Parity-Based Diagnosis in UAVs: Detectability and Robustness Analyses

Georgios Zogopoulos-Papaliakos and Kostas J. Kyriakopoulos

Abstract— **Parity-Based methodologies for fault diagnosis in UAVs often result in nonlinear residual generators. Still, a systematic framework to perform detectability and robustness analyses of residual generators does not exist. In this work, detectability and robustness metrics for static and dynamic residuals are presented, while numerical methods, specifically Particle Swarm Optimization, are employed to calculate them. The results are used to characterize the performance of a fault detection system. An application on a UAV model is shown, based on real flight data.**

I. Introduction

The state of the art in Unmanned Aerial Vehicle (UAV) automation has mostly achieved estimation and control requirements. Modern efforts for increased autonomy are pushing towards fault diagnosis, which focuses on Fault Detection and Isolation (FDI). Indeed, embedding UAVs into the civilian airspace calls for enhanced safety requirements. These can be satisfied if UAVs can monitor their health status, detect faults and prevent failures [1].

Fault diagnosis techniques are majorly categorized into model-based and data-based. The latter utilize existing process history and are preferred when large amounts of logged data of faulty conditions are available. They include Neural Networks, Pattern Recognition and Statistical Classification [2], [3]. However, they are not applicable in the design stage of a UAV, or if there are not enough experimental data.

On the other hand, model-based techniques can be applied on a mathematical model of the vehicle, which is usually already available on the design stage. This can be advantageous, as the capabilities of the diagnostic system can be explored earlier and iterations can be faster. The most prominent techniques of this category are Parameter Estimation, Observer-based methods and Parity-space methods [2]–[4]. In this work, we shall focus on model-based methodologies.

Traditionally, most attempts towards model-based UAV FDI refer to linear models [5]–[10]. Closed-form, optimal solutions for specified performance exist in these cases.

More recently, works which are based on non-linear models have also been presented. These commonly involve manual design and tuning of diagnostic observers. The resulting solutions are well-behaved and highly performant, but are also cumbersome to derive and their design cannot be automated [11]–[14].

More systemic approaches for the design of FDI systems do exist: Parity-space is the most notable example [8]. By exploiting *analytical redundancy* and rearranging the system equations, residual expressions (also known as *Analytically Redundant Relations (ARR)* or *residual generators*) can be formulated, producing residual signals. Comparing residuals against static or adaptive thresholds leads to a diagnostic decision.

For linear systems, it has been proven that Parity-space methods are equivalent to other approaches [15]. For algebraic, non-linear systems, variable elimination techniques [11], [16] have been used with limited success. The problem is further complicated by the large number of equations found in detailed UAV models [17]. Recently, a qualitative approach is preferred: a graph abstraction of the system, the *structural graph* [1], is constructed and graph algorithms are employed to extract structural residual generators [18].

A. Sensitivity and Robustness

In general, Parity-space residual generators are made available in the form of complex, non-linear (and potentially non-analytic) functions of state, inputs, measurements, faults and disturbances. In such cases, quantifying the influence of the faults to a residual signal is not trivial [18]. Additionally, noise and uncertainty contribute a stochastic or causal component to the residual, leading to non-zero values in fault-free situations and altering its response in faulty situations. Consequently, the usefulness of a residual generator and the appropriate threshold selection become obscure. This is the motivating force of this work: acquiring a quantitative metric of the contribution of faults, noise and uncertainty to non-linear residual signals.

To that goal, further analyses are required: detectability analysis to determine the actual fault contribution in a residual signal and robustness analysis to assess the contribution of noise and uncertainty. The resulting metrics are indicative of the residual signal performance, noise rejection capabilities and enable an educated selection of detection thresholds.

In a repeating pattern, publications addressing detectability and robustness analyses or a combination of both already exist for linear systems [4], [9], [10], [19], [20]. For non-linear systems, only frameworks other than Parity-space have tackled this problem. In [21], logged fault responses were used as basis of an optimization problem, forming a residual residual signal with maximum sensitivity and robustness simultaneously. In [22] non-linear observers were combined with H_∞ methods to introduce robustness to the FDI system.

B. Contribution

In this work we tackle the problems of characterizing the detectability and robustness performance of non-linear ARRs, commonly found in detailed UAV models. We derive quantitative corresponding metrics by employing numerical

The authors are with Control Systems Lab, School of Mechanical Engineering, National Technical University of Athens, Greece {gzogop, kkyria}@mail.ntua.gr

methods, specifically Particle Swarm Optimization. Finally, we showcase our methods on a model of a fixed-wing UAV and real logged data.

II. Modeling of Residual Expressions

A. UAV Model

The aircraft models presented in FDI literature are usually compact and idealized, or even contain only part of the vehicle dynamics, so that concepts can be better conveyed. However, a practical FDI system based on analytical redundancy can only be as detailed as the system model underlying it. Consequently, with the intention of supporting an arbitrarily high level fault isolation resolution, we consider a large-scale model of a UAV, implemented as an interconnected set of subsystems affected by their own faults. We choose to work with a fixed-wing form factor, but the methods presented in this work could be applied in multirotor models as well. Typically, rigid-body dynamics, aerodynamics, propulsion, actuators, sensors and payloads are incorporated into a UAV model.

There is a multitude of methodologies which can be used to derive the involved submodels. Most commonly, mathematical modeling uses first principles, manufacturer data (tabular data, provided coefficients etc.) and model identification procedures. All subsystems are lumped into a set of N, generally non-linear, equations:

$$\mathcal{G}=\{h_i(\boldsymbol{y},\boldsymbol{u},\boldsymbol{x},\dot{\boldsymbol{x}},\boldsymbol{d},\boldsymbol{f})=0, \quad i=1,...,N\} \tag{1a}$$
$$\boldsymbol{x}\in\mathbb{D}_x, \dot{\boldsymbol{x}}\in\mathbb{D}_{\dot{x}}, \boldsymbol{y}\in\mathbb{D}_y, \boldsymbol{u}\in\mathbb{D}_u, \boldsymbol{d}\in\mathbb{D}_d, \boldsymbol{f}\in\mathbb{D}_f \tag{1b}$$

where $\boldsymbol{y}$ is the measurements vector over all the submodels, $\boldsymbol{u}$ is the inputs vector, $\boldsymbol{x}$ is the vector of internal variables, $\boldsymbol{d}$ are disturbances and $\boldsymbol{f}$ are fault variables. $\mathbb{D}_x$, $\mathbb{D}_{\dot{x}}$, $\mathbb{D}_y$, $\mathbb{D}_u$, $\mathbb{D}_d$ and $\mathbb{D}_f$ are the domains of each corresponding variable vector. Auxiliary state variables can be introduced so as only first derivatives of variables appear in (1), which has a form known as a Differential-Algebraic Equation system (DAE).

Assumption 1: The domains $\mathbb{D}_x$, $\mathbb{D}_{\dot{x}}$, $\mathbb{D}_y$, $\mathbb{D}_u$, $\mathbb{D}_d$ and $\mathbb{D}_f$ are convex.

Due to lack of space we shall not include a full example of a model with so many equations. The reader is directed to our previous work [23], where a fixed-wing UAV model is presented, with more than a hundred equations and variables.

B. Fault Modeling

In the context of the presented methodology, no disturbance and fault modeling is performed, other than specifying them as unknown inputs, additive to the equations they contribute and lying within their predefined domains [18].

C. Residual Generator Formulation

Using the equations of (1) and employing existing methodologies [24], [25], subsets of equations can be formed (under conditions not covered in this work) which consist of a square equation system and one more ARR. By solving the system and substituting variable values into the ARR, the residual generator is evaluated.

In the simplest case, residual generators are static systems:

$$\boldsymbol{0}=\boldsymbol{h}_a(\boldsymbol{x}_a,\boldsymbol{z},\boldsymbol{d},\boldsymbol{f}) \tag{2a}$$
$$r=h_r(\boldsymbol{x}_a,\boldsymbol{z},\boldsymbol{d},\boldsymbol{f}) \tag{2b}$$

where $\boldsymbol{h}_a$ is a known square system of algebraic equations, which can be solved for its variables $\boldsymbol{x}_a$, given inputs, and measurements (combined in a concatenated vector $\boldsymbol{z}$), disturbances and faults. h_r is the residual generator expression, evaluated for the residual signal r.

Incorporating the solution of $\boldsymbol{x}_a$ into (2b), we get the lumped expression:

$$r=h_r^*(\boldsymbol{z},\boldsymbol{d},\boldsymbol{f})=0 \tag{3}$$

Note that since (2b) reflects the real system, including disturbances and faults, $h_r^*=0$ is always satisfied: it is a *compatibility condition* for $(\boldsymbol{z},\boldsymbol{d},\boldsymbol{f})$ and a manifold for these variables is formed:

$$\mathbb{M}=\{(\boldsymbol{z},\boldsymbol{d},\boldsymbol{f}):h_r^*(\boldsymbol{z},\boldsymbol{d},\boldsymbol{f})=0\} \tag{4}$$

On the other hand, a real-world, implemented FDI system doesn't have access to the unknowns $\boldsymbol{d}$ and $\boldsymbol{f}$. The hatted counterparts of $\boldsymbol{h}_a$ and h_r, which represent the ideal and healthy system, are formed by setting $\boldsymbol{d}=\boldsymbol{f}=\boldsymbol{0}$.

$$\boldsymbol{0}=\hat{\boldsymbol{h}}_a(\hat{\boldsymbol{x}}_a,\boldsymbol{z}), \quad \boldsymbol{z}\in\mathbb{M} \tag{5a}$$
$$\hat{r}=\hat{h}_r(\hat{\boldsymbol{x}}_a,\boldsymbol{z}) \tag{5b}$$

Thus solving $\hat{\boldsymbol{h}}_a$ for the fault-free, disturbance-free estimate $\hat{\boldsymbol{x}}_a$ and substituting into $\hat{h}_r$ will form $\hat{h}_r^*$ (equivalent to (3)) and result in a $\hat{r}$ which equals 0 only when the actual system is fault- and disturbance-free as well. This is the core of the Parity-space approach.

For the dynamic case, the following assumption is made:

Assumption 2: All dynamic residual generators are of Index-0 or Index-1 Differential-Algebraic Equations (DAE). This might be a rather strong assumption, but it is necessary for all submodels of $\mathcal{G}$ to be solved by readily available software libraries, at least numerically, if not analytically [26], [27]. Even if the original system does not meet this requirements, in [23] we have shown how a system model can be edited so that this assumption is guaranteed.

The dynamic residual then takes the form:

$$\dot{\boldsymbol{x}}_d=\boldsymbol{h}_d(\boldsymbol{x}_d,\boldsymbol{x}_a,\boldsymbol{z},\boldsymbol{d},\boldsymbol{f}), \quad (\boldsymbol{x}_d,\boldsymbol{x}_d,\boldsymbol{x}_a,\boldsymbol{z},\boldsymbol{d},\boldsymbol{f})\in\mathbb{M} \tag{6a}$$
$$0=\boldsymbol{h}_a(\boldsymbol{x}_d,\boldsymbol{x}_a,\boldsymbol{z},\boldsymbol{d},\boldsymbol{f}) \tag{6b}$$
$$r=h_r(\boldsymbol{x}_d,\dot{\boldsymbol{x}}_d,\boldsymbol{x}_a,\boldsymbol{z},\boldsymbol{d},\boldsymbol{f}) \tag{6c}$$

where $\boldsymbol{x}_d$ is the state variables vector, $\boldsymbol{x}_a$ is the algebraic variables vector and $\mathbb{M}$ is defined similarly to (4).

Similarly to the algebraic case (5), the actual implemented FDI system has the form:

$$\dot{\hat{\boldsymbol{x}}}_d=\hat{\boldsymbol{h}}_d(\hat{\boldsymbol{x}}_d,\hat{\boldsymbol{x}}_a,\boldsymbol{z}), \quad \boldsymbol{z}\in\mathbb{M} \tag{7a}$$
$$0=\hat{\boldsymbol{h}}_a(\hat{\boldsymbol{x}}_d,\hat{\boldsymbol{x}}_a,\boldsymbol{z}) \tag{7b}$$
$$\hat{r}=\hat{h}_r(\hat{\boldsymbol{x}}_d,\dot{\hat{\boldsymbol{x}}}_d,\hat{\boldsymbol{x}}_a,\boldsymbol{z}) \tag{7c}$$

We add one more assumption, whose purpose will become evident in Section IV:

Assumption 3: For any constant $\boldsymbol{z}$, the (now autonomous) dynamic system (7) is driven to an equilibrium point, implying that $\forall \boldsymbol{z} \lim_{t\to\infty} \dot{\hat{x}}_d{=}0$, $\lim_{t\to\infty}(\hat{\boldsymbol{x}}_d, \hat{\boldsymbol{x}}_a){=}(\bar{\boldsymbol{x}}_d, \bar{\boldsymbol{x}}_a)$ and $\lim_{t\to\infty} r{=}c \in \mathbb{R}$.

Remark: The above assumption is not uncommon in dynamic subsystems found in UAVs. One such example is the UAV motor subsystem, which reaches constant RPM under constant input command. Another example is the longitudinal velocity of a UAV, which eventually becomes constant for a given propeller thrust. A dynamic subsystem which does not satisfy this assumption is the altitude of the UAV, which does not come to an equilibrium for a constant vertical velocity; instead it maintains a constant rate of change and the residuals corresponding to such subsystems cannot be analyzed under the present methodology.

Additionally, without loss of generality regarding detectability and robustness analyses, for the rest of this work each residual generator shall be subject to only one fault.

D. Problem Formulation

Informally, the two problems tackled in this work can be formulated as follows: Given a residual generator $\hat{h}_r$, 1) what is its sensitivity to fault f and 2) what is its sensitivity to disturbances $\boldsymbol{d}$ in absence of fault ($f = 0$).

III. Residual Analysis

A. Robustness Analysis

As discussed previously, we seek a quantitative expression for the disturbance response of the residual. This is helpful in finding a minimum threshold value, below which the residual is expected to fluctuate, even in absence of faults.

Generally, the sensitivity of a residual to disturbances is expressed as $\partial\hat{r}/\partial\boldsymbol{d}$. For the case of non-linear systems, the evaluation of the partial derivative is difficult, especially finding an analytical expression, which may be a function of $\boldsymbol{x}$, $\boldsymbol{z}$ and $\boldsymbol{d}$. Even worse, (5) or (7) may be very large (tens of equations)

Keeping in mind that the end-goal is to quantify the disturbance response of the residual, a slightly different approach is chosen. Instead of calculating the partial derivative of the residual to the disturbance, a bound for absolute residual response ($|\hat{r}|$) is sought directly [4]. For that purpose S_d^+ is defined as the absolute maximum residual response over all operating conditions under maximum disturbance influence and in absence of faults.

1) Static Systems: First, the (worst-case) *maximum disturbance function* is defined as the supremum of the residual over the measurements $\boldsymbol{z}$:

$$r_{\boldsymbol{d}}^+(\boldsymbol{d}) = \sup_{\boldsymbol{z}} \left(\left|\hat{h}_r^*(\boldsymbol{z})\right|\right), \quad (\boldsymbol{z}, \boldsymbol{d}, \boldsymbol{f} = 0) \in \mathbb{M} \tag{8}$$

Then, the *maximum disturbance response* is then be defined as the supremum of $r_{\boldsymbol{d}}^+(\boldsymbol{d})$ over the disturbance range:

$$S_{\boldsymbol{d}}^+ = \sup_{\boldsymbol{d}}(r_{\boldsymbol{d}}^+(\boldsymbol{d})), \quad (\boldsymbol{z}, \boldsymbol{d}, \boldsymbol{f} = 0) \in \mathbb{M} \tag{9}$$

It is emphasized that in both optimization steps ($\boldsymbol{z}$, $\boldsymbol{d}$) must lie in the manifold $\mathbb{M}$, i.e. satisfy the compatibility condition ($h^*(\boldsymbol{z}, \boldsymbol{d}, \boldsymbol{f} = 0) = 0$).

2) Dynamic Systems: Dynamic systems are more complicated to handle, since the residual response is also a function of time (t). The lumped form of (7c), $\hat{h}_r^*$, contains the solution of $\hat{\boldsymbol{x}}_d$ (the DAE state variables) which depends on the estimated initial state. It can be re-written as

$$\hat{r}(t) = \hat{h}_r^*(t, \boldsymbol{z}(t), \hat{\boldsymbol{x}}_0) \tag{10}$$

and it becomes evident that the maximum value of each realization of $r(t)$ is of interest. The infinite norm is used to reflect this. Consequently, the dynamic equivalents of (8) and (9) are defined as:

$$r_{\boldsymbol{d}}^+(\boldsymbol{d}(t)){=}\sup_{\boldsymbol{z}(t),\boldsymbol{x}_0}\left(\left\|\left|\hat{h}_r^*(t, \boldsymbol{z}(t), \hat{\boldsymbol{x}}_0)\right|\right\|_\infty\right) \tag{11a}$$

$$S_{\boldsymbol{d}}^+{=}\sup_{\boldsymbol{d}(t)}(r_{\boldsymbol{d}}^+(\boldsymbol{d}(t)) \tag{11b}$$

$$(\boldsymbol{z}(\,t\,), \boldsymbol{d}(t)) \in \mathbb{M}\ \forall t \tag{11c}$$

B. Detectability Analysis

In the faulty case, the fault variable takes non-zero values ($f \neq 0$) while disturbances continue to manifest. Detectability analysis can be covered by the exact same framework as robustness analysis. However, there are now two target metrics of interest.

Firstly, we present the residual response over all operating conditions and under maximum fault influence S_f^+. This is the equivalent to $S_{\boldsymbol{d}}^+$ and is defined as:

$$r_f^+(f) = \sup_{\boldsymbol{z},\boldsymbol{d}}\left(\left|\hat{h}_r^*(\boldsymbol{z})\right|\right), \quad (\boldsymbol{z}, \boldsymbol{d}, f) \in \mathbb{M} \tag{12a}$$

$$S_f^+ = \sup_f(r_f^+(f)), \quad (\boldsymbol{z}, \boldsymbol{d}, f) \in \mathbb{M} \tag{12b}$$

Equally important is the worst case residual performance (small response) while under maximum fault influence (S_f^-).

$$r_f^-(f) = \inf_{\boldsymbol{z},\boldsymbol{d}}\left(\left|\hat{h}_r^*(\boldsymbol{z})\right|\right), \quad (\boldsymbol{z}, \boldsymbol{d}, f) \in \mathbb{M} \tag{13a}$$

$$S_f^- = \sup_f(r_f^-(f)), \quad (\boldsymbol{z}, \boldsymbol{d}, f) \in \mathbb{M} \tag{13b}$$

This metric is crucial because it reveals operating points where the influence of the fault to the residual is smaller or non-existent, due to non-linearity. It is notable that both S_f^+ and S_f^- encompass a maximization over the fault domain to capture the maximum fault response, which may not correspond to f_{min} or f_{max}.

Similarly, for the dynamic case the detectability metrics are:

$$r_f^-(f(t)){=}\inf_{\boldsymbol{z}(t),\boldsymbol{d}(t),\boldsymbol{x}_0}\left(\left\|\left|\hat{h}_r^*(t, \boldsymbol{z}(t), \boldsymbol{d}, \hat{\boldsymbol{x}}_0)\right|\right\|_\infty\right) \tag{14a}$$

$$r_f^+(f(t)){=}\sup_{\boldsymbol{z}(t),\boldsymbol{d}(t),\boldsymbol{x}_0}\left(\left\|\left|\hat{h}_r^*(t, \boldsymbol{z}(t), \boldsymbol{d}, \hat{\boldsymbol{x}}_0)\right|\right\|_\infty\right) \tag{14b}$$

$$S_f^-{=}\sup_{f(t)}(r_f^-(f(t)) \tag{14c}$$

$$S_f^+{=}\sup_{f(t)}(r_f^+(f(t)) \tag{14d}$$

$$(\boldsymbol{z}(\,t\,), \boldsymbol{d}(t), f(t)) \in \mathbb{M}\ \forall t \tag{14e}$$

Detectability and robustness metrics are valuable; in the scope of a single residual, they provide insight on the Signal-to-Noise ratio and allow educated choice of detection thresholds. Additionally, they allow for performance comparison among multiple residuals sensitive to the same fault.

IV. Numerical Implementation

As it was stated in the previous section, trying to analytically calculate the suprema and infima of the sensitivity and robustness metrics, for multi-variable non-linear functions or systems is unrealistic. Instead, a numerical approach can be pursued, allowing the analysis of residuals to be performed systematically, even automatically, greatly facilitating FDI system design.

Starting from detectability analysis, it is obvious that the required maximization of (12) is a multi-variable, non-continuous, non-linear optimization problem. While traditional optimization methods could be used to tackle the optimization, the large dimension of the search space (possibly tens of variables) imposes severe calculation costs. Moreover, the cost function may not be sufficiently smooth for gradient-based methods.

Particle Swarm Optimization (PSO) is chosen to solve the maximization problem [28]. According to this heuristic optimization method, a set of particles is considered, with coordinates $\boldsymbol{x}_n \in \mathbb{D}_x$ each, initialized randomly over the domain. A random initial velocity $\boldsymbol{v}_n$ is also assigned to each particle. Each particle can evaluate the function under optimization on its position and remember the lowest cost it ever met and the coordinates this happened $\boldsymbol{p}_n$. During each iteration, every particle establishes communication with a neighbourhood of K other particles, randomly selected with replacement, with which it shares its best coordinates. The coordinates with the lowest cost for the whole neighbourhood are $\boldsymbol{g}_n$.

The equations of motion for each particle are, for every dimension d of the particle position $\boldsymbol{x}_n$

$$v_n^d \leftarrow c_1 v_n^d + c_2 r(p_n^d - x_n^d) + c_2 r(g_n^d - x_n^d) \tag{15a}$$
$$x_n^d \leftarrow x_n^d + v_n^d \tag{15b}$$

where c_1 is the particle self-confidence (inertia), c_2 is a parameter representing confidence to already existing lower costs, and r is a uniformly random variable in the domain $[0, 1]$.

In case a particle is about to leave its domain along the direction of a dimension, its position is set to

$$x_n^d \leftarrow min(max(x_n^d + v_n^d, x_{min}^d), x_{max}^d) \tag{16}$$

and its speed is set to zero.

This method is favored because:

- It does not require the existence of a gradient or even continuity for the functions of $\mathcal{G}$, removing the assumption of smoothness which may be problematic in the context of highly non-linear, possibly piece-wise residual generators.
- Domain constraints for $\mathbb{D}_{\boldsymbol{z}}$, $\mathbb{D}_f$, $\mathbb{D}_{\boldsymbol{d}}$ can be inherently handled.
- Optimization constraints can be incorporated, e.g. (3).
- It has tuning parameters with explicit meaning.
- It is quite robust, in the sense that broad variations in the optimization parameters do not prevent convergence.

All of (8)-(13) are optimizations with equality constraints, implied by the manifold $\mathbb{M}$. In PSO, soft constraints can be used to enforce equality constraints [29]:

$$J_c = h_r^*(\boldsymbol{z}, \boldsymbol{d}, f) \tag{17}$$

In the simple case of (12), where the residual generator is a purely algebraic non-linear system, one can write:

$$S_f^+ = \underset{\boldsymbol{z},\boldsymbol{d},f}{max} \left(\left| \hat{h}_r^*(\boldsymbol{z}) \right| - k_1(J_c) \right) \tag{18}$$

The calculation of S_f^- is more complicated, because it constitutes a minimax problem, instead of a simple maximization:

$$S_f^- = \underset{f}{max} \left(\underset{\boldsymbol{z},\boldsymbol{d}}{min} \left(\left| \hat{h}_r^*(\boldsymbol{z}) \right| + k_1(J_c) \right) - k_2(J_c) \right) \tag{19}$$

k_1 and k_2 are convex functions defined in [29]. The inner minimization is solved by PSO, while the outer maximization, a single-variable optimization, can be delegated to a more typical algorithm, e.g. Matlab's `fmincon`.

For the case where the residual generator is a dynamic system (a DAE at the most general), the optimization problem is more involved. Notice how the optimizations (14) are actually over the infinite domain of all possible signals $f(t)$, $\boldsymbol{d}(t)$ and $\boldsymbol{z}(t)$. Even if a time interval $[t_0, t_d]$ is discretized and tackled numerically, the explosion in optimization dimension in considerable.

A possible discount in the search space would be to assume constant values for $\boldsymbol{u}$, $\boldsymbol{d}$ and f, solve the DAE to generate compatible outputs $\boldsymbol{y}(t)$ (such that $(\boldsymbol{u}, \boldsymbol{y}(t), f) \in \mathbb{M}\ \forall t \in [t_0, t_d]$) and construct the residual signal. However, the dependence on $[t_0, t_d]$ is still significant: it should be long enough for fault dynamics to manifest.

This approach is also rejected because a) solving the DAE is too costly in the context of thousands of evaluations required by the PSO sampling and b) in the FDI context the separation between input and output variables is not clear in regard to the residual generator expression: both user inputs and system measurements are inputs to the FDI system.

Instead, a compromise is made where the residual response of DAEs is examined only on steady-state conditions (i.e. $\dot{\boldsymbol{x}}_d$=0). This at least provides a quantification of the steady-state residual response. The existence of such equilibrium is guaranteed by Assumption 3.

For each sampled $\boldsymbol{z}$, $\boldsymbol{d}$ and f, a solution pair $(\bar{\boldsymbol{x}}_d, \bar{\boldsymbol{x}}_a)$ is found by solving the system:

$$0 = \boldsymbol{h}_a(\boldsymbol{x}_d, \boldsymbol{x}_a, \boldsymbol{z}, \boldsymbol{d}, f) \tag{20a}$$
$$0 = \boldsymbol{h}_d(\boldsymbol{x}_d, \boldsymbol{x}_a, \boldsymbol{z}, \boldsymbol{d}, f) \tag{20b}$$

The solution exists because of Assumption 3 and the fault response metrics are formed as:

$$J_c = h_r^*(\boldsymbol{z}, \boldsymbol{d}, f, \hat{\boldsymbol{x}}_{d,0} = \bar{\boldsymbol{x}}_d) \tag{21a}$$
$$S_f^+ = \underset{\boldsymbol{z},\boldsymbol{d},f}{max} \left(\left| \hat{h}_r^*(\boldsymbol{z}, \hat{\boldsymbol{x}}_{d,0}{=}\bar{\boldsymbol{x}}_d) \right| - k_1 J_c \right) \tag{21b}$$

$$S_f^- = \max_{f}\left(\min_{\boldsymbol{z},\boldsymbol{d}}\left(\left|\hat{h}_r^*(\boldsymbol{z},\hat{\boldsymbol{x}}_{d,0}{=}\bar{\boldsymbol{x}}_d)\right|+k_1 J_c\right)-k_2 J_c\right) \quad (21c)$$

The aforementioned PSO implementation for detectability analysis is directly applicable to robustness analysis, as shown in the previous section. In fact, the related calculation cost is smaller, because setting $f=0$ reduces the optimization dimension by 1.

A. Explicit Differentiation - An Exception

Consider the small system of a residual generator:

$$r{=}y-\dot{x}-d \quad (22a)$$

$$x{=}u+f \quad (22b)$$

Despite appearances, this is not a DAE, because the state variable x is calculated from the input u and the fault in (22b), differentiated and substituted in (22a), where it is compared with the other input y and the disturbance d. It is possible to systematically check if a residual generator implies a DAE, but it is out of the scope of this work.

In such subsystems, the use of the simple Implicit Euler formula gives rise to algebraic systems, which can be solved by substitution. In this example, past values of u are needed to calculate the residual $r_k=\hat{r}^*$, which are available anyway.

More importantly, it is evident that the residual is sensitive to the derivative of the fault and not its instantaneous value. This is a common result which forces us to sample consecutive values of the differentiated variables when performing PSO, to account for the effect of the derivative.

This discretization technique can also be embedded in dynamic subsystems which include input derivatives, adding algebraic variables to the DAE.

V. Case Study

In this Section, an algebraic subsystem of a fixed-wing UAV is employed to demonstrate the concepts and algorithms, intentionally small enough to be manually tractable. It covers, among others, a fault in the Angle-of-Sideslip (AoS) sensor. The first equation is the AoS definition, the second is part of the rigid-body lateral kinematics and the third is the accelerometer measurement in the body y-axis.

$$v=\sin(\beta_f)(V_{a,f}) \quad (23a)$$

$$\dot{v}=F_y/m+ru+pw \quad (23b)$$

$$a_y=F_y/m-\sin(\phi_f)\cos(\theta_f)g \quad (23c)$$

β is the Angle-of-Sideslip, V_a is the airspeed, ϕ and θ are the roll and pitch Euler angles, $[u,v,w]$ is the Body-frame inertial velocity vector, F_y is the lateral force, p and r are the Body-frame angular velocities along the x and z axes and a_y is the Body-frame lateral acceleration.

a_y, p, r, u, w, g and m are quantities and parameters which are known or measured, but subject to disturbances or uncertainty. For example, the available measurement for the roll rate is $\tilde{p}{=}p{+}d_p$, with d_p being the corresponding disturbance. $V_{a,f}$, β_f, ϕ_f and θ_f are also measured, but the corresponding sensors are subject to faults, e.g. $\beta_f{=}\tilde{\beta}{+}f_\beta$. The algebraic variables vector is $\boldsymbol{x}_a=[v,\dot{v},F_y]$.

TABLE I

VARIABLE DOMAINS

β	-0.79 - 0.79	θ	-0.35 - 0.35	w	-5 - 5
V_a	20 - 35	a_y	-15 - 15	p	-2 - 2
ϕ	-1.05 - 1.05	u	20 - 35	r	-2 - 2

TABLE II

FAULT AND DISTURBANCE DOMAINS

f_β	-1.57 - 1.57	d_{V_a}	-2 - 2	d_w	-1 - 1
f_{V_a}	-5 - 5	d_ϕ	-0.0349 - 0.0349	d_p	-0.2 - 0.2
f_ϕ	-0.087 - 0.087	d_θ	-0.0035 - 0.0035	d_r	-0.15 - 0.15
f_θ	-0.087 - 0.087	d_{a_y}	-2 - 2	d_m	-0.2 - 0.2
d_β	-0.0262 - 0.0262	d_u	-2 - 2	d_g	-0.05 - 0.05

In this overdetermined system any equation can be used as a residual generator, based on the square 2x2 system that the other two equations form. We select (23b) as the residual generator equation. The residual derivation encompasses explicit differentiation.

The residual generator $\hat{h}_r^*$ is formed:

$$\begin{aligned}&\hat{h}_r{=}\dot{v}-F_y/m-ru-pw\\ \Rightarrow\ &\hat{h}_r^*{=}\frac{sin\beta_{f,k}V_{a,f,k}-sin\beta_{f,k-1}V_{a,f,k-1}}{dt}\\ &\qquad-\tilde{a}_{y,k}-sin(\phi_{f,k})cos(\theta_{f,k})-\tilde{r}_k\tilde{u}_k-\tilde{p}_k\tilde{w}_k\end{aligned} \quad (24)$$

Proceeding with fault sensitivity and robustness analyses, the specified variable domains are shown in Table I and domains for faults and disturbances are shown in Table II. All units are in SI. Matlab's `particleswarm` is employed and the optimization parameters used are shown in Table III.

The results are shown in Tables IV and V. Each column refers to one of the four specified faults. Each fault is examined in absence of the rest. It is worth noting that only f_β excites the residual sufficiently to become detectable. f_{V_a} can, at-best, barely overcome the worst-case disturbances. f_ϕ and f_θ practically do not affect the residual, despite contributing to it.

This example system was purposely selected, as it corresponds to a real-world event: During a test-flight for data collection with the flying testbed of our laboratory, the Angle of Sideslip sensor was damaged and came loose during flight

TABLE III

PARTICLE SWARM OPTIMIZATION PROPERTIES: AOS SENSOR FAULT

Metric/Parameter	S_f^+	S_f^-	S_d^+
Size of Algebraic Equation System		14	
Total Number of Variables		39	
Swarm size	500	30	500
Maximum Iterations		200	
Maximum Stall Iterations		5	
Optimization Dimension	21	21	20

TABLE IV

RESPONSE METRICS

	f_β	f_{V_a}	f_ϕ	f_θ
S_f^-	249.71	5.23	0.75	0.14
S_f^+	368.77	64.35	32.66	29.35
S_d^+		29.18		

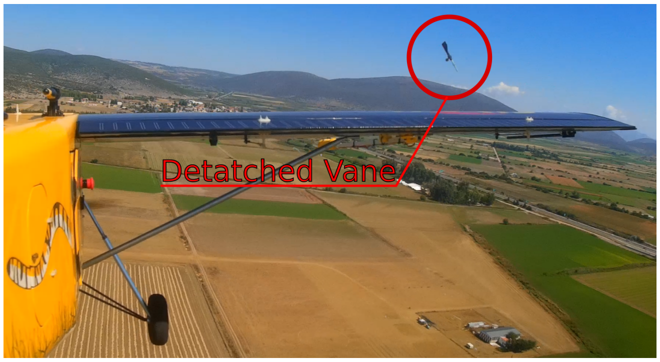

Fig. 1. Failure on Angle-of-Attack Sensor

(Fig. 1). The complete dataset collected from this flight can be found at [30].

Under the assumption that only f_β manifests, the corresponding analytical residual fault response expression is:

$$r = \dot{\tilde{V}}_a(\sin\tilde{\beta} - \sin(\tilde{\beta} + f_\beta)) \\ +V_a(\cos\tilde{\beta}\dot{\tilde{\beta}} - \cos(\tilde{\beta} + f_\beta)(\dot{\tilde{\beta}} + \dot{f}_\beta) \qquad (25)$$

Thus, the fault response is affected both by f_β and its derivative $\dot{f}_\beta$, which is to be expected, since this is a residual obtained through differentiation.

The residual value, calculated from the system measurements and inputs is plotted in blue in Figure 2 (middle). For reference, a fault estimate and its derivative are displayed, calculated using additional analytical redundancy stemming from the v estimate provided by the Inertial Navigation System (INS) of the UAV. Based on this estimate, the expected value of the residual, as a function of the fault, is plotted in red. The S_β^-, S_β^+ and S_d^+ values are also marked.

The residual indeed exhibits the expected level of response while f_β and $\dot{f}_\beta$ are large. When the value of $\dot{f}_\beta$ drops, the residual also diminishes, which is to be expected. The calculated residual peaks higher than S_β^+ only sparsely and by a small margin; this may be attributed to bad estimates of disturbance limits or unmodeled dynamics.

The robustness threshold is very low, barely visible in Fig. 2. However, Fig. 3, which depicts a previous, fault-free portion of the flight gives a much better picture. Over 500 s, and across a large variation of state (bottom subfigure) the residual never exceeds the prescribed robustness threshold, with very few exceptions. In fact, analysis shows an intermittent increase in f_β during these occurrences.

Evidently, the preliminary, model-based results of fault response and robustness fit the real flight data very tightly.

TABLE V

PSO Solution Time (seconds)

	f_β	f_{V_a}	f_ϕ	f_θ
S_f^-	176.59	126.37	119.15	104.36
S_f^+	56.13	40.87	22.00	21.60
S_d^+	22.92			

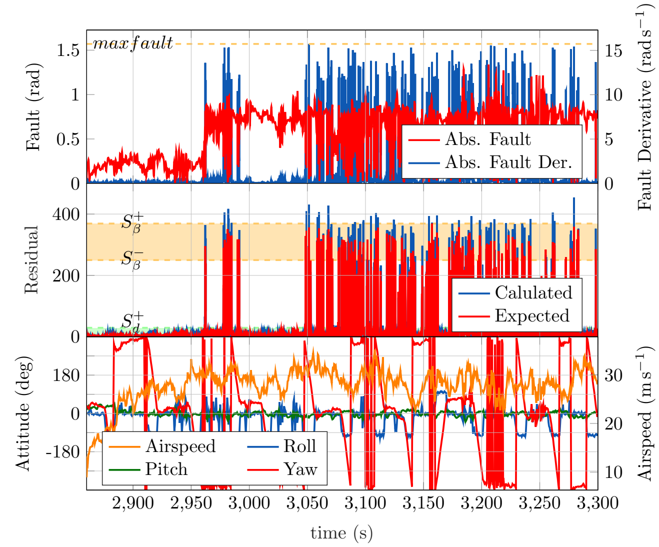

Fig. 2. AoS Fault, Residual Response and UAV State. The residual is much more sensitive to the derivative of the fault.

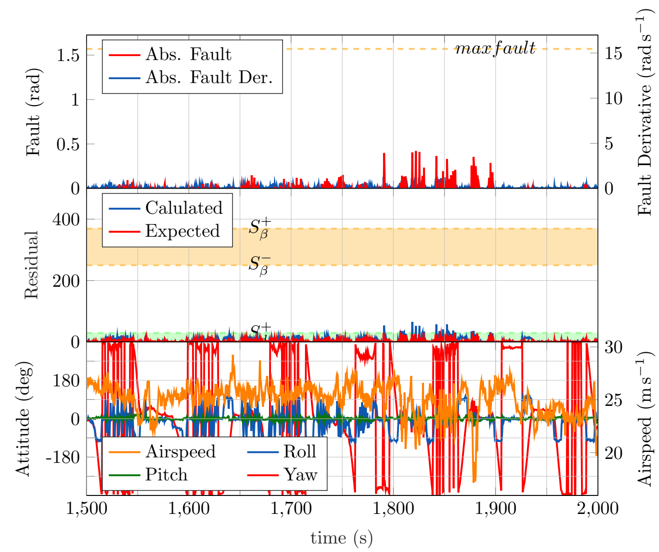

Fig. 3. Fault-Free Residual and Trajectories

VI. Conclusions and Future Work

In this work we have examined the fault sensitivity and robustness of non-linear residuals, in the context UAV FDI. Employing Particle Swarm Optimization, we have proposed a model-based, numerical method, suitable for both algebraic and dynamic subsystems of resigual generators.

An extended application of the methodology was performed on a residual sensitive to Angle-of-Sideslip sensor faults. Residual response levels were estimated a-priori and were then compared to data from a real faulty scenario. The results were matching closely.

In the future, we shall direct our efforts to test the presented method in experiments with dynamic residual generators. Also we intend to expand the metric formulations, leading to adaptive thresholds and increased autonomy in UAV systems.

References

[1] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*, 3rd ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016.

[2] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 293–311, Mar 2003.

[3] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual Reviews in Control*, vol. 32, no. 2, pp. 229–252, Dec 2008.

[4] S. Ding, *Model-Based Fault Diagnosis Techniques*. Springer, 2013.

[5] D. Henry, A. Zolghadri, F. Castang, and M. Monsion, "A new multi-objective filter design for guaranteed robust FDI performance," in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228)*, vol. 1. IEEE, 2001, pp. 173–178. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=980093

[6] I. Izadi, Q. Zhao, and T. Chen, "An optimal scheme for fast rate fault detection based on multirate sampled data," *Journal of Process Control*, vol. 15, no. 3, pp. 307–319, 2005.

[7] P. Castaldi, W. Geri, M. Bonfè, S. Simani, and M. Benini, "Design of residual generators and adaptive filters for the FDI of aircraft model sensors," *Control Engineering Practice*, vol. 18, no. 5, pp. 449–459, May 2010.

[8] J. Marzat, H. Piet-Lahanier, F. Damongeot, and E. Walter, "Model-based fault diagnosis for aerospace systems: a survey," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 226, no. 10, pp. 1329–1360, Oct 2012.

[9] D. Eriksson, M. Krysander, and E. Frisk, "Using quantitative diagnosability analysis for optimal sensor placement," *IFAC Proceedings Volumes*, vol. 45, no. 20, pp. 940–945, jan 2012. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84867074985{&}partnerID=tZOtx3y1http://linkinghub.elsevier.com/retrieve/pii/S1474667016348753

[10] H. Khorasgani, D. E. Jung, G. Biswas, E. Frisk, and M. Krysander, "Robust residual selection for fault detection," in *53rd IEEE Conference on Decision and Control*. IEEE, Dec 2014, pp. 5764–5769.

[11] E. Frisk, "Residual Generator Design for Non-Linear, Polynomial Systems - A Gröbner Basis Approach," *IFAC Proceedings Volumes*, vol. 33, no. 11, pp. 957–962, Jun 2000.

[12] A. Freddi, S. Longhi, and A. Monteriù, "A Diagnostic Thau Observer for a Class of Unmanned Vehicles," *Journal of Intelligent & Robotic Systems*, vol. 67, no. 1, pp. 61–73, jul 2012. [Online]. Available: http://link.springer.com/10.1007/s10846-012-9650-4

[13] V. Reppa, S. Timotheou, M. M. Polycarpou, and C. G. Panayiotou, "Optimization of observer design for sensor fault detection of nonlinear systems," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, no. Cdc. IEEE, dec 2017, pp. 5155–5160. [Online]. Available: http://ieeexplore.ieee.org/document/8264423/

[14] R. C. Avram, X. Zhang, and J. Muse, "Quