Joint feedback control and fault diagnosis disambiguation

Ion Matei[1], Maksym Zhenirovsky[2], Johan de Kleer[3], and Kai Goebel[4]

[1,2,3,4] *Palo Alto Research Center, Palo Alto, CA, 94304, USA*
imatei@parc,com
mazhenir@parc.com
dekleer@parc.com
kgoebel@parc.com

ABSTRACT

This paper proposes a model-based diagnosis approach to detect and isolate intermittent faults in complex systems that operate under feedback control. The feedback control attempts to compensate for model uncertainties and deviations from nominal behavior, but these uncertainties are crucial for accurate fault diagnosis. We focus on faults that are observable only in a particular region of the state space, which is rarely reached in nominal behavior. To address this, we present an approach that considers both control requirements and diagnosis uncertainty in an optimization problem similar to model-predictive control. We compute perturbations on control signals that forces the system to reach states where faults are detectable. We apply our approach to a quadrotor system under motion feedback control, demonstrating the effectiveness of our method. Our approach has the potential to improve the resilience of complex systems by quickly detecting and recovering from disruptive events.

1. INTRODUCTION

The resilience of complex systems depends on the ability to quickly detect, respond to, and recover from disruptive events, such as faults. Therefore, developing effective fault diagnosis algorithms for systems under feedback control is crucial. However, diagnosing faults in such systems can be challenging, particularly when dealing with intermittent faults that are difficult to accurately diagnose due to feedback control actions. In this paper, we propose an approach that addresses this issue by applying diagnosis algorithms to a system under feedback control to detect and isolate faults while satisfying control-related requirements.

The control and diagnosis algorithms operate on opposing sides of the system. Feedback control aims to eliminate deviations from nominal behavior by compensating for model uncertainties. However, these uncertainties are at the core of accurate diagnosis as they indicate the presence of faults. Therefore, we focus on diagnosing faults that are observable only when the system is in a particular region of the state space, which is rarely reached in nominal behavior. To achieve this, we utilize a model-based diagnosis (MBD) approach, which has a long history in the artificial intelligence (de Kleer, Mackworth, & Reiter, 1992) and control fields (Gertler, 1998),(Isermann, 2005),(Patton, Frank, & Clark, 2000). Traditional MBD approaches in the control community include filters such as Kalman (Kalman, 1960) and particle filters (Gordon, Salmond, & Smith, 1993; Arulampalam, Maskell, & Gordon, 2002)) or optimization-based techniques that estimate parameters whose deviation from nominal values indicate the presence of a fault.

We introduced in (Matei, Zhenirovskyy, de Kleer, & Goebel, 2022) an approach to improving diagnostic certainty by generating control inputs to disambiguate faults in a fuel system. However, that approach did not consider control objectives and focused solely on diagnosis. In this paper, we take a step further by considering a system under feedback control and aim to determine small changes applied to the control signals generated by control algorithms that achieve two objectives: (i) keeping the system in a desired region of the state space and (ii) reducing diagnosis uncertainty. Our approach is similar to model-predictive control (MPC) (Garcia, Prett, & Morari, 1989), which solves an optimization problem to determine control inputs over a time horizon. In our case, the objective function in the MPC algorithm is geared towards minimizing diagnosis uncertainty while control requirements are implemented through optimization constraints. The uncertainty minimization objective is defined in terms of similarity functions among observations corresponding to the ambiguous fault modes. By minimizing the similarity among these observations, we reduce the information theoretic entropy function applied to a random variable defining the fault mode, thus we reduce the uncertainty.

To showcase our method, we applied it to a quadrotor system, which is controlled by a PD cascade feedback control scheme. The quadrotor system is nonlinear, and nominal control inputs must be changed carefully due to safety constraints.

Notations: We use bold letters to denote vectors. To represent discrete time dependency, we use the notation $x(t_k) = x_k$, for time instants t_k. A sequence of variables over time, i.e., a time series, $\{x_k\}_{k=0}^{K}$ is denoted by $x_{0:K}$. We denote the probability distribution function (p.d.f.) of a random variable X by $f_X(x)$. We represent the conditional p.d.f. of $X|Y$ by $f_{X|Y}(x|y)$. When there is no loss of clarity, we omit the subscript notation of $f_{X|Y}(x|y)$, that is, we will use $f(x|y)$. We denote by $\mathcal{N}(\mu, \Sigma)$ the multivariate Gaussian distribution with mean μ and covariance matrix Σ.

Paper structure: In Section 2, we present the system model and introduce the diagnosis problem. In Section 3, we describe an optimization-based approach for diagnosing faults that estimates fault parameters using the system model. In Section 4, we discuss our approach for disambiguating diagnoses using an optimal control approach, followed by showcasing the application of our approach on a quadrotor system that has intermittent faults in its rotors.

2. PROBLEM STATEMENT

The scope of our analysis is to investigate parametric faults, where each fault mode is characterized by a scalar parameter. We define a set of fault parameters $\mathcal{F} = \{f_1, f_2, \ldots, f_N\}$, where N is an integer number. Under normal operating conditions, each fault parameter f_i has a nominal value $\bar{f}_i$, and the parameter vectors are assumed to remain close to these nominal values.

To diagnose the health of the system, we use MBD. Our approach involves analyzing a closed-loop system consisting of a nonlinear model of the system and a controller that computes the inputs to the system. More precisely, the system can be represented as follows:

$$
\begin{aligned}
\dot{\boldsymbol{x}} &= f(\boldsymbol{x}, \boldsymbol{u}; \boldsymbol{p}), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0, && (1)\\
\boldsymbol{y} &= h(\boldsymbol{x}) + \boldsymbol{v}, && (2)\\
\boldsymbol{u} &= \mathcal{C}(\boldsymbol{y}; \boldsymbol{x}_f), && (3)
\end{aligned}
$$

where $\boldsymbol{x}$, $\boldsymbol{u}$, $\boldsymbol{p}$, and $\boldsymbol{y}$ represent the state, input, fault parameter, and output vectors, respectively. We assume that the control inputs are generated by a map $\mathcal{C}$, which utilizes the output measurements to guide the system towards a final state $\boldsymbol{x}_f$. The output measurements are affected by independent and identically distributed (i.i.d.) additive noise, which we denote by $\boldsymbol{v}$. We assume that this noise follows a Gaussian distribution, with a zero mean and a covariance matrix of Σ_v.

