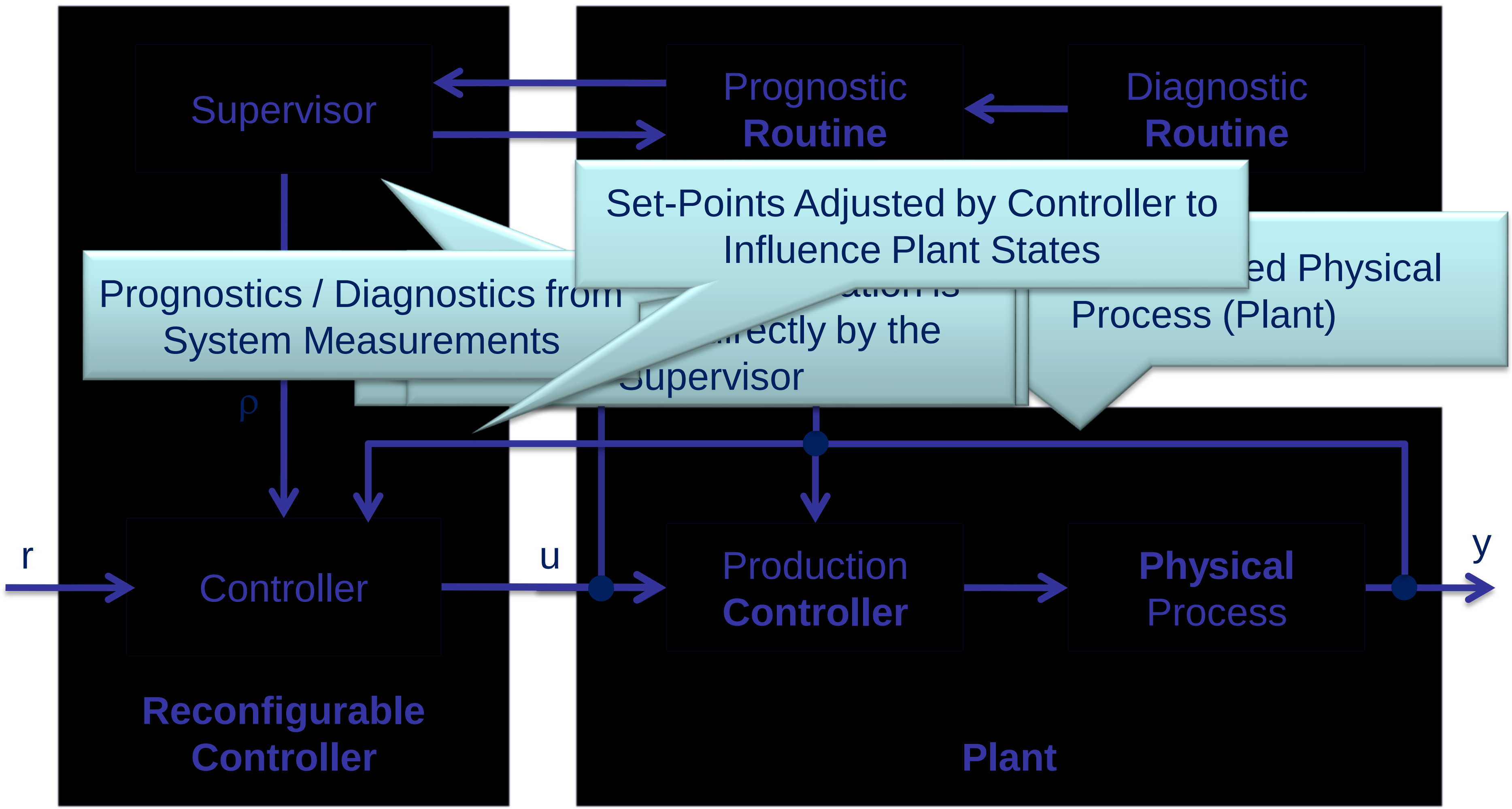
Health vs. Performance

- How is RUL related to performance?
- How can performance be reduced?
- What are the factors?
  - Application
  - Operating conditions

Architecture Dependent

The Control Architecture

Reconfigurable Control Architecture



The Control Architecture

Optimization Criteria for MPC

Health vs. Performance

Assumptions

RUL Related States vs. Tracking Error

Adaptation parameter ρ adjusts cost

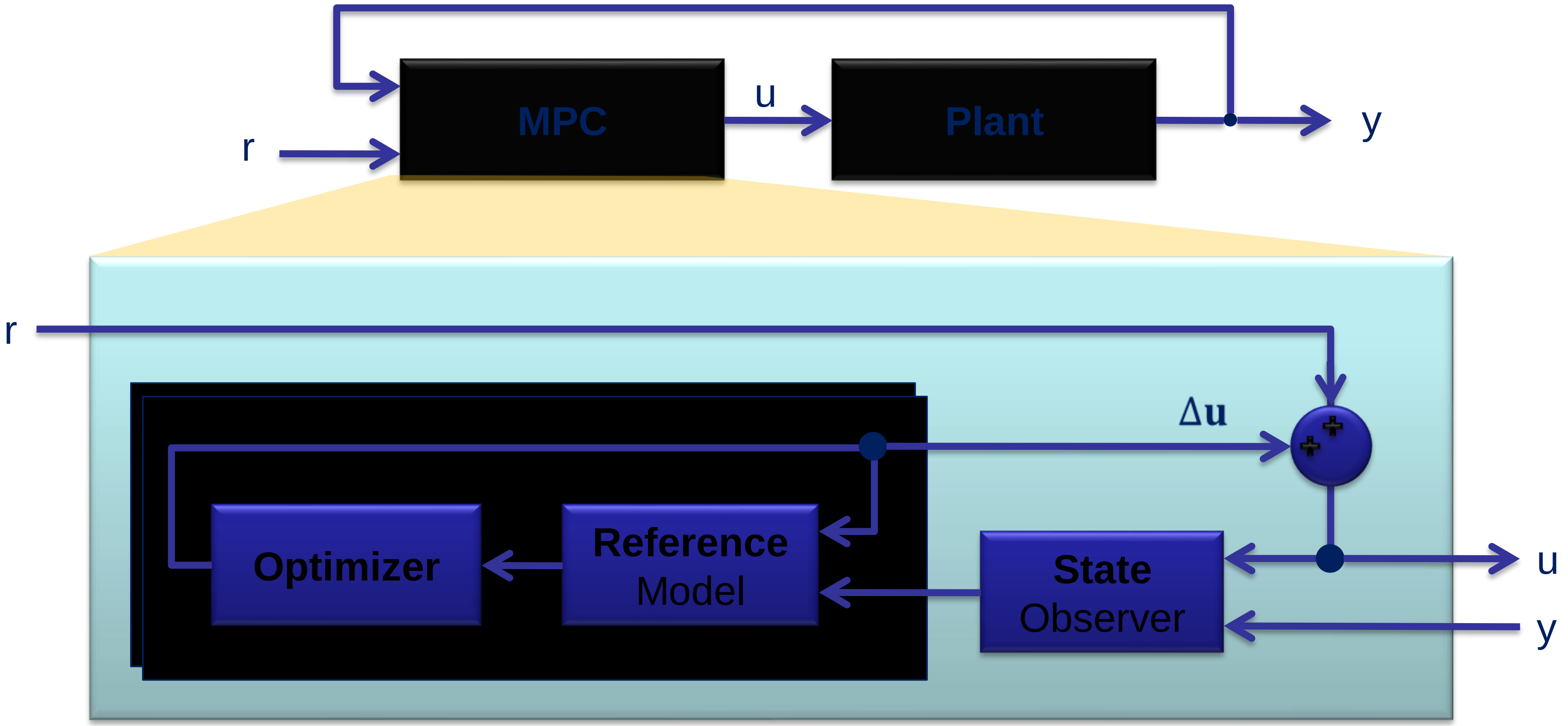
- The cost function:

$$J = \min_{\Delta \mathbf{u}} \int_{t_0}^{t_0+T} [(\mathbf{x} - \mathbf{x}^\star)^\top (\rho \mathbf{Q})(\mathbf{x} - \mathbf{x}^\star) + \Delta \mathbf{u}^\top \mathbf{R} \Delta \mathbf{u}]\, dt$$

- Subject to the constraints,

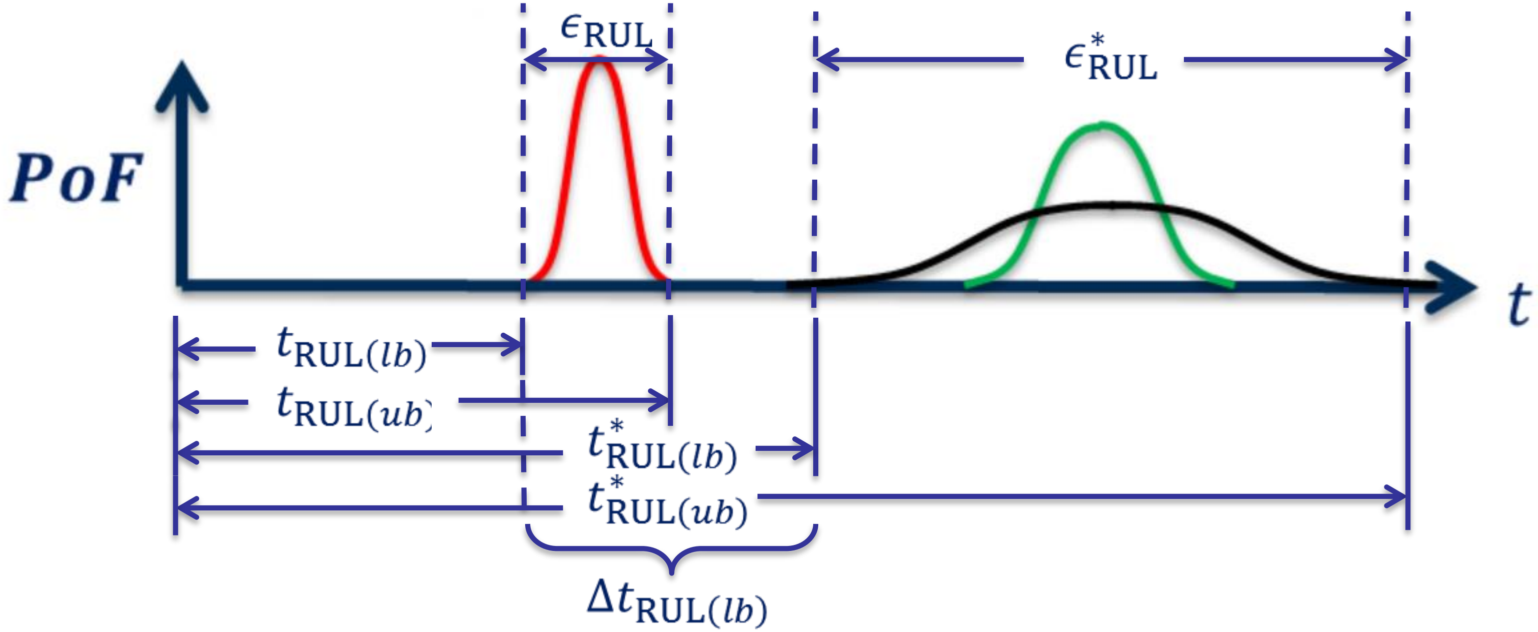
$$\begin{cases} \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}(t) \leq \Delta \mathbf{u}_{\max} \\ \mathbf{u}_{\min} \leq \mathbf{u}(t) \leq \mathbf{u}_{\max} \end{cases}$$

Composite system – Plant coupled with MPC controller



Stability and Uncertainty Analysis

Measurements



Definition (RUL Gain)

The resulting RUL gained after reconfiguration,

$$\Delta t_{\mathrm{RUL}(lb)} \triangleq t^\star_{\mathrm{RUL}(lb)} - t_{\mathrm{RUL}(lb)}$$

