Sampling-Based Fault Diagnosis and Change Detection for Stochastic Dynamical Systems

Ioannis A. Raptis and Vincent Kwao

North Carolina A&T State University
Department of Electrical and Computer Engineering
Autonomous Robotic Systems Laboratory

February 23, 2024

Outline

- Introduction
- Nonlinear State Estimation
- The Fault Sensitive Filter
- Actuator Fault Diagnosis of a Quadrotor
- Result
- Conclusion

Introduction [1/6]: Motivation

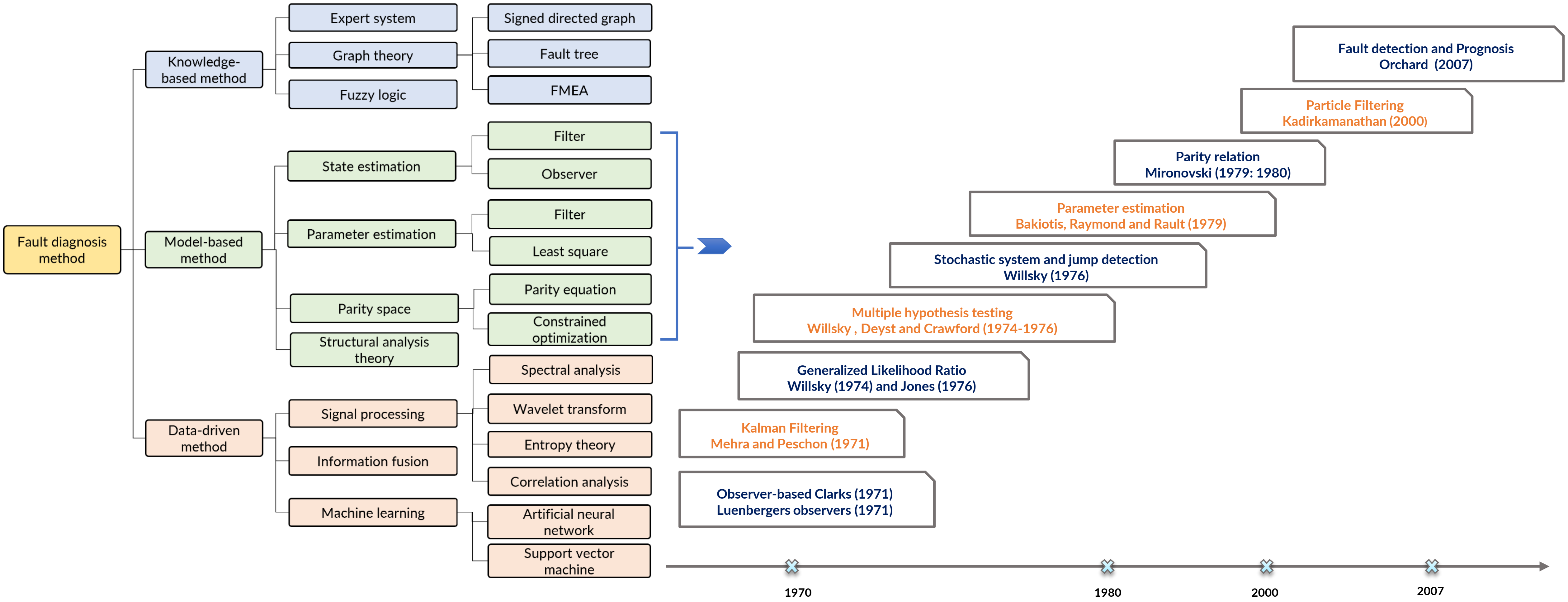
- A fault, is simply a deviation from expected operation.
- Aircraft, like all other dynamic systems are susceptible to faults that may lead to catastrophic failures.
- Aircraft faults can be categorized as sensor, actuator or, structural faults.
- Regardless of the categorization, a timely diagnosis of a fault improves the chances of averting down-times, breakdowns, and catastrophic failures.

Introduction [2/6]: Background

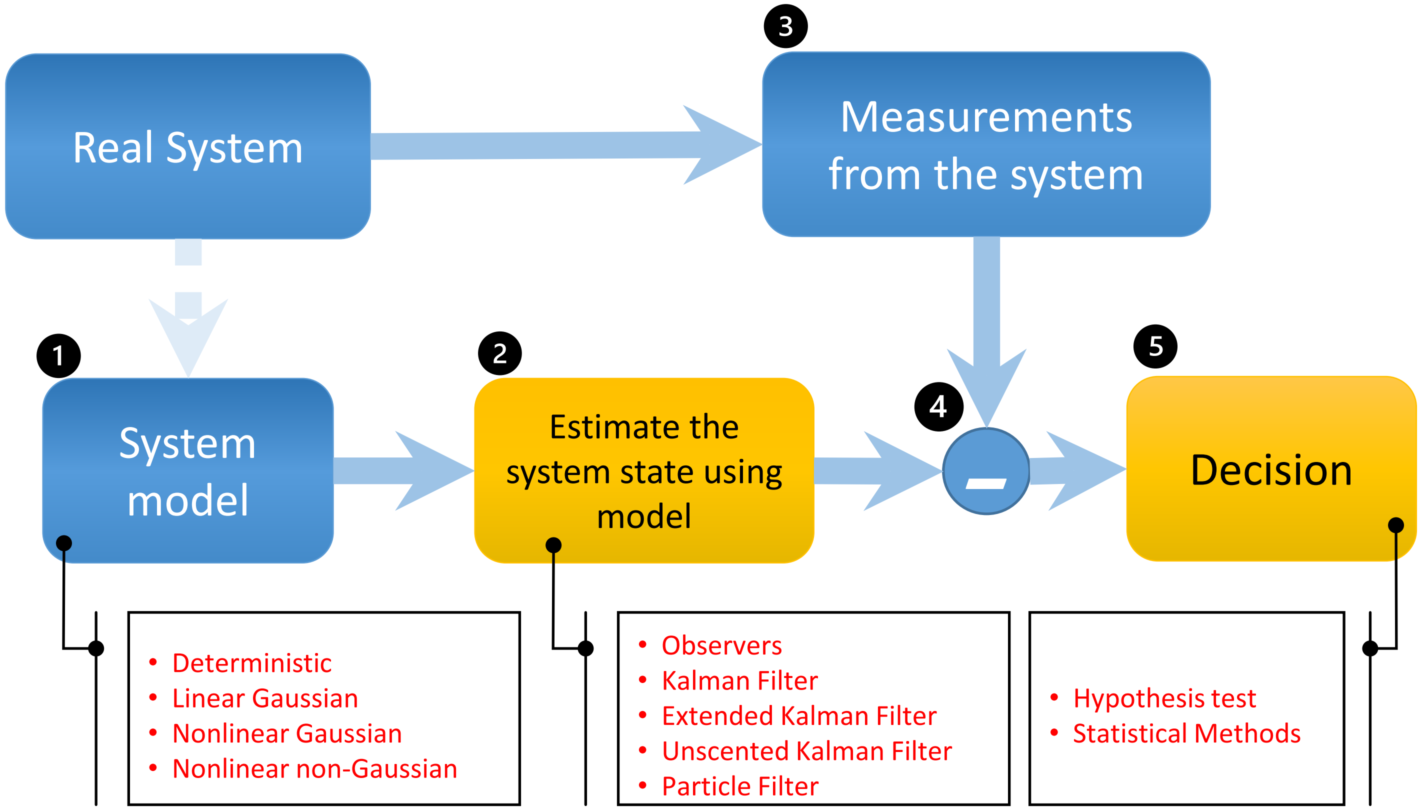
- Fault Diagnosis (FD) algorithms replace hardware-based, cost-ineffective sensory redundancy with algorithmic, software-driven solutions.
- Model-based FD methods are capable of modeling intricate systems and quantifying fault severity. They offer a range of established solutions to enhance system reliability.
- Sampling-based methods are well-suited for state estimation of high-dimensional systems, particularly for problems that are difficult of impossible to solve analytically.

Develop a sampling-based, model-based method for FD that can handle the analytically intractable nonlinear, non-Gaussian aircraft dynamics with algorithmic implementation that is straightforward and well suited for seamless integration into embedded flight control systems.

Introduction [3/6]: Taxonomy & History

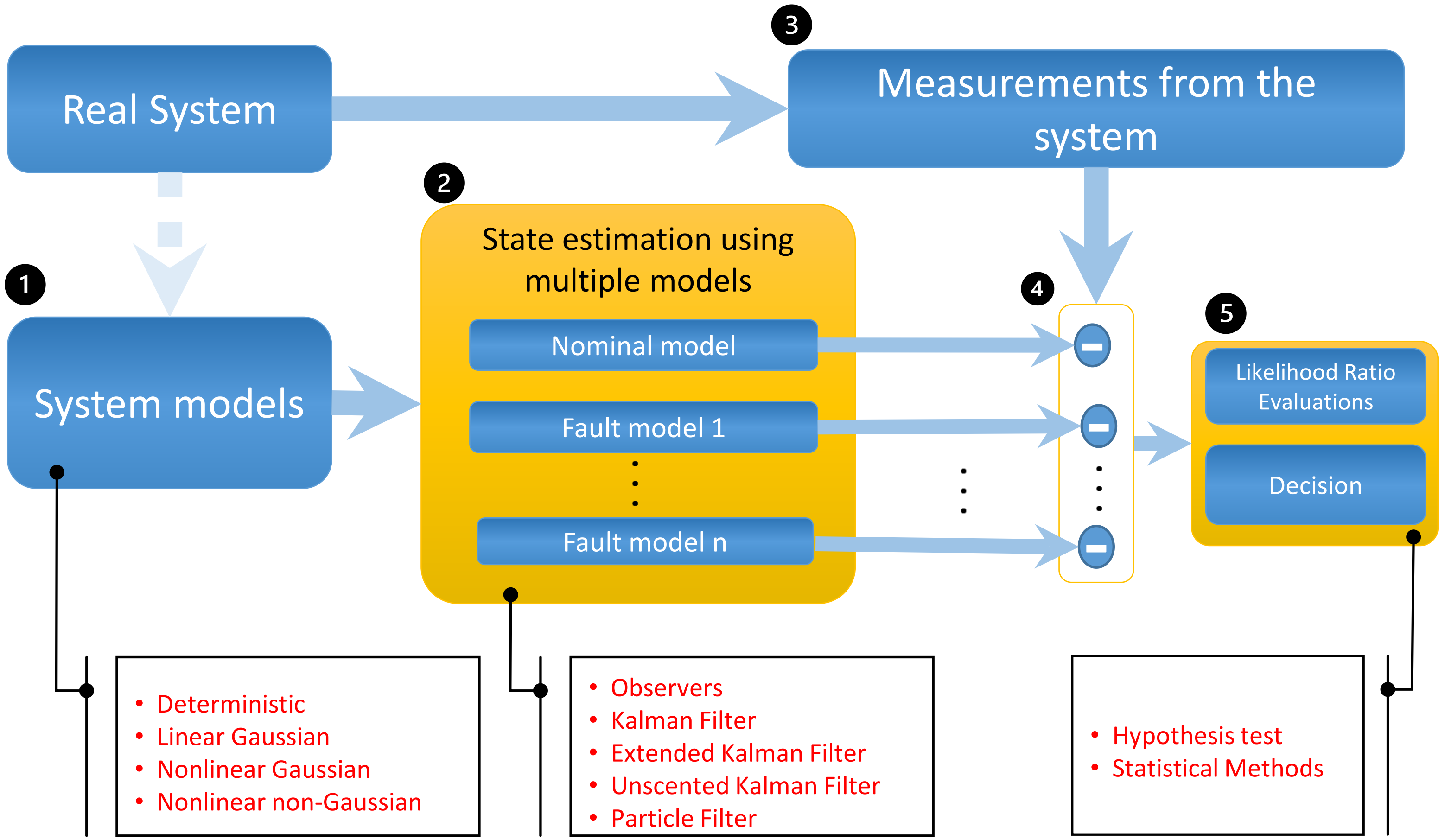


Introduction [4/6] : Model Based Detection



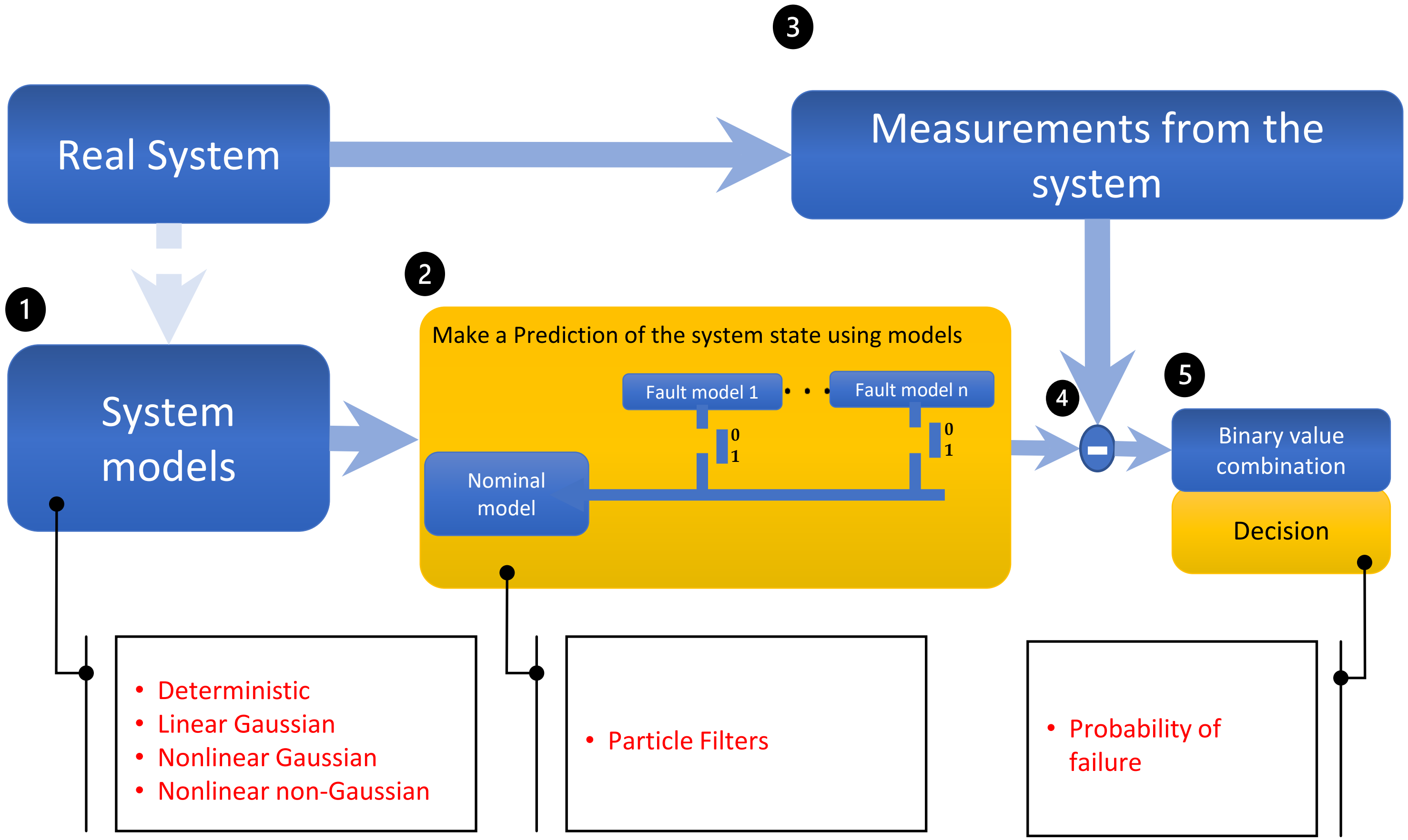
Identifying deviations from expected performance in a dynamic system.

Introduction[5/6] : Multiple Model Framework (MMF)



Detecting and Isolating one out of potential faults in a dynamic system.

Introduction[6/6] : An Alternative Formulation



Mathematical Problem Formulation

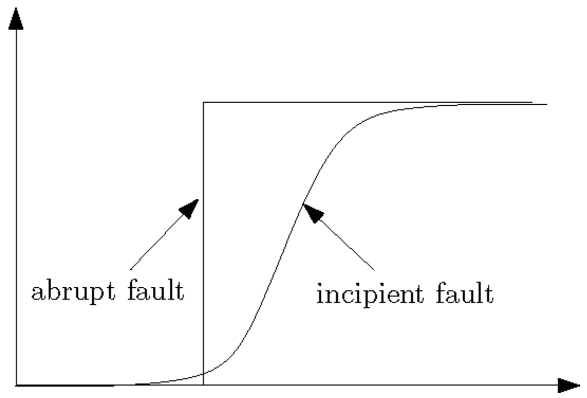
- Consider a nonlinear dynamic system S described by:

$$S : \begin{cases} x(k) = \underbrace{f(x(k-1), u(k-1))}_{\text{nominal dynamics}} + \underbrace{\sum_{j=1}^{M} \beta(k-k_0^j) \cdot g^j(x(k-1), u(k-1))}_{\text{fault dynamics}} + \underbrace{v(k-1)}_{\text{model uncertainty noise}} \\ z(k) = h(x(k)) + \omega(k) \end{cases}$$

$$\begin{aligned} k \in \mathbb{Z}^+ &: \text{Time index.} \\ x(k) \in \mathbb{R}^{n_x} &: \text{State vector.} \\ u(k) \in \mathbb{R}^{n_u} &: \text{Control input vector.} \\ z(k) \in \mathbb{R}^{n_z} &: \text{Observation measurement vector.} \end{aligned}$$

$v(k) \in \mathbb{R}^{n_x}$ and $\omega(k) \in \mathbb{R}^{n_z}$ are the system process and measurement noise, respectively.

$$\beta(k-k_0^j) = \begin{cases} 0 & \text{if} \quad k < k_0^j \quad \text{absence} \\ \underbrace{1}_{\text{abrupt}} \quad \text{or} \quad \underbrace{1 - c^{-(k-k_0)}, c > 1}_{\text{incipient}} & \text{if} \quad k \geq k_0^j \quad \text{presence} \end{cases}$$



- **Goal**: Generate a statistical characterization of the fault modes that can trigger fault alarms.

From difference equations to FD

- We focus on the benchmark state-transition model under consideration:

$$x(k) = f\left(x(k-1), u(k-1)\right) + \sum_{j=1}^{M} \beta(k-k_0^j) \cdot g^j(x(k-1), u(k-1)) + v(k-1).$$

- We want to estimate the values of the step functions $\{\beta(k-k_0^j)\}_{j=1}^M$ when a new measurement $z(k)$ becomes available.

- Define $\mathbb{B}(k) = [\beta(k-k_0^1), \ldots, \beta(k-k_0^M)]$. We will call it the change vector. A more compact way of writing the dynamics is the following

$$x(k) = f\left(x(k-1), u(k-1)\right) + G\Big(\mathbb{B}(k), u(k-1)\Big) + v(k-1).$$

- In our context, FD differs from classical model selection by certain key characteristics:

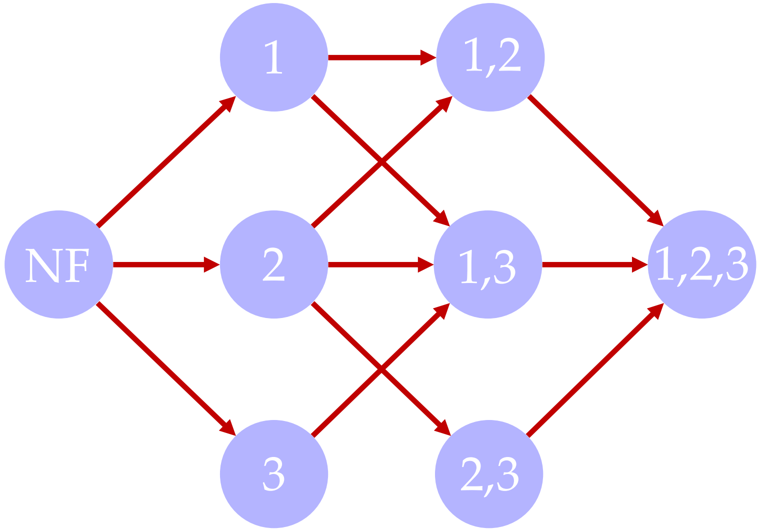
  1. Faults are persistent; the system cannot recover from them. This implies that $\beta(k-k_0^j) = 1$ for every $k \geq k_0^j$.
  2. The model inducing the change is unknown and, hence, the transition probability $p(\beta(k) = 1|\beta(k-1) = 0)$ are both unspecified.
  3. Multiple faults may occur at various time instances.

A Discrete-Event Viewpoint

If our archetypal dynamical model is viewed as a discrete-event system, with M being the number of fault modes, it will consist of 2^M states. It is the number of all k-combinations for all k, that is,

$$\sum_{k=0}^{M} \binom{M}{k} = 2^M,$$

where $\binom{M}{0}$ corresponds to the "healthy" (no fault state). For example, for $M = 3$, we will have $2^M = 8$ states and the following state transition diagram:



A Decision Theory for Multiple Models Viewpoint

- Addressing the detection of a specific fault mode among multiple possibilities, or confirming the absence of faults, involves analyzing $m + 1$ models, labeled by $j = 0, 1, \ldots, m$:

$$x^j(k) = f^j(x^j(k-1), w^j(k-1)),$$
$$z(k) = h(x^j(k), v^j k).$$

  This requires identifying a transition from the standard model ($h = 0$) to any "faulty" model ($h = 1, \ldots, m$).

- The approach involves calculating state estimates $\hat{x}^j(k)$ for each model at every timestep and conducting a test, like a posterior or log-likelihood ratio, to detect system jumps, necessitating a set of parallel estimators.

- In our case, we have $m = 2^M$. A key hurdle is the "curse of dimensionality," where the system's expanding state space and increasing fault modes escalate the complexity of the Multiple Models (MM) estimation problem, especially when using computationally intensive estimators such as particle filtering.

A Parameter Estimation Viewpoint

- Consider the dual challenge of state and parameter estimation in its general form:

$$
\begin{aligned}
x(k) &= f\big(x(k-1), \theta, v(k-1)\big) \\
z(k) &= h(x(k), n(k))
\end{aligned}
$$

- The goal is to estimate both the state $x(k)$ and the parameter vector θ. A common strategy in the literature converts θ into an additional state vector, modeling its transition through a Gaussian random walk:

$$\theta(k) = \theta(k-1) + \zeta(k-1),$$

  with $\zeta(k) \sim \mathcal{N}(0, \Sigma_\zeta)$.

- An intuitive method is to view the change vector $\mathbb{B}(k)$ as a parameter vector.
- However, this direct method faces two main issues:
  1. Literature indicates that a random walk can inflate covariance, leading to more diffused posteriors.
  2. Unlike constant parameters, the change vector varies over time.
- The key insight is converting the dual challenge of parameter and state estimation into a single state estimation task. This is done by treating the parameter as a state and defining an appropriate state-transition difference equation for it.

Nonlinear State Estimation: The Particle Filter

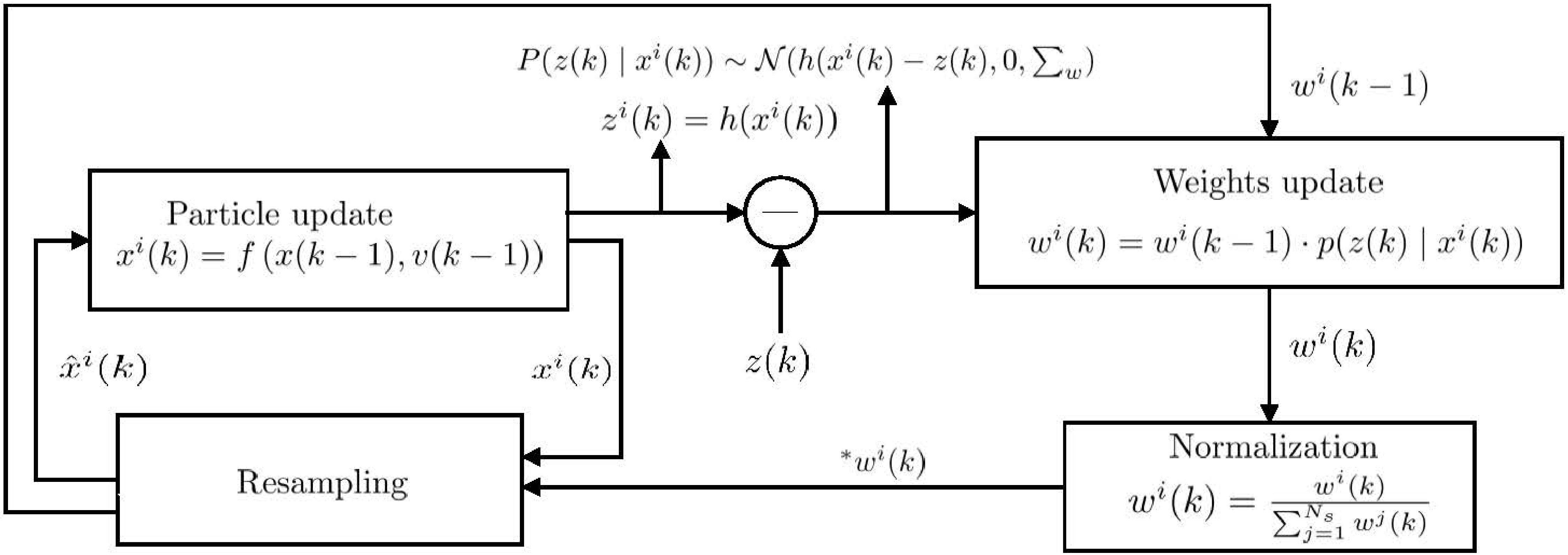
- In general, the bayesian estimation process enriches a prior probability distribution with the information from a likelihood function (by multiplication) to yield an enhanced posterior probability distribution.

- The Particle Filter (PF) idea is to represent a non-Gaussian posterior probability distribution $p(x(k) \mid z(k))$ by a set of randomly drawn particles and weights $\{x^i(k), w^i(k)\}_{i=1}^{N_s}$ such that the expected value of the posterior can be approximated by

$$p(x(k) \mid z(k)) = \sum_{i=1}^{N_s} w^i(k)\delta(x(k) - x^i(k))$$

$$\hat{x} \triangleq \mathrm{E}[p(x(k) \mid z(k))] = \frac{1}{N_s}\sum_{i=1}^{N_s} w^i(k) \cdot x^i(k)$$

Block Diagram of the Particle Filter



Core Contribution: PF Approach for FD

- We introduce a novel state estimation method designed for FD, focusing on the development of a state estimator (filter) with a dual purpose: state estimation and fault identification.
- For ease of reference, this methodology is termed the Failure-Sensitive Filter (FSF).
- We describe a state-transition model for the FSF, which enhances traditional state estimation by integrating fault identification capabilities.
- The FSF is ultimately realized as an algorithm composed of several well-defined modules, each contributing to its innovative dual-purpose functionality.

FSF: A Hybrid System Formulation

- The FSF combines a PF with a noise driven binary transition mechanism.
- The state vector of the FSF is $\mathcal{X}(k) = [x^c(k); \underbrace{b_1(k); \cdots ; b^M(k)}_{\mathcal{B}(k)}] \in \mathbb{R}^{n_x + \cdot M}, \quad x^c(k) \triangleq x(k)$

$$x^c(k) = \underbrace{f(x(k-1), u(k-1)) + \ldots \sum_{j=1}^{M} g^j(x(k-1), u(k-1)) \cdot \boxed{b_j(k-1)} + \tilde{v}(k-1)}_{\text{evolution of continuous states}}$$

$$\boxed{b_j(k)} = \underbrace{\Phi(b_j(k-1), n_j(k-1))}_{\text{evolution of binary states}}, j = 1, \ldots, M$$

$$z(k) = \underbrace{h(x(k)) + \omega(k)}_{\text{measurement}}$$

$$\Phi(x) : \text{ map realizing } p(b_j(k) | b_j(k-1))$$

- The statistical inference of $\mathcal{B}(k)$ yields an estimation of the change vector $\mathbb{B}(k)$.
- The FSF is a nonlinear, non-Gaussian hybrid filter.
- The output of the diagnosis module is the probabilitie of failure $E\left[b_j(k) \mid z(k)\right]_{j=1}^{M}$

Probabilistic Model of the Faults

- A plausible model for $p\left(\beta(k)|\beta(k-1)\right)$ is structured as follows:

$$p\left(\beta(k)|\beta(k-1)\right) = \begin{cases} \eta, & \text{if } \beta(k-1) = 0 \text{ and } \beta(k) = 0; \\ 1-\eta, & \text{if } \beta(k-1) = 0 \text{ and } \beta(k) = 1; \\ 1, & \text{if } \beta(k-1) = 1 \text{ and } \beta(k) = 1; \end{cases}$$

  where $\eta \in (0, 1)$ represents the probability that a fault mode does not emerge in one time step.

- This probability, η, ordinarily unknown, is approximated in the FRF to have a significantly high value (close to one) to reflect the low probability of fault occurrence.

Role of Binary States Transition

- In the FRF, the distribution that dictates fault mode transitions, represented by $p\left(b_j(k)|b_j(k-1)\right)$, is required to at least match the support of $p\left(\beta(k)|\beta(k-1)\right)$.
- Sampling from $p\left(b_j(k)|b_j(k-1)\right)$ within the FRF is achieved through the random sampling map $\Phi\left(b_j(k-1), n_j(k-1)\right)$.
- This map ensures that the updated state $b_j(k)$ behaves as a random variable consistent with the transition distribution $p\left(b_j(k)|b_j(k-1)\right)$.
- The driving noise sequence, $n_j(k) \sim \mathcal{U}(0,1)$, is modeled using the uniform distribution, chosen for its simplicity and the ease it offers in generating random samples.

Binary State-Transition Design

- The recommended state-transition function of binary states, $\Phi(\cdot,\cdot)$, has the following succinct format:

$$\Phi\left(b_j(k-1), n_j(k-1)\right) = \begin{cases} 0, & \text{if } b_j(k-1) = 0 \text{ and } n_j(k-1) \leq \tilde{\eta}; \\ 1, & \text{otherwise.} \end{cases}$$

- For the given state-transition map, it is straightforward to confirm the following:

  1. The probability that $b_j(k)$ remains 0 given $b_j(k-1)$ was 0 is equal to the probability that the noise $n_j(k-1)$ is less than or equal to $\tilde{\eta}$, which is $\tilde{\eta}$ itself, that is, $p\left(b_j(k) = 0 | b_j(k-1) = 0\right) = P\left(n_j(k-1) \leq \tilde{\eta}\right) = \tilde{\eta}$.
  2. The probability that $b_j(k)$ remains 1 given $b_j(k-1)$ was 1 is certain, or 1, that is, $p\left(b_j(k) = 1 | b_j(k-1) = 1\right) = 1$.

Remarks on the FSF Model

- These density values fully characterize the behavior of the binary state $b_j(k)$ based on its previous state $b_j(k-1)$ and the influence of the noise $n_j(k-1)$ modeled by the state-transition function.

- A value of $\tilde{\eta}$ equal to 0.5 reflects the FSF designer's belief that the likelihood of a fault occurring within one time step is equal to the likelihood of it not occurring.

- Our selection of the function Φ is an ergonomic method for maintaining the binary states as random variables and seamlessly integrating them into the difference equation of the FSF.

FSF Algorithm: Initialization

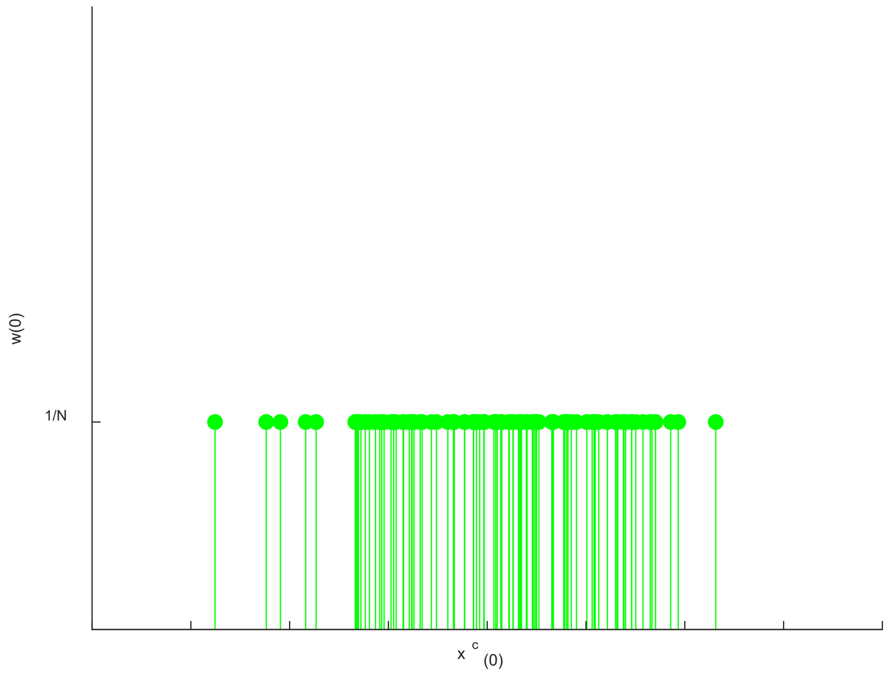
- Draw a set of N continuous-valued state samples

$$\left\{x_c^i(0) \sim \mathcal{N}\left(x(0), \Sigma_{x,0}\right) : i = 1, 2, \ldots, N\right\}$$

  from a multi-variate Gaussian distribution where the mean $x(0) \in \mathbb{R}^{n_x}$ represents the belief about the initial state, and the covariance $\Sigma_{x,0} \in \mathbb{R}^{n_x \times n_x}$ quantifies the uncertainty surrounding this belief.

- It is pressumed that the process is initiated being "healthy," i.e., $\mathbb{B}(k) = \mathbf{0}_M$; hence, all particles of the binary states are set to zero, succinctly expressed as $\left\{\mathcal{B}^i(0) = \mathbf{0}_M : i = 1, 2, \ldots, N\right\}$.

- The initial N FSF state particles are formed as $\mathcal{X}^i(k) = \left[x_c^i(k); \mathcal{B}^i(k)\right]$, with all their weights set equally, namely $w^i = 1/N$ for $i = 1, \ldots, N$.



FSF Algorithm: State Prediction or Particle Update

- The portion of the particles corresponding to the continuous-valued state is updated by

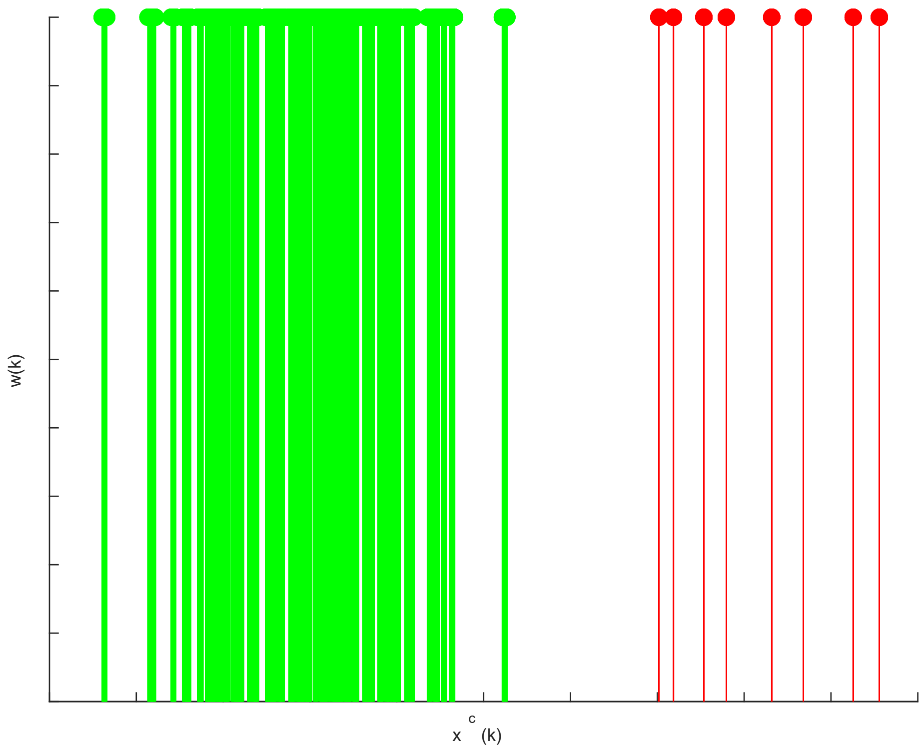
$$x^i(k) = f\left(x^i(k-1), u(k-1) + v^i(k-1)\right),$$

where $\tilde{v}^i(k) \sim \mathcal{N}\left(\mathbf{0}_{n_x}, \Sigma_x\right)$.

- The prediction for the binary state particles is determined by

$$b^i_j(k) = \Phi\left(b^i_j(k-1), n^i_j(k-1)\right),$$

- According to the design of Φ, "faulty" binary particles remain faulty. The "healthy" particles remain healthy with a probability $\tilde{\eta}$, with a value considerably greater than 0.5, and convert into faulty ones with a small probability $1 - \tilde{\eta}$.



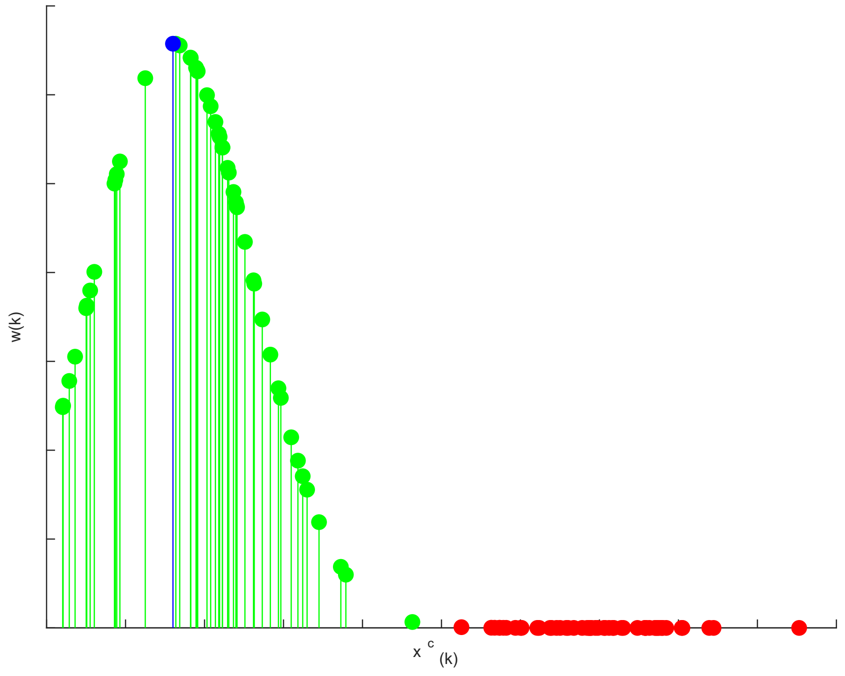
FSF Algorithm: State Correction or Weight Update

- Upon receiving the measurement $z(k)$, the likelihood probability density function (pdf) is determined, which ascertains the importance, or weight, of each predicted particle.

- In the broadest sense, for the bootstrap filter, the unnormalized weights are computed as
$$\tilde{w}^i(k) \propto p\left(z(k)|x^i(k)\right).$$

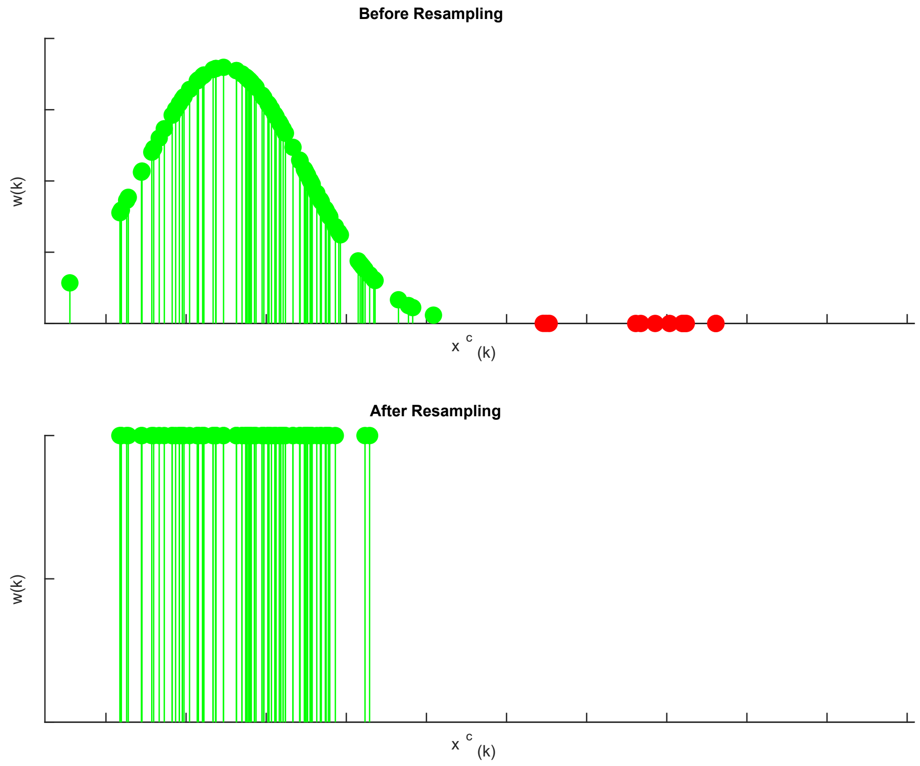
- In practice, the greater the deviation of the predicted observation $z^i(k)$ from the actual observation, the lower the assigned weight to that particle.

- The weights derived from the likelihood calculation are then normalized using the formula:
$$w^i(k) = \frac{\tilde{w}^i(k)}{\sum_{i=1}^{N} \tilde{w}^i(k)},$$
ensuring that the collection $\{\mathcal{X}^i(k), w^i(k)\}_{i=1}^N$ forms a probabilistic representation that approximates the posterior pdf $(\mathcal{X}(k)|z(1:k))$.



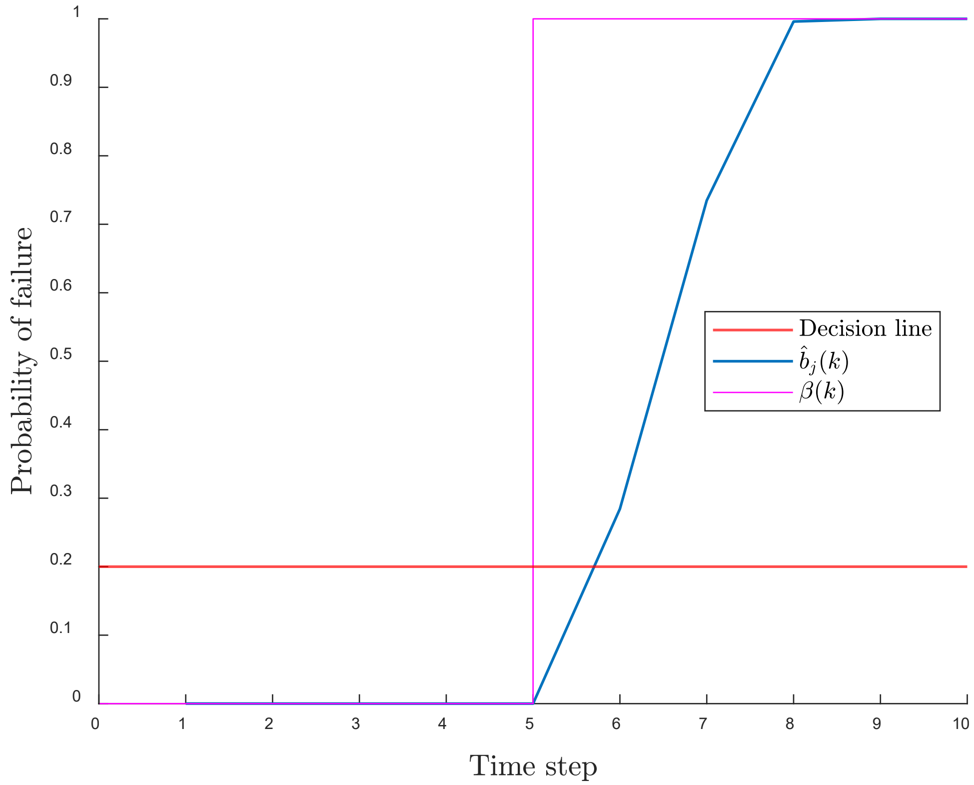
FSF Algorithm: Resampling

- Resampling replaces particles that have low importance weights with those having higher weights.
- This step is essential to mitigate the inherent limitations of using a finite set of particles to approximate a continuous distribution.
- Consequently, resampling leads to the elimination of particles that, following the weight update phase, possess minimal weights. After resampling, the state particles (samples) are uniformly reassigned an equal weight of $1/N$, ensuring that each particle is equally likely to be selected in future prediction steps.
- This mechanism helps concentrate the particle set on more probable regions of the state space, enhancing the filter's efficiency and accuracy in representing the system's posterior distribution.

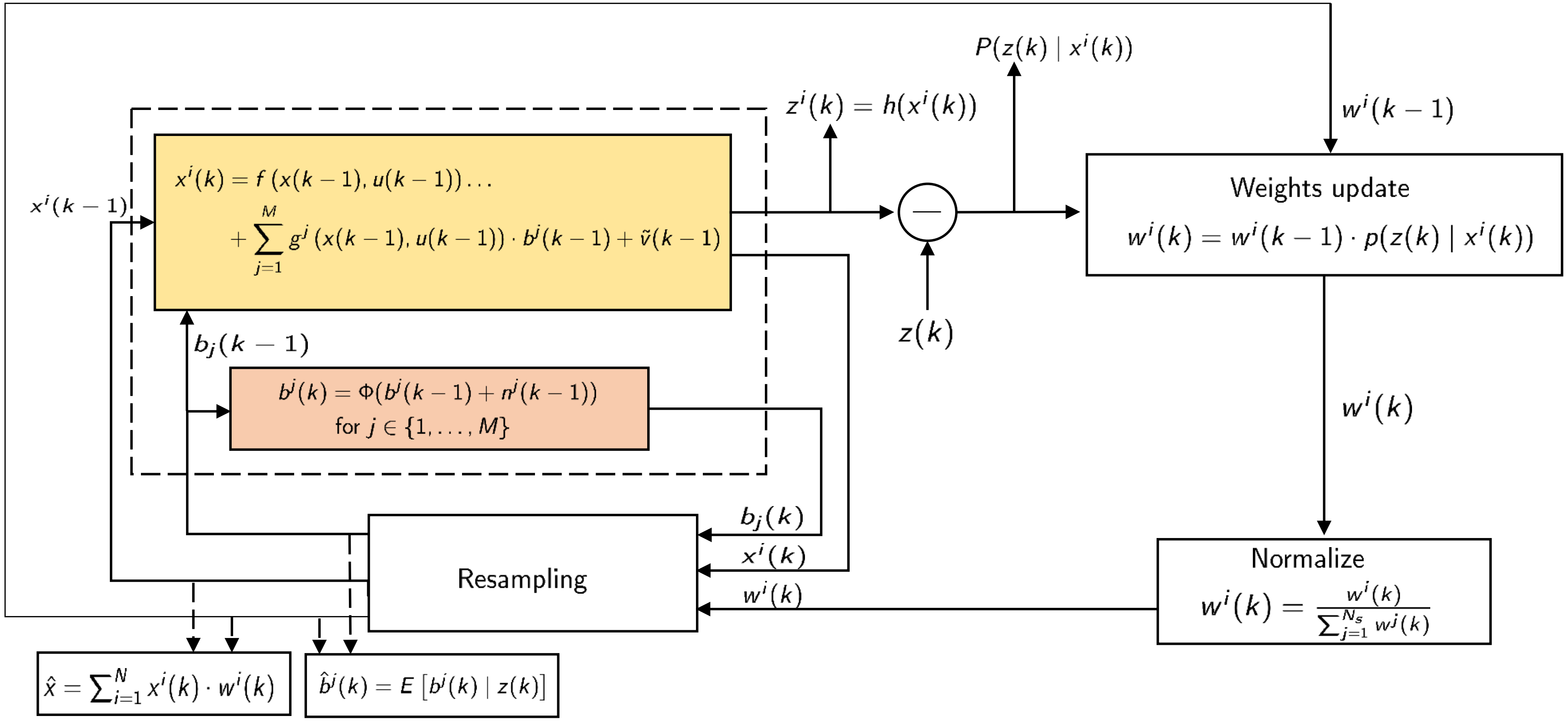


FSF Algorithm: Detection and Isolation

- The determination of a fault's occurrence within the FSF framework is made through a straightforward test: comparing the estimated probability of a fault, denoted as $\hat{b}_j$, against a predefined threshold $\alpha \in (0, 1)$ at each time step. An alarm is triggered if $\hat{b}_j(k) > \alpha$.
- Selecting an appropriate threshold, α, is a nuanced process influenced by practical experience, the prevailing noise level, and the degree of separation between the fault and nominal models.
- The FSF's design, particularly its representation of each fault mode's state by b_j for $j \in \{1, 2, ..., M\}$, facilitates fault isolation by tracking the temporal evolution of $\hat{b}_j(k)$ for specific fault mode detection.
- Consequently, this setup enables the simultaneous detection and identification of multiple faults by observing the binary states corresponding to each fault.



FSF: The Algorithm



Fault-Sensitive Filter output

Actuator Fault Diagnosis: A Quadrotor

- Control Law: Healthy Model.

$$\mathbb{T} = \sigma(x)$$

- Control Law: Fault Model.

$$\bar{\mathbb{T}}_i(k) = (1 - \sigma b_i(k))\mathbb{T}_i(k)$$

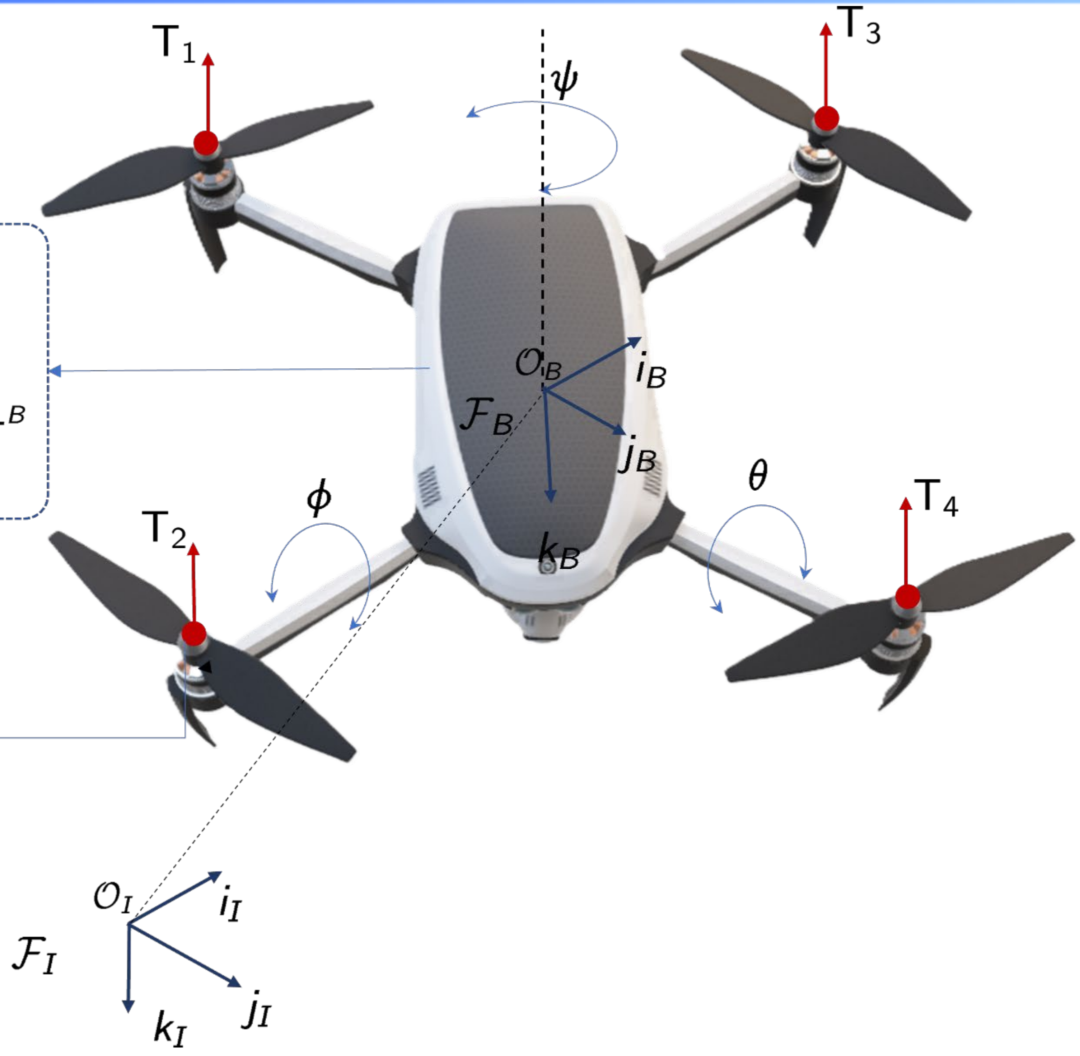
- Resultant Force

$$f^B(\mathbb{T}) = \begin{bmatrix} 0 \\ 0 \\ -(\bar{T}_1 + T_2 + T_3 + T_4) \end{bmatrix} + R^T \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$

Quadrotor dynamics

$$\begin{aligned} \dot{p}^I &= v^I \\ \dot{v}^I &= \frac{1}{m} R f^B(\mathbb{T}) \\ \dot{\omega}^B &= \mathcal{I}^{-1}\left(\mathcal{I}\omega^B \times \omega^B\right) + \mathcal{I}^{-1}\tau^B \\ \dot{\Theta} &= \Psi(\Theta)\omega^B . \end{aligned}$$

$$\begin{bmatrix} x^c(k) \\ \hat{b}_i(k) \end{bmatrix}$$ Augmented State in FSF



Actuator Fault Diagnosis: Equations of Motion

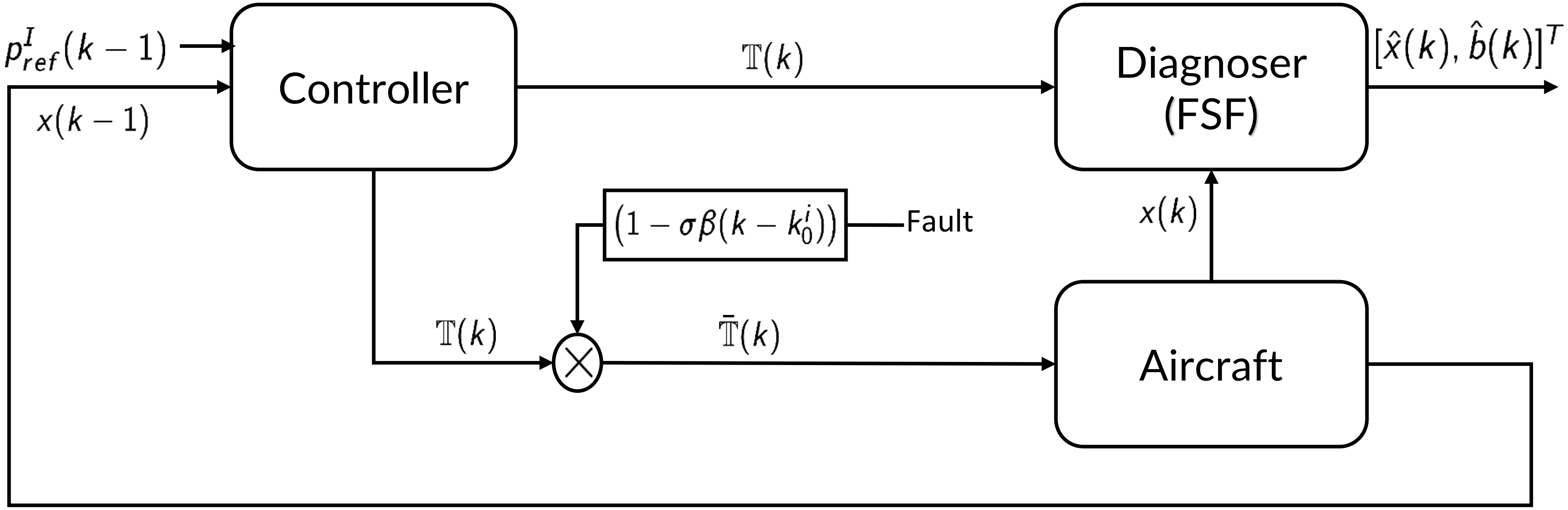
- Discretized equations of motion.

$$
\begin{aligned}
p^I(k) &= p^I(k-1) + \Delta T v^I(k-1) \\
v^I(k) &= v^I(k-1) + \Delta T \cdot \frac{1}{m} f^B(\bar{\mathbb{T}}(k)) \\
\Theta(k) &= \Theta(k-1) + \Delta T \Psi\left(\Theta_k\right) \omega^B(k) \\
\omega^B(k) &= \omega^B(k-1) + \Delta T \left[\mathcal{I}^{-1}\left(\mathcal{I}\omega^B \times \omega^B\right) + \mathcal{I}^{-1}\tau^B(\bar{\mathbb{T}}(k))\right]
\end{aligned}
$$

where $e_3 = [0 \quad 0 \quad 1]^T$ and ΔT denotes the sampling period.

- Let $b_i \in \{0, 1\}$, with $i = 1, \ldots, 4$, being the binary state variable that signifies a fault in the thrust production T_i of rotor i.

Actuator Fault Diagnosis: Quadrotor + FSF

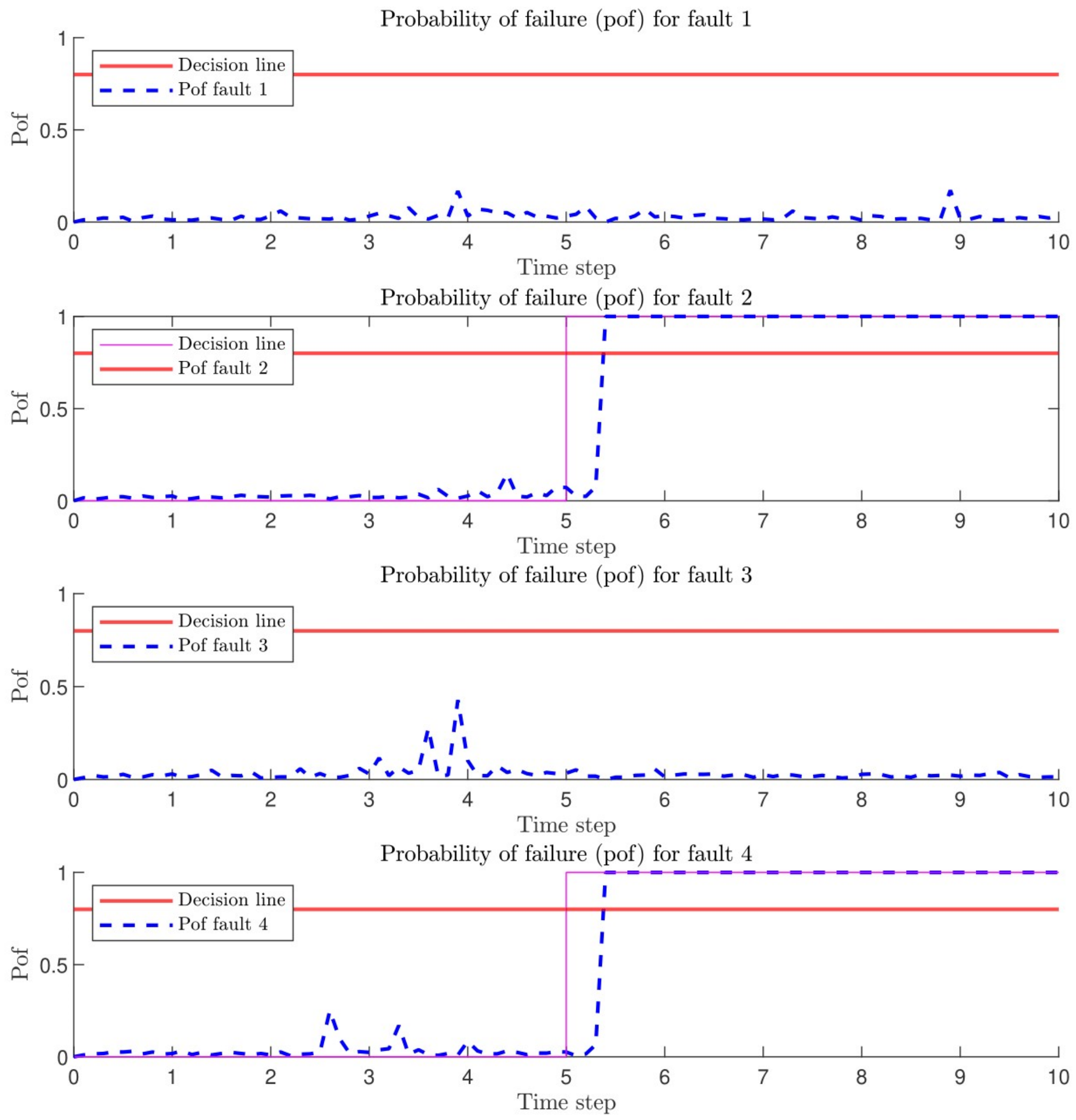


Block diagram depicting the signal interconnection of the quadrotor with the FSF

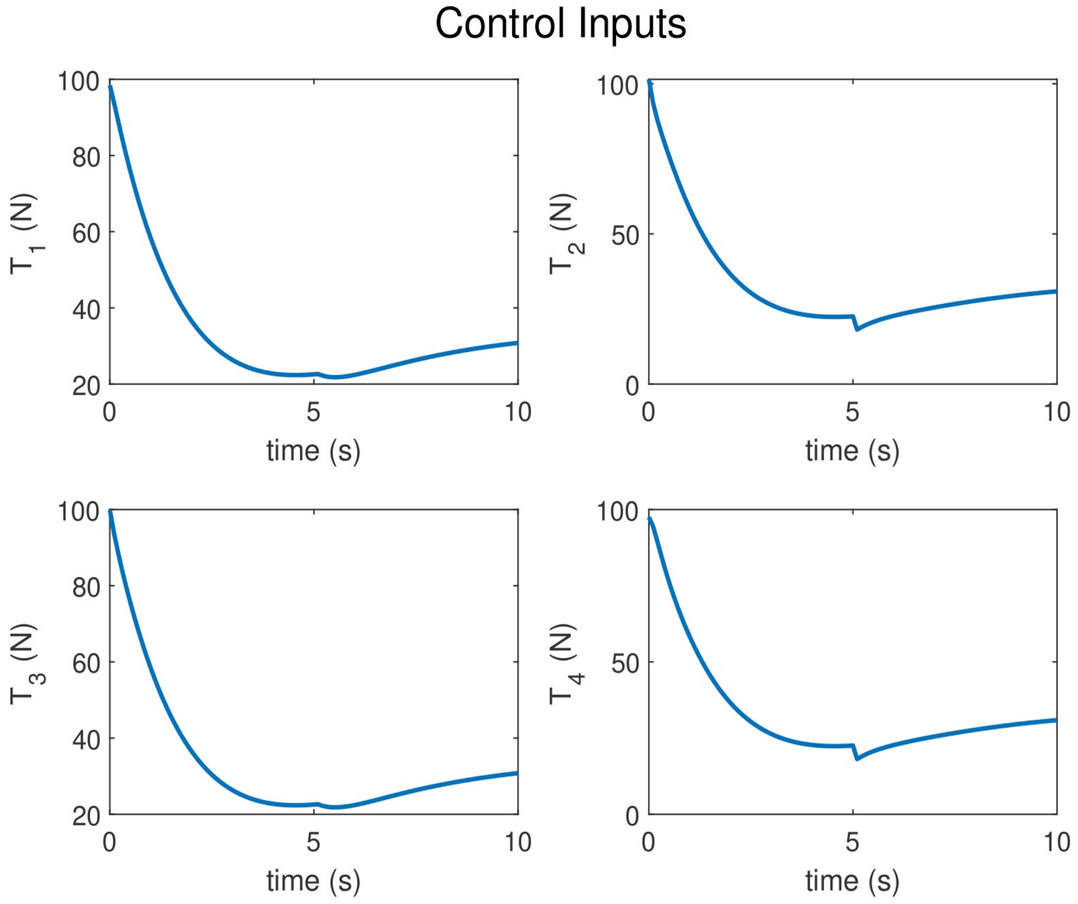
Simulation

- We simulate a scenario involving a loss of control effectiveness in two actuators.
- The quadrotor is initialized for takeoff from the ground and commanded to hover at an altitude of 100 meters.
- The simulation is set to run for 10 seconds. A fault is injected at 5 seconds after takeoff.
- Specifically, the fault injected is a simultaneous loss of 20% control effectiveness in two rotors. The entire simulation is implemented in Matlab.

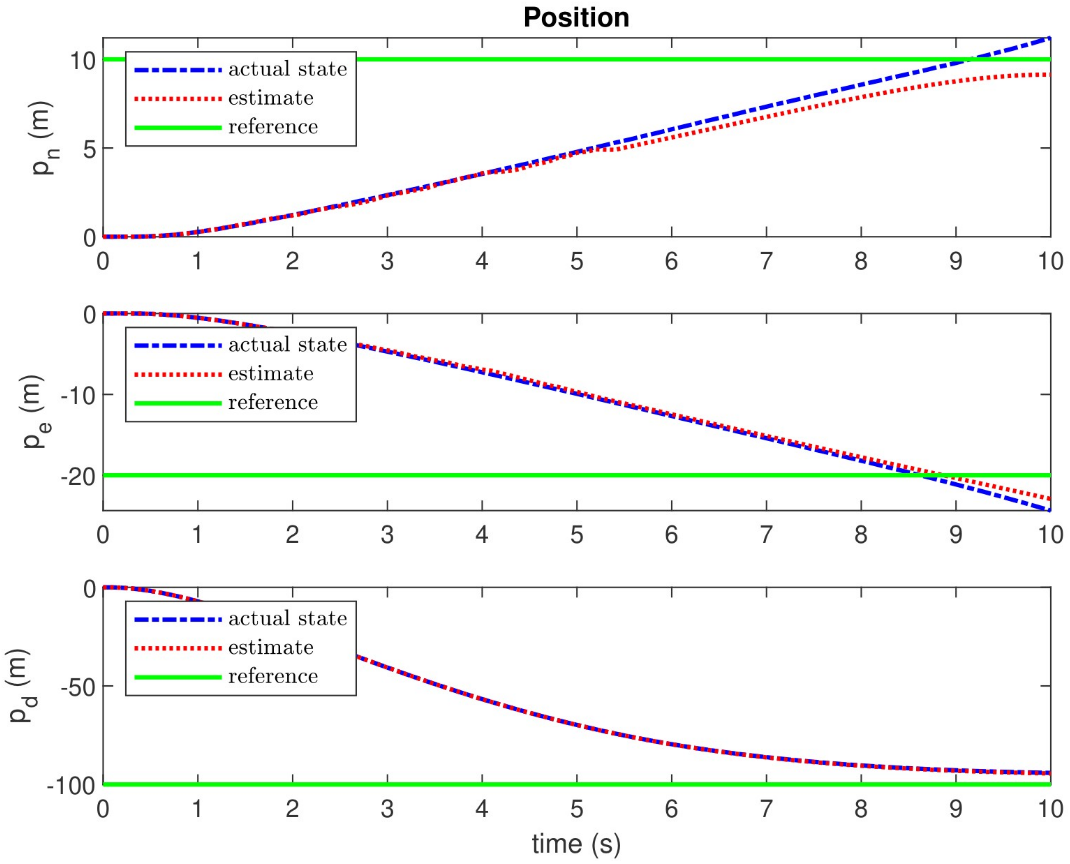
Result: Probability of failure



Result:

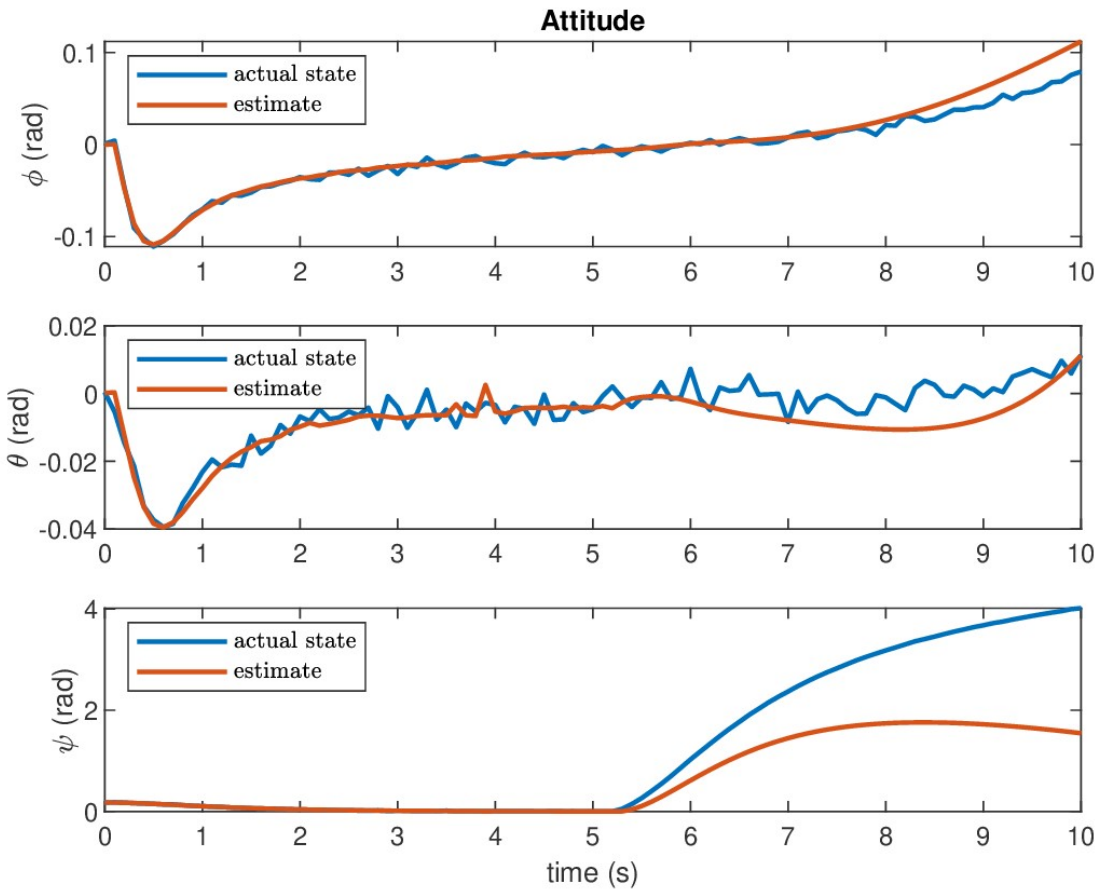


Control Inputs
T 1 (N)
T 2 (N)
T 3 (N)
T 4 (N)
time (s)

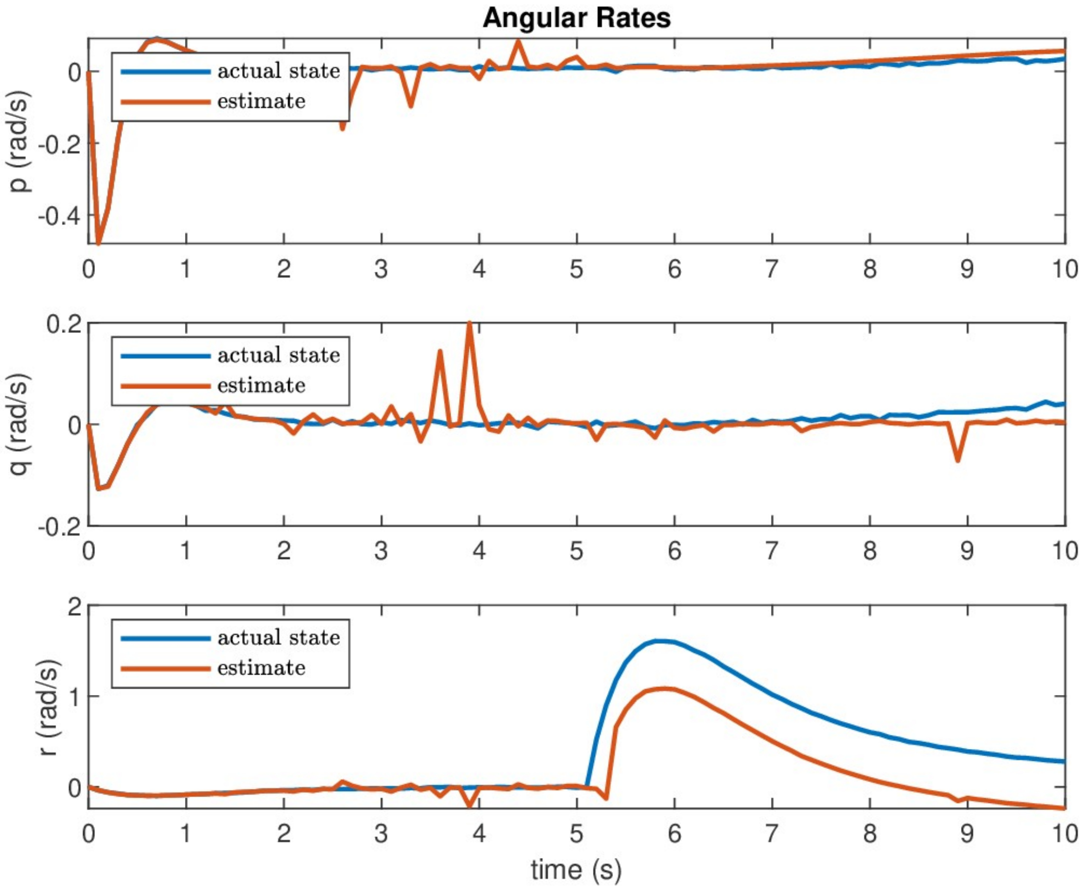


Position
actual state
estimate
reference
p n (m)
p e (m)
p d (m)
time (s)

Result:



Attitude
actual state
estimate
ϕ (rad)
θ (rad)
ψ (rad)
time (s)



Angular Rates
actual state
estimate
p (rad/s)
q (rad/s)
r (rad/s)
time (s)

Conclusion

- Our approach utilizes a hybrid dynamic system formulation, featuring binary and continuous-value states, that streamlines the identification of multiple, potentially concurrent faults.
- Only a single set of state estimate is computed during the estimation step.
- The resulting filter eliminates the computational overhead associated with the realization of the bank of estimators exhibited in conventional Multiple Model FD (MMFD) approaches.
- In addition, our method boasts a straightforward algorithmic implementation that is well-suited for real-world integration in embedded flight control systems.

Participants and Support

Vincent Kwao
Ph.D. Candidate
North Carolina A&T State University

The End

Thank you for your patience!
Questions?