We define a set of fault parameters $\mathcal{F} = \{f_1, f_2, \ldots, f_N\}$, where N is an integer number. We assume that each fault parameter f_i has a nominal value $\bar{f}_i$, and in normal conditions, the parameter vectors remain close to these nominal values.

We use the single fault scenario, i.e., no two faults became active at the same time. The fault event is defined by $\{|p_i - \bar{p}_i| > \varepsilon_i, p_{-i} = \bar{p}_{-i}\}$, where ε_i is a positive scalar. The scalar ε_i depends on the measurement noise and the sensitivity of the behavior of the system to changes in parameter p_i. The fault magnitude is determined by estimating the value of the system parameter p_i. Given a sequence of input and output measurements over the time horizon τ, the diagnosis problem consists of computing the conditional probability $\mathbb{P}(|p_i - \bar{p}_i| > \varepsilon_i | \boldsymbol{y}_{0:\tau}, \boldsymbol{u}_{0:\tau})$, for all i together with the estimation of the parameter p_i. An ambiguous diagnosis appears when there exist a least two faults i and j so that $\mathbb{P}(|p_i - \bar{p}_i| > \varepsilon_i | \boldsymbol{y}_{0:\tau}, \boldsymbol{u}_{0:\tau}) \approx \mathbb{P}(|p_j - \bar{p}_j| > \varepsilon_j | \boldsymbol{y}_{0:\tau}, \boldsymbol{u}_{0:\tau})$, meaning that their probability is roughly the same, impeding a clear decision on what fault is the root cause of the observed anomalous behavior.

To illustrate our approach to fault detection and disambiguation, we consider a quadrotor system model (Bouabdallah & Siegwart, 2007), whose state variables are the positions and angles, and the linear and angular velocities. The model was derived by considering various physical effects such as rolling, pitching and yawing moments, and rotor dynamics. We assume that the perturbations from hover flight are small. Thus, the transformation matrix between the rate of change of the orientation angles ($\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$) and the body angular velocities can be considered as unity matrix. The state space representation is given by

$$
\begin{aligned}
\ddot{\phi} &= b_1 U_2, && (4)\\
\ddot{\theta} &= b_2 U_3, && (5)\\
\ddot{\psi} &= b_3 U_4, && (6)\\
\ddot{x} &= \frac{u_x}{m} U_1, && (7)\\
\ddot{y} &= \frac{u_y}{m} U_2, && (8)\\
\ddot{z} &= g - \frac{\cos\phi \cos\theta}{m} U_1, && (9)
\end{aligned}
$$

where ϕ, θ and ψ are the pitch, roll and yaw angles, and the tuple x, y, z is the position of the quadrotor in the 3-d space. The variables u_x and u_y are expressed as functions of the angles, i.e., $u_x = \cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi$, and $u_y = \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi$ The inputs to the system are linear combinations of the (scaled) angular velocities generated by the rotors:

$$
\begin{aligned}
U_1 &= b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2), && (10)\\
U_2 &= b(-\Omega_2^2 + \Omega_4^2), && (11)\\
U_3 &= b(\Omega_1^2 - \Omega_3^2), && (12)\\
U_4 &= b(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2). && (13)
\end{aligned}
$$

In controlling quadrotors, the typical approach involves three cascading stages: altitude, position, and attitude control. To simplify the process, we utilize PD-based control strategies. However, for those interested in PID-based control schemes, we refer readers to (Bouabdallah & Siegwart, 2007). Given a set of desired linear position and velocities $(x_r, y_r, z_r, \dot{x}_r, \dot{y}_r, \dot{z}_r)$, a PD controller for setting the quadrotor at altitude z_r is

$$U_1 = \frac{m}{\cos\phi\cos\theta}(g - k_3^l z_e - d_3^l \dot{z}_e),$$

where $z_e = z - z_r$, and k_3^l, d_3^l being the proportional and derivative coefficients of the controller. The motion in the (x, y) plan is determined by changing the angles of the quadrotor. For example, first we use PD controllers to compute quantities used to set the angles:

$$\alpha_x = \frac{m}{U_1}(-k_1^l x_e - d_1^l \dot{x}_e),$$

$$\alpha_y = \frac{m}{U_1}(-k_2^l y_e - d_2^l \dot{y}_e),$$

where $x_e = x - x_r$, $y_e = y - y_r$. These quantities are used to set the pitch and roll reference angles: $\hat{\phi}_r = \alpha_x \sin(\psi) - \alpha_y \cos(\psi)$, $\hat{\theta}_r = \alpha_x \cos(\psi) - \alpha_y \sin(\psi)$. The last step is to determine inputs for tracking the reference angles:

$$U_2 = \frac{1}{b1}(-k_1^a \phi_e - d_1^a \dot{\phi}_e),$$

$$U_3 = \frac{1}{b2}(-k_2^a \theta_e - d_2^a \dot{\theta}_e),$$

$$U_4 = \frac{1}{b3}(-k_3^a \psi_e - d_3^a \dot{\psi}_e),$$

with $\phi_e = \phi - \hat{\phi}_r$, $\theta_e = \theta - \hat{\theta}_r$ and $\psi_e = \psi - \psi_r$, where ψ_r is the yaw reference angle. We tuned the parameters of the controller so that the quadrotor is able to reach a hovering position (0,0,2) from an initial hovering position of (1,1,0) in less than 2 seconds. We assume that in the nominal case, the angular velocities of the rotors are upper bounded, i.e., $\Omega_i \leq \Omega_{max}^2$, where $\Omega_{max} = 1000$rad/s. As such, the control inputs U_i are bounded as follows: $0 \leq U_1 \leq 4b\Omega_{max}^2$, $-b\Omega_{max}^2 \leq U_2 \leq b\Omega_{max}^2$, $-b\Omega_{max}^2 \leq U_3 \leq b\Omega_{max}^2$ and $-2b\Omega_{max}^2 \leq U_4 \leq 2b\Omega_{max}^2$.

The numerical, model parameters are given by L=0.3 m, r=0.1 m, m=1.2 kg, and g=9.81 $\frac{\text{m}}{\text{s}^2}$. In addition, we have the following parameter definitions: $i_{xx} = \frac{2mr^2}{5} + 2mL^2$, $i_{yy} = i_{xx}, i_{zz} = \frac{2mr^2}{5} + 4mL^2, b_1 = \frac{L}{i_{xx}}, b_2 = \frac{L}{i_{yy}}, b_3 = \frac{L}{i_{zz}}$ and $b = 2.5 \times 10^{-5}$. The controller parameters are given by k_1^l=5.587, d_1^l=3.482, k_1^a=84.817, d_1^a=12.753, k_2^l=5.356, d_2^l=3.364, k_2^a=86.574, d_2^a=12.773, k_3^l=16.165, d_3^l=7.428, k_3^a=1283.786, d_3^a=1283.786.

Table 1. Controller parameters

k_1^l=5.587	d_1^l=3.482	k_1^a=84.817	d_1^a=12.753
k_2^l=5.356	d_2^l=3.364	k_2^a=86.574	d_2^a=12.773
k_3^l=16.165	d_3^l=7.428	k_3^a=1283.786	d_3^a=1283.786

Figure 1 displays the 3D trajectory of the quadrotor while under feedback control. The task involved moving the quadrotor from its initial hovering position of (1,1,0) to the final hovering position of (0,0,2).

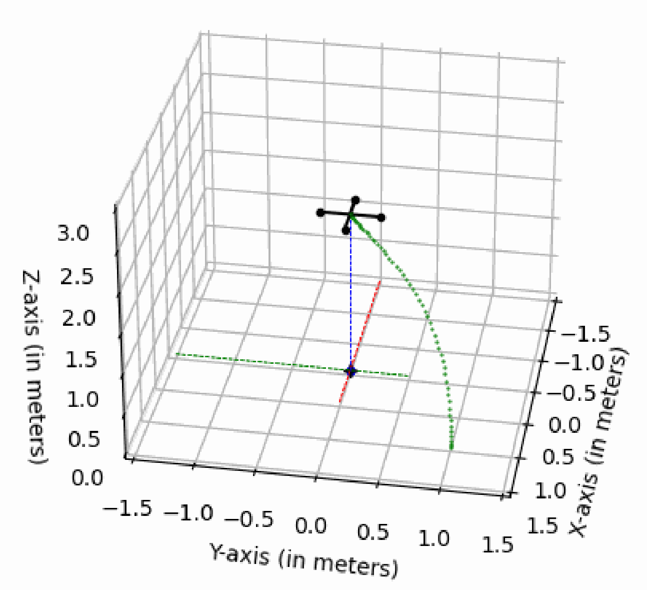

Figure 1. Feedback control generated trajectory for moving from (1,1,0) to (0,0,2) in nominal conditions.

To model intermittent faults in the rotors, we assume that the maximum velocity the rotor can generate decreases, and is expressed by $\Omega_i \leq p_i \Omega_{max}^2$, where $p_i \in [0, 1]$ represents the fault parameters we aim to estimate. Under nominal conditions, the fault parameters have a value of 1. Previous research has studied various types of rotor faults; for instance, (Guzmán-Rabasa et al., 2019) considers additive rotor faults, while (Avram, 2016) assumes multiplicative rotor fault models.

Our focus on intermittent rotor faults is due to their challenging nature: these faults only become noticeable when the rotors need to operate at their maximum angular velocity, making them harder to detect and isolate. It is worth noting that reaching the upper limit of maximum velocities may not be necessary, depending on the mission requirements for the quadrotor.

3. Model-based diagnosis

To test the effects of the nominal and perturbed control signals on diagnosis, we utilize an optimization-based approach to estimate the fault parameters. The optimization problem takes a time series of output measurements (e.g., position, angles, and linear and angular velocities) over a time horizon τ, denoted as $\boldsymbol{y}_{0:\tau}$, and estimates a fault parameter for a single fault scenario. The formal description of the optimization problem is given by

$$\min_{p_i} \sum_{k=0}^{\tau} \left(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^i\right)^T \Sigma_v^{-1} \left(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^i\right), \tag{14}$$

where $\boldsymbol{y}_k$ are the measurements at times t_k and $\hat{\boldsymbol{y}}_k^i$ are the predicted outputs when the system is in fault mode i. To generate estimates of the fault parameters $\hat{p}_j$ for $j \neq i$, we solve such optimization problems for each fault mode. To determine which fault is active, we compute empirical probabilities using the formula:

$$q_i \propto \mathcal{N}(\hat{\boldsymbol{y}}_k^i, \Sigma_v).$$

Here, $\hat{\boldsymbol{y}}_k^i$ is obtained by simulating the model for a 5 seconds time interval, with the fault parameter $\hat{p}_i$ while setting the remaining fault parameters to their nominal values. These empirical probabilities serve as proxies for the conditional probability density functions, given by:

$$f(p_i|\boldsymbol{y}_{0:\tau}) = \frac{\prod_{k=0}^{\tau} f(\boldsymbol{y}_k|p_i)f(p_i)}{\int \prod_{k=0}^{\tau} f(\boldsymbol{y}_k|p_i)f(p_i)dp_i}.$$

The proxies become the probability density function $f(p_i|\boldsymbol{y}_{0:\tau})$ when the prior distributions of the fault parameters are uniform.

We do not emphasize the choice of diagnosis algorithms since our focus is on the content of the output measurements that lead to a diagnosis. Filtering-based methods such as particle filters (Arulampalam et al., 2002) or various versions of the Kalman filter (McElhoe, 1966; Julier & Uhlmann, 1997), where the fault parameters are transformed into state variables, are appropriate for estimating the fault parameters. Qualitative diagnosis algorithms, such as analytical redundant relations (ARRs) (Staroswiecki, 2000; Staroswiecki & Comtet-Varga, 2001), can also be used, although they have to be coupled with fault magnitude estimation algorithms. These have the advantage that they do not require fault models, but they typically need more sensors to generate an unambiguous diagnosis solution. We speed up the diagnosis algorithm, by solving the optimization problems corresponding each fault hypothesis using parallel processes.

To show what a diagnosis algorithm would produce when using nominal control inputs, only we created a scenario where we sequentially injected faults to each of the four rotors. We considered a 60% loss of maximum squared angular velocity and perfect measurements. The output data is based on moving the quadrotor from an initial position of (1,1,0) to a final position of (0,0,2). Figure 2 shows the trajectories for the state variables and the control inputs when the first rotor is faulty. The diagnosis results are presented in Table 2. It can be observed that under all fault scenarios, the diagnosis is inaccurate and ambiguous when the trajectories are generated by the nominal controller. This is due to the insufficient exertion of the controller, which fails to push the system to the regime where the maximum velocity limitation becomes evident and observable. Using a PID controller instead of a PD controller would not help either, since the actuation degradation cannot be supplemented by the integral term of the controller.