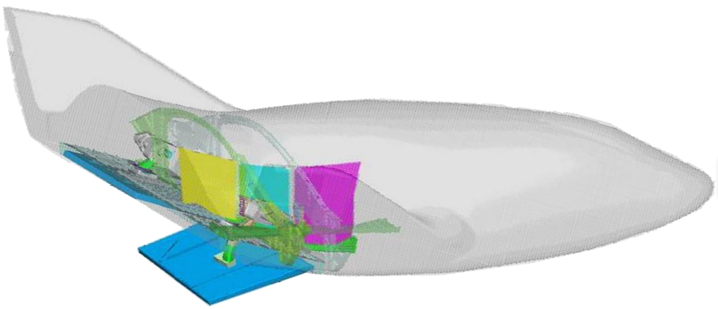
Definition (Confidence Interval)

The confidence interval width of the reconfigured RUL,

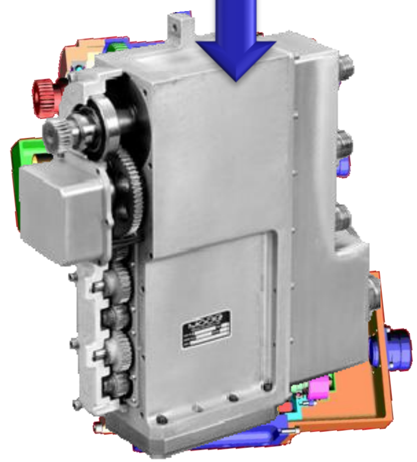
$$\epsilon^*_{\mathrm{RUL}} \triangleq t^\star_{\mathrm{RUL}(ub)} - t^\star_{\mathrm{RUL}(lb)}$$

Example Application

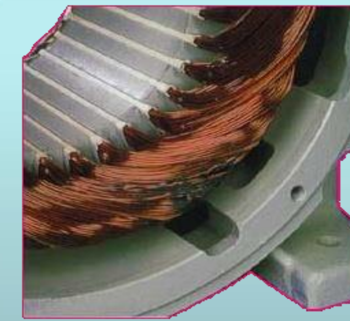
Electro-Mechanical Actuator



System
X38 Crew Re-entry Vehicle



Sub-System
Electro-Mechanical Actuator (EMA)

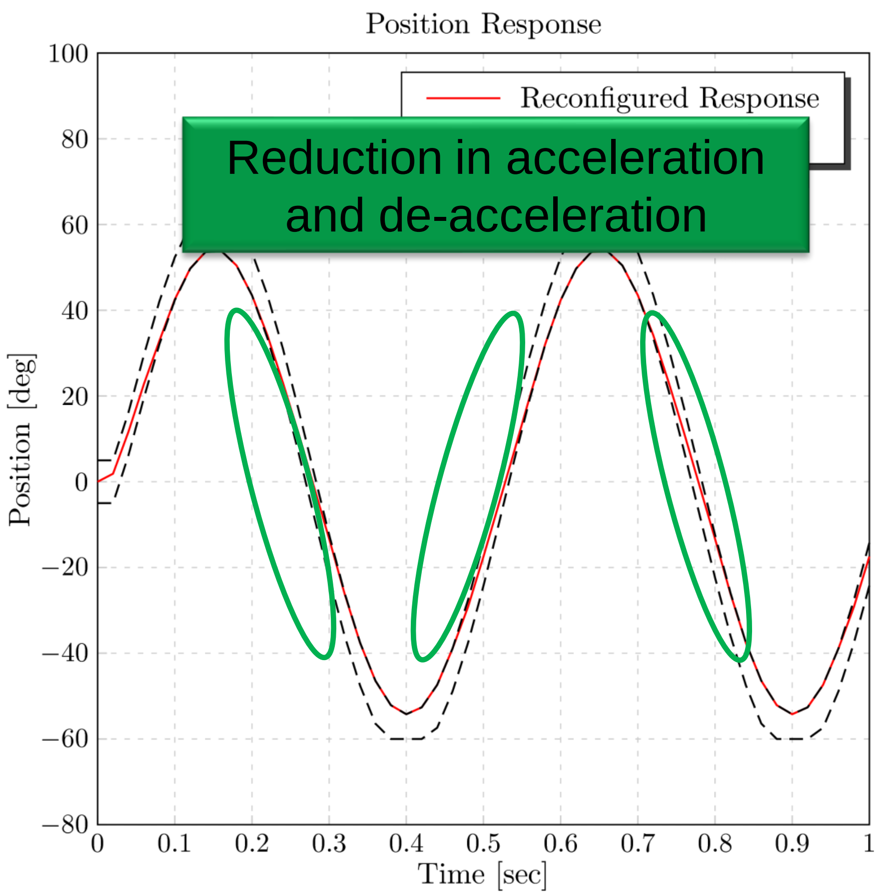


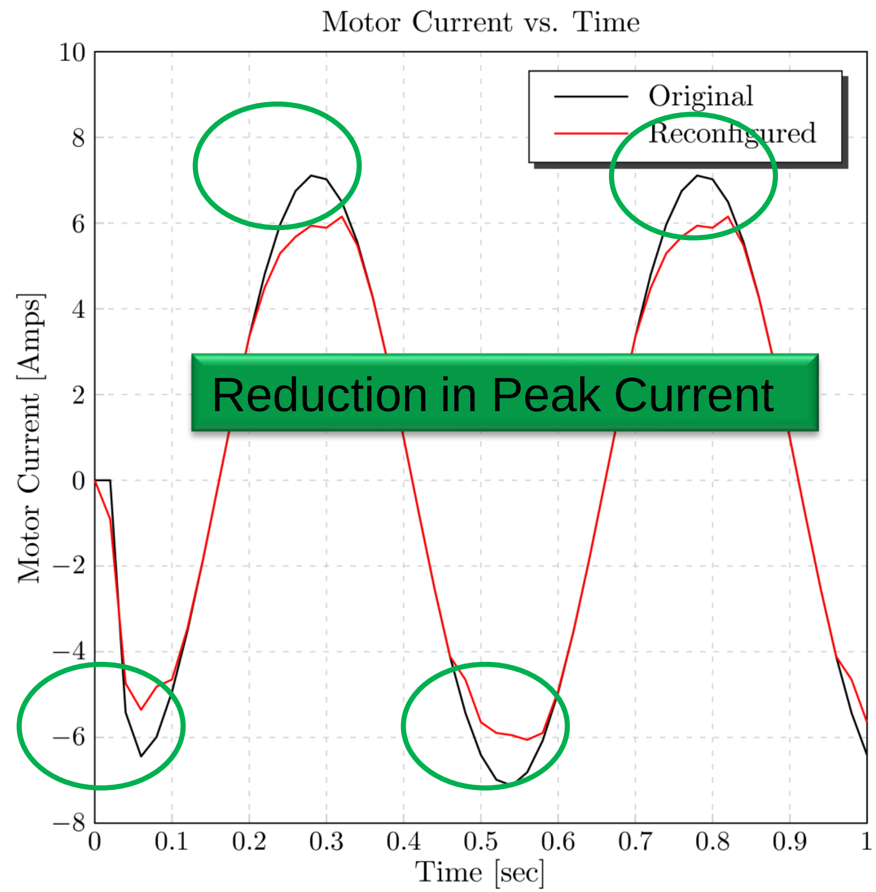
Fault Mode
Winding Insulation

Component
Brushless DC Motor

Example Application
Reconfiguration Feasibility

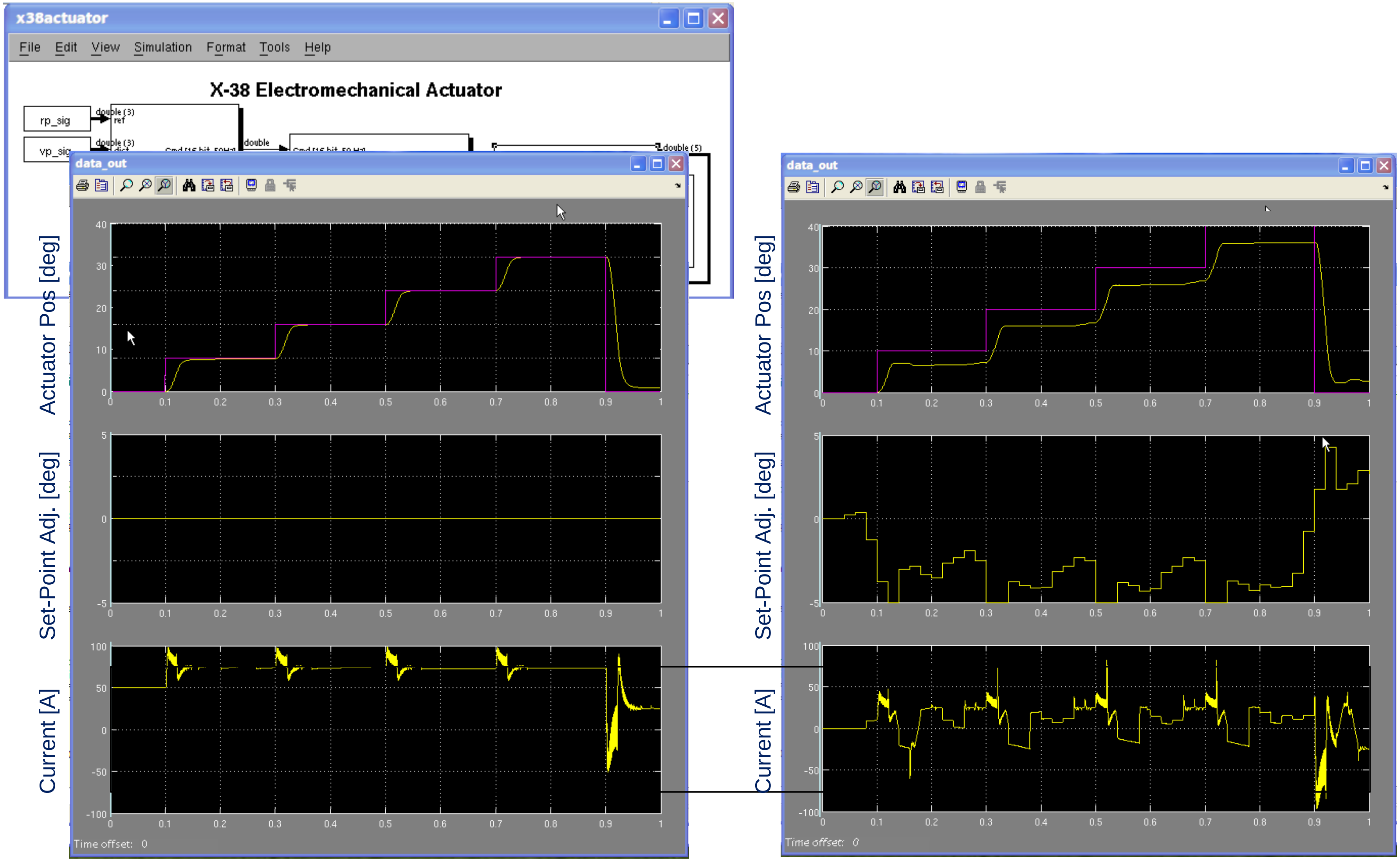
- Consider worst case: $\rho \gg 1, \mathbf{Q} = \mathrm{diag}([1\ 0\ 0\ 0\ 0])$ and $\mathbf{R} = 1$.
- Deterministic with no external load ($\mathbf{v} \equiv \mathbf{0}$).
- Simulated case: $p = 5$ and $\eta = 0.19$ (implies feasibility)





Example Application

Non-Linear System / Demonstrate Feasibility



How to Measure “Success”

- Software-in-the-loop simulation of all constituent modules of the prognostics-enhanced reconfigurable control strategy.
- Hardware-in-the-loop simulation/demonstration
- Testbed demonstration
- TRL-moving up!
- Other

Looking into the Crystal Ball

- Key integration issues: The Human-Automation interface; from reliability to control and design
- Emphasis on Design for Autonomy
- Synergy between the PHM designer, the system designer and the control engineer
- High-Confidence Systems!
- More Success Stories: Convincing the non-believers