Table 2. Diagnosis results under nominal control

Scenario	Fault parameter estimates ($\hat{p}1$,$\hat{p}2$,$\hat{p}3$,$\hat{p}4$)	Fault probabilities ($q1$,$q2$, $q3$, $q4$)
$p1$=0.4	0.369,0.352,0.352,0.352	0.25,0.25,0.25,0.25
$p2$=0.4	0.369,0.352,0.352,0.352	0.25,0.25,0.25,0.25
$p3$=0.4	0.369,0.352,0.352,0.352	0.25,0.25,0.25,0.25
$p4$=0.4	0.369,0.352,0.352,0.352	0.25,0.25,0.25,0.25

4. FAULT DISAMBIGUATION

In this section, we present our methodology for designing perturbations on control inputs that lead to unambiguous diagnostic solutions. Specifically, we first formulate an optimal control problem to compute these perturbations. Then, we apply this formulation to the use case of a quadrotor.

4.1. Control perturbation design

Our goal is to design controller input perturbations that generate output measurements with sufficient information for accurate fault diagnosis. To achieve this, we formulate an optimal control problem that, when applied to a set of ambiguous fault modes, determines a set of input perturbations capable of producing output measurements that are as dissimilar as possible.

To measure the similarity between two output measurement vectors, we define a similarity metric. Let i be the index of a fault mode, $\boldsymbol{p}_i$ be the fault parameters that describe this mode, and $\boldsymbol{y}_i(k)$ represent the outputs measured in this fault mode at time t_k. The similarity between the outputs of the system in two fault modes i and j is defined as $s_{i,j} = -\frac{1}{\tau}\sum_{k=1}^{\tau} |\boldsymbol{y}_i(k) - \boldsymbol{y}_j(k)|^2$. The cumulated similarity among outputs in all fault modes is given by $\mathcal{S} = \sum_{i>j} s_{i,j}$. To reduce the variance between similarities when designing the input perturbations, we introduce the quantity $\mathcal{D} = \sum_{i,j \neq l,m} |s_{i,j} - s_{l,m}|$ as a regularization function. Our objective is to minimize $\mathcal{S} + \lambda\mathcal{D}$, where λ is a regularization parameter. When designing the perturbations, we ensure that they do not adversely affect the system's mission. For instance, if the objective is to reach a final position, we impose this constraint when learning the perturbations. The fault parameters $\boldsymbol{p}_i$ impose constraints on the values that the control input can take under fault mode i, constraints which

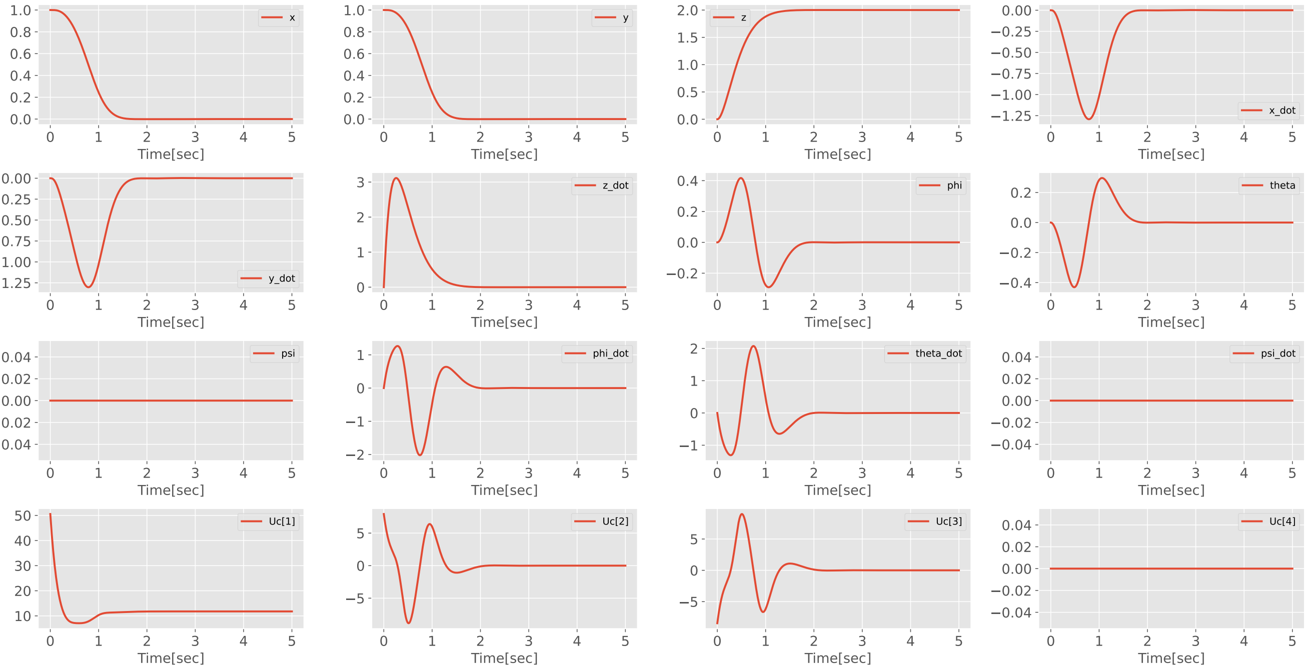

Figure 2. Trajectories of the state variables and the control inputs for p_1=0.4, and $p_2 = p_3 = p_4 = 1$.

we denote by $\mathcal{U}_i$. We also add state constraints to ensure that the system remains within a safe set of states, denoted by $\mathcal{X}$.

With these definitions, we can formulate the optimal control problem that computes the disambiguating input perturbations:

$$\begin{array}{rcl} \min_{\delta \boldsymbol{u}} & & \mathcal{S} + \lambda \mathcal{D}, \forall i \\ \text{subject to:} & & \\ \dot{\boldsymbol{x}}_{\boldsymbol{i}} & = & f(\boldsymbol{x}_i, \boldsymbol{u}_i + \delta \boldsymbol{u}; \boldsymbol{p}_i)\,, \boldsymbol{x}_i(0) = \boldsymbol{x}_0, \forall i \\ \boldsymbol{y}_i & = & h(\boldsymbol{x}_i; \boldsymbol{p}_i) + \boldsymbol{v}_i, \forall i \\ \boldsymbol{u}_i & = & \mathcal{C}(\boldsymbol{y}_i; \boldsymbol{x}_f), \forall i \\ \boldsymbol{u}_i + \delta \boldsymbol{u} & \in & \mathcal{U}_i, \forall i \\ \boldsymbol{x}_i & \in & \mathcal{X}, \\ \boldsymbol{x}_i(\tau) & = & \boldsymbol{x}_f, \forall i, \end{array}$$

where i is the fault index, $\boldsymbol{u}_i$ is the vector of controls generated by the PD controllers when the quadrotor is in fault mode i.

The perturbation vector $\delta \boldsymbol{u} = [\delta u_1, \delta u_2, \delta u_3, \delta u_4]$ must be valid for all fault modes and, therefore, its values must satisfy the constraint $\boldsymbol{u}_i + \delta \boldsymbol{u} \in \mathcal{U}_i$ for all i. To account for multiple fault modes, we create duplicates of the system model, and the complexity of the optimization problem increases as the number of fault modes considered jointly increases. Although CasADi (J. A. E. Andersson, Gillis, Horn, Rawlings, & Diehl, 2019) is a valid option for solving the optimization problem, we have chosen to use a gradient-free optimization algorithm and a Function Mockup Unit (FMU) (Blochwitz et al., 2011) representation of the closed-loop system. We simulate the FMU with the control perturbations as inputs and instantiate it with the set of fault parameters $\boldsymbol{p}_i$ to generate various fault modes. Gradient-free algorithms enables us to use non-differentiable operators such as `min` or `max`, to impose constraints on the input and state vector. The FMU simulations were done using the `pyfmi` Python package (C. Andersson, Akesson, & Fuhrer, 2016), and the numerical optimizations were executed using Powell algorithm from Python `scipy`'s optimization package.

We model the input perturbations as a piecewise linear function with a hyperparameter that controls the number of change points over the time horizon, allowing us to manage the number of optimization parameters.

To enforce safety state constraints and final state constraints, we add terms to the cost function. For example, the term $\mu\left(\max\{0, \text{dist}(\mathcal{X}, \boldsymbol{x}_i) - \varepsilon\} + \max\{0, |\boldsymbol{x}_f - \boldsymbol{x}_i| - \varepsilon\}\right)$, where μ is a large positive scalar, `dist` is a distance operator, and ε is a small positive scalar, enforces the necessary constraints. The resulting input perturbations can be used to diagnose the system before executing a mission. However, if too many fault modes are considered at once, learning the control input perturbations may fail since we over-constrain the perturbations. Therefore, a better approach is to limit the number of jointly considered fault modes, learn perturbations for groups of joint fault modes, and apply them sequentially for each such groups.

4.2. Results for the quadrotor use case

In Section 3, we showed that the diagnosis algorithm under nominal control fails to produce the correct diagnosis in a single fault scenario where all fault modes have the same magnitude. Therefore, at a minimum we can determine disambiguation input perturbation for rotor faults with the same fault magnitude. We discretize the fault magnitude domain $[0, 1]$ and generate perturbation inputs for each discrete fault magnitude. We specifically generate perturbations for the fault magnitudes $0.4, 0.5, 0.6, 0.7, 0.8, 0.9$, and for each of these values, we consider 4 fault modes that correspond to a loss in the maximum angular velocity that each rotor can achieve.

Let p represent the fault magnitude, and let $\boldsymbol{p}_1 = (p, 1, 1, 1)$, $\boldsymbol{p}_2 = (1, p, 1, 1)$, $\boldsymbol{p}_3 = (1, 1, p, 1)$, and $\boldsymbol{p}_4 = (1, 1, 1, p)$ be the fault parameters used to instantiate the system model. Each vector of fault parameters induces a constraint set for the inputs. For instance, for $\boldsymbol{p}_1$, the control inputs must satisfy $0 \leq U_1 \leq b(p+3)\Omega^2_{max}$, $-b\Omega^2_{max} \leq U_2 \leq b\Omega^2_{max}$, $-b\Omega^2_{max} \leq U_3 \leq bp\Omega^2_{max}$, and $-b(p+1)\Omega^2_{max} \leq U_4 \leq 2b\Omega^2_{max}$. Similar inequalities can be determined by the remaining fault parameter vectors. The quadrotor mission is to move from a hovering initial position (1,1,0) to a hovering final position (0,0,2) in under 4 seconds. Here, hovering implies that the angles, linear and angular velocities are zero. The state safe set is defined by the cylinder $(x-1)^2 + (y-1)^2 \leq 4$ and $0 \leq z \leq 3$. We limit the number of change points for the perturbations δu_i to 20 points, resulting in a total of 4×20 optimization variables.

We present the results of the optimization for two cases: $p = 0.4$ and $p = 0.9$. Figures 3, 4, 5, and 6 show the perturbation inputs and simulated outputs under the four fault modes for the two considered fault magnitudes. It is evident that for $p = 0.4$, the differences in outputs for the four fault modes are distinct, whereas this is not the case for $p = 0.9$. This is expected since the effects of the fault modes at this fault magnitude level are not significant. Moreover, the main differences are in the x, y positions and angles, while the z direction remains unaffected across all fault modes. It is important to recall that U_1 controls the altitude z and is upper-bounded by $b(p+3)\Omega^2_{max}$, while U_2 and U_3 are upper-bounded by $bp\Omega^2_{max}$, but not simultaneously. These two inputs control the angles and thus the position in the x, y plane. Consequently, U_2 and U_3 are much more sensitive to the fault effects.

Subsequently, we applied an optimization-based diagnosis using the perturbation inputs for each fault magnitude. We added noise to the simulated output measurements up to 30dB signal-to-noise ratio (SNR). Tables 3 through 14 display the results. For the given noise level, all diagnoses were accurate. The parameters were correctly identified, and the empirical probabilities for the true fault modes were dominant. Figure 7 shows an example of trajectories generated by the perturbed PD feedback control with a fault magnitude of $p = 0.4$ on rotor 1. Notable changes in the quadrotor's angles are observed until it settles in its final position. It is this type of behavior that results in rich output data, ultimately leading to accurate fault diagnosis.

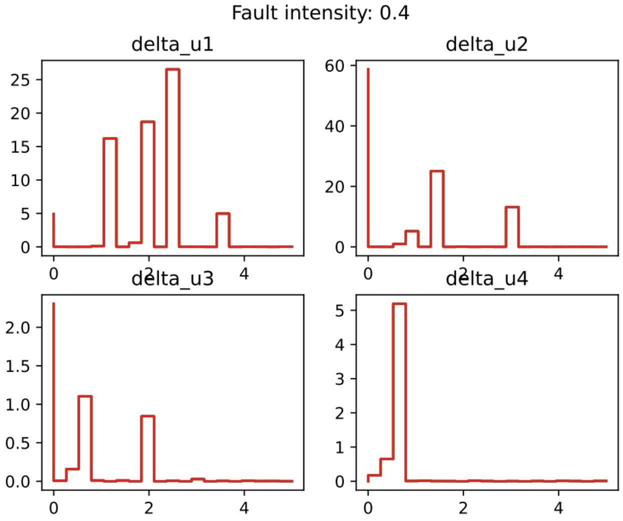

Figure 3. Input perturbations for fault magnitude $p = 0.4$.

Table 3. Fault magnitude 0.4: fault parameter estimates.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.4	0.4	0.865	0.928	0.785
$p2$=0.4	0.875	0.4	0.859	0.872
$p3$=0.4	0.817	0.86	0.4	0.842
$p4$=0.4	0.808	0.806	0.773	0.4

Table 4. Fault magnitude 0.4: empirical probabilities.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.4	0.981	0.006	0.006	0.006
$p2$=0.4	0.01	0.971	0.01	0.01
$p3$=0.4	0.027	0.026	0.921	0.026
$p4$=0.4	0.029	0.029	0.029	0.914

Table 5. Fault magnitude 0.5: fault parameter estimates.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.5	0.5	0.506	0.533	0.504
$p2$=0.5	0.598	0.5	0.535	0.518
$p3$=0.5	0.588	0.494	0.499	0.504
$p4$=0.5	0.591	0.499	0.529	0.5

Table 6. Fault magnitude 0.5: empirical probabilities.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.5	0.964	0.01	0.013	0.012
$p2$=0.5	0.021	0.936	0.023	0.02
$p3$=0.5	0.121	0.017	0.795	0.067
$p4$=0.5	0.057	0.016	0.072	0.855

The accuracy of the diagnosis may be compromised by the amount of noise, particularly when fault magnitudes are low.

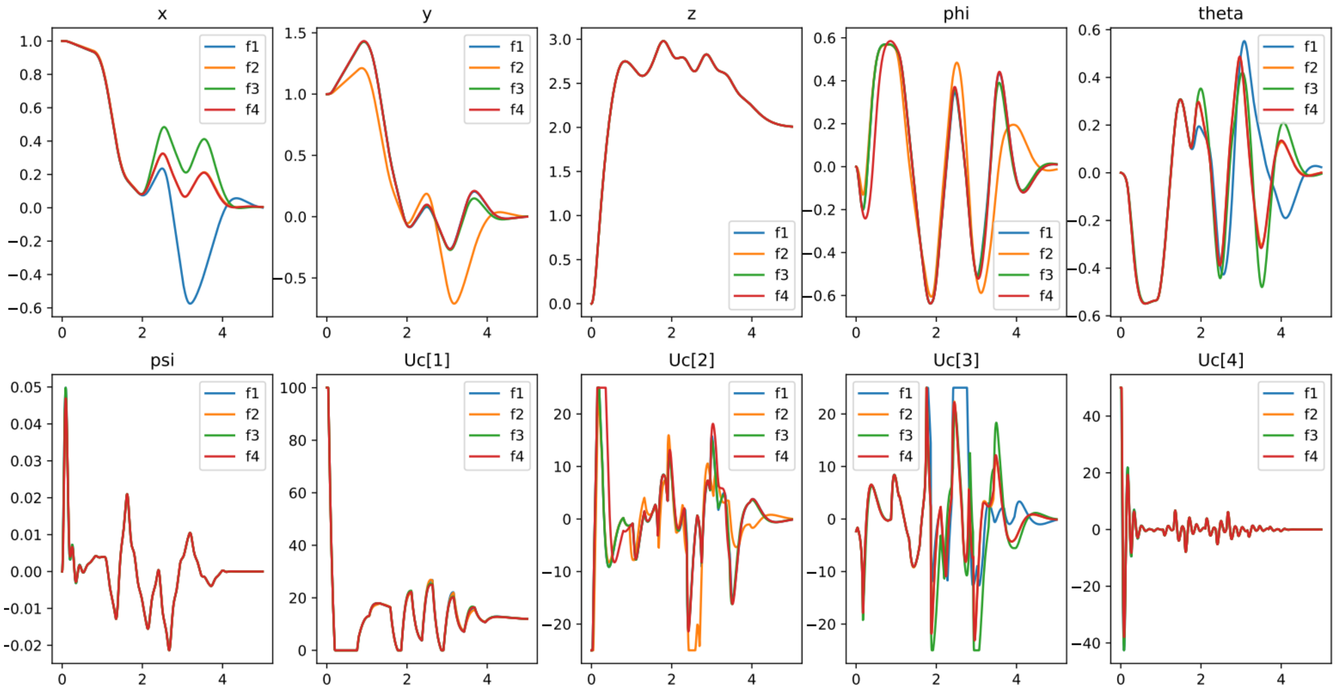

Figure 4. Simulated outputs under single fault rotor scenarios under the effect of the input perturbations for fault magnitude $p = 0.4$.

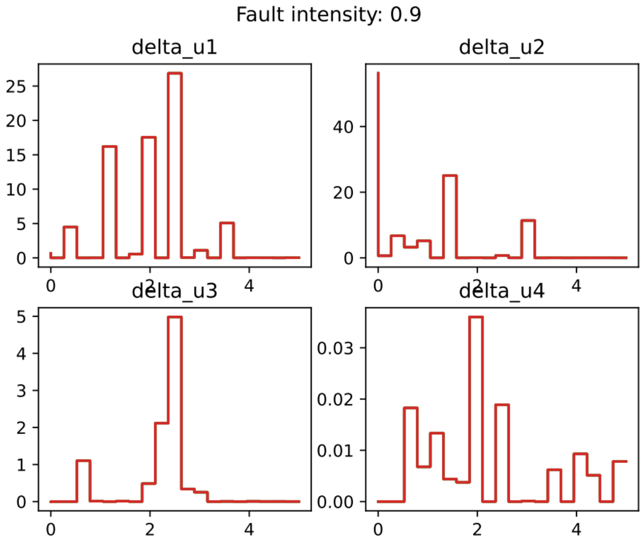

Figure 5. Input perturbations for fault magnitude $p = 0.9$.

Table 7. Fault magnitude 0.6: fault parameter estimates.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.6	0.6	0.667	0.3	0.608
$p2$=0.6	0.706	0.6	0.503	0.672
$p3$=0.6	0.71	0.677	0.6	0.673
$p4$=0.6	0.708	0.673	0.595	0.6

Table 8. Fault magnitude 0.6: empirical probabilities.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.6	0.92	0.022	0.035	0.023
$p2$=0.6	0.054	0.833	0.06	0.053
$p3$=0.6	0.171	0.102	0.557	0.17
$p4$=0.6	0.092	0.071	0.122	0.715

Table 9. Fault magnitude 0.7: fault parameter estimates.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.7	0.7	0.648	0.3	0.617
$p2$=0.7	0.802	0.699	0.671	0.753
$p3$=0.7	0.804	0.782	0.701	0.779
$p4$=0.7	0.803	0.782	0.678	0.7

Table 10. Fault magnitude 0.7: empirical probabilities.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.7	0.922	0.018	0.042	0.018
$p2$=0.7	0.116	0.618	0.143	0.123
$p3$=0.7	0.182	0.181	0.391	0.246
$p4$=0.7	0.143	0.136	0.197	0.525

To investigate this, we reduce the signal-to-noise ratio (SNR)

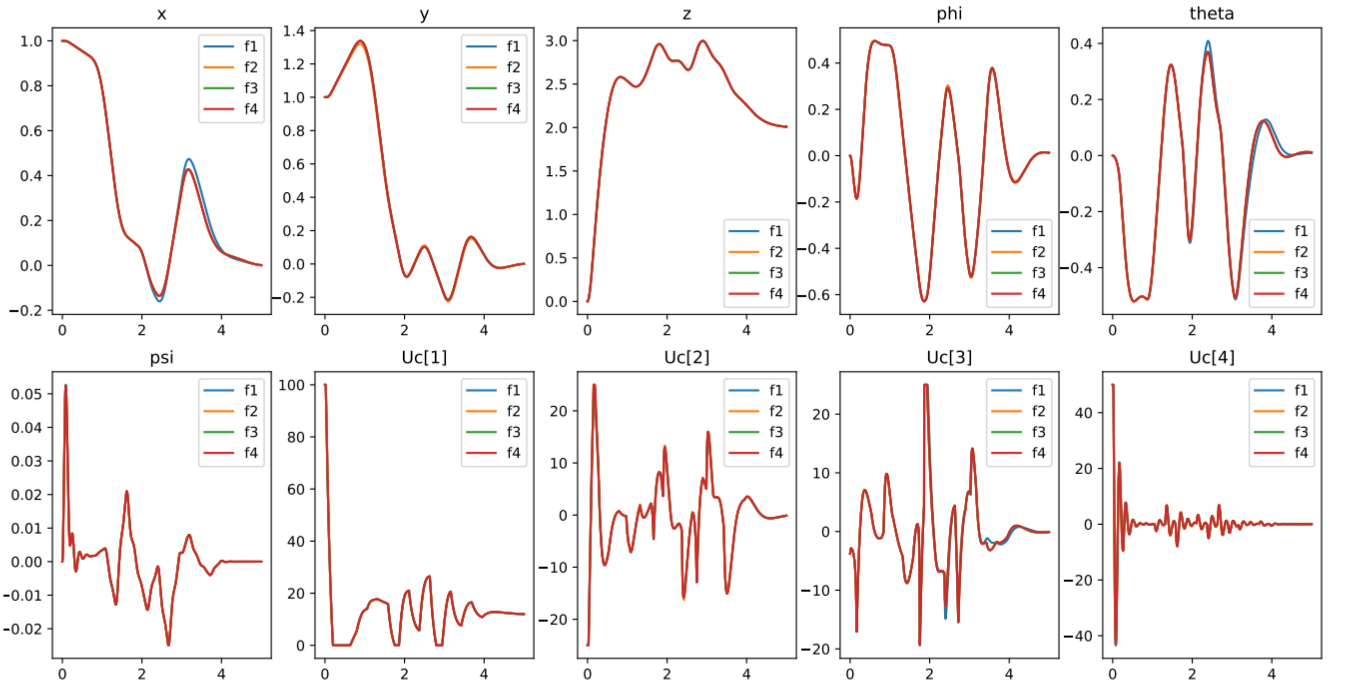

Figure 6. Simulated outputs under single fault rotor scenarios under the effect of the input perturbations for fault magnitude $p = 0.9$.

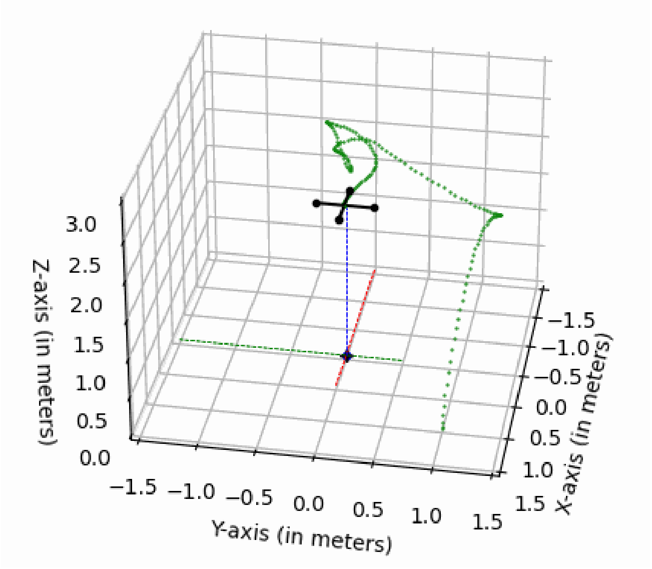

Figure 7. Quadrotor trajectory trace when rotor 1 loses 60% of the maximum velocity it can develop: the effects of the perturbation inputs.

Table 11. Fault magnitude 0.8: fault parameter estimates.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.8	0.8	0.687	0.3	0.663
$p2$=0.8	0.911	0.8	0.756	0.837
$p3$=0.8	0.911	0.852	0.798	0.84
$p4$=0.8	0.911	0.85	0.764	0.801

Table 12. Fault magnitude 0.8: empirical probabilities.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.8	0.895	0.023	0.06	0.023
$p2$=0.8	0.179	0.406	0.216	0.199
$p3$=0.8	0.211	0.212	0.316	0.261
$p4$=0.8	0.196	0.195	0.254	0.355

Table 13. Fault magnitude 0.9: fault parameter estimates.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.9	0.9	0.901	0.773	0.853
$p2$=0.9	0.971	0.902	0.896	0.933
$p3$=0.9	0.973	0.946	0.9	0.937
$p4$=0.9	0.971	0.945	0.893	0.899

Table 14. Fault magnitude 0.9: empirical probabilities.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.9	0.512	0.156	0.17	0.163
$p2$=0.9	0.212	0.295	0.249	0.244
$p3$=0.9	0.221	0.25	0.268	0.261
$p4$=0.9	0.219	0.246	0.261	0.274

to 20 dB and perform the diagnosis for two extreme cases: $p = 0.4$ and $p = 0.9$. The resulting tables are shown in Tables 15 through 18. In all cases, the fault parameters are estimated correctly, but the main impact is on the probabilities of fault modes. Specifically, for $p = 0.9$, all fault modes are nearly equally likely. Since this fault magnitude has a minor impact on the quadrotor's behavior, choosing an incorrect fault mode does not result in any significant consequences.

Table 15. Fault magnitude 0.4: fault parameter estimates under SNR=20dB.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.4	0.400	0.862	0.928	0.786
$p2$=0.4	0.875	0.399	0.848	0.870
$p3$=0.4	0.825	0.860	0.400	0.854
$p4$=0.4	0.810	0.806	0.776	0.398

Table 16. Fault magnitude 0.4: empirical probabilities under SNR=20dB.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.4	0.845	0.052	0.052	0.052
$p2$=0.4	0.135	0.595	0.135	0.135
$p3$=0.4	0.221	0.25	0.268	0.261
$p4$=0.4	0.138	0.138	0.138	0.587

Table 17. Fault magnitude 0.9: fault parameter estimates under SNR=20dB.

	$\hat{p}1$	$\hat{p}2$	$\hat{p}3$	$\hat{p}4$
$p1$=0.9	0.901	0.902	0.773	0.860
$p2$=0.9	0.970	0.901	0.888	0.928
$p3$=0.9	0.971	0.945	0.910	0.939
$p4$=0.9	0.971	0.943	0.902	0.901

Table 18. Fault magnitude 0.9: empirical probabilities under SNR=20dB.

	$q1$	$q2$	$q3$	$q4$
$p1$=0.9	0.289	0.234	0.240	0.237
$p2$=0.9	0.245	0.255	0.250	0.250
$p3$=0.9	0.247	0.250	0.252	0.251
$p4$=0.9	0.246	0.250	0.251	0.253

5. Conclusions

This paper addresses the challenge of ambiguous fault diagnosis in the presence of intermittent faults where sensor measurements do not provide sufficient information to compute fault probabilities with high certainty. Specifically, we focused on the problem of actuator faults and proposed an optimal control approach to disambiguate diagnosis solutions. We considered systems under feedback control and designed perturbations on the control inputs generated by the feedback control loop to reduce the similarity between measurements in ambiguous fault modes. To increase simulation efficiency, we used gradient-free algorithms with functional mock-up units (FMUs) as computational model representations. We ensured that the optimization formulation did not affect the primary control objective and that the system remained in a safe set of states. To control the complexity of the optimization problem, we limited the number of change points of the perturbations. Our approach was demonstrated on a quadrotor example, and we showed that by using disambiguation perturbations together with control inputs generated by the motion control algorithm, we could isolate faults with high probability and accurately estimate their magnitude. These results demonstrate the effectiveness and applicability of our approach for addressing ambiguous fault diagnosis in practical systems.

References

Andersson, C., Akesson, J., & Fuhrer, C. (2016). *PyFMI: A python package for simulation of coupled dynamic models with the functional mock-up interface* (Tech. Rep. No. 2). Lund University.

Andersson, J. A. E., Gillis, J., Horn, G., Rawlings, J. B., & Diehl, M. (2019). CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, *11*(1), 1–36.

Arulampalam, M. S., Maskell, S., & Gordon, N. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, *50*, 174–188.

Avram, R. (2016). *Fault Diagnosis and Fault-Tolerant Control of Quadrotor UAVs* (Unpublished doctoral dissertation). Wright State University.

Blochwitz, T., Otter, M., Arnold, M., Bausch, C., Elmqvist, H., Junghanns, A., … Clauß, C. (2011). The functional mockup interface for tool independent exchange of simulation models. In *Proceedings of the 8th international modelica conference* (p. 105-114).

Bouabdallah, S., & Siegwart, R. (2007, Oct). Full control of a quadrotor. In *2007 ieee/rsj international conference on intelligent robots and systems* (p. 153-158). doi: 10.1109/IROS.2007.4399042

de Kleer, J., Mackworth, A., & Reiter, R. (1992). Characterizing diagnoses and systems. *"Journal of Artificial Inteligence"*, *56*(2–3), 197–222.

Garcia, C. E., Prett, D. M., & Morari, M. (1989). Model predictive control: Theory and practice - A survey. *Automatica*, *25*(3), 335 - 348.

Gertler, J. (1998). *Fault-detection and diagnosis in engineering systems*. New York: Marcel Dekker.

Gordon, N., Salmond, D., & Smith, A. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proc. F Radar Signal Process. UK*, *140*(2), 107.

Guzmán-Rabasa, J. A., López-Estrada, F. R., González-Contreras, B. M., Valencia-Palomo, G., Chadli, M., & Pérez-Patricio, M. (2019). Actuator fault detection and isolation on a quadrotor unmanned aerial vehicle modeled as a linear parameter-varying system. *Measurement and Control*, *52*(9-10), 1228-1239.

Isermann, R. (2005). Model-based fault-detection and diagnosis - status and applications. *Annual Reviews in Control*, *29*(1), 71 - 85.

Julier, S. J., & Uhlmann, J. K. (1997, July). New extension of the Kalman filter to nonlinear systems. In I. Kadar (Ed.), *Signal processing, sensor fusion, and target recognition vi* (Vol. 3068, p. 182-193).

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, *82*(Series D), 35–45.

Matei, I., Zhenirovskyy, M., de Kleer, J., & Goebel, K. (2022). A control approach to fault disambiguation. *Annual Conference of the PHM Society*, *14*(1).

McElhoe, B. A. (1966, July). An Assessment of the Navigation and Course Corrections for a Manned Flyby of Mars or Venus. *IEEE Transactions on Aerospace Electronic Systems*, *2*(4), 613-623.

Patton, R. J., Frank, P. M., & Clark, R. N. (2000). *Issues of fault diagnosis for dynamic systems*. Springer-Verlag London.

Staroswiecki, M. (2000). Quantitative and qualitative models for fault detection and isolation. *Mechanical Systems and Signal Processing*, *14*(3), 301 - 325.

Staroswiecki, M., & Comtet-Varga, G. (2001). Analytical redundancy relations for fault detection and isolation in algebraic dynamic systems. *Automatica*, *37*(5), 687 - 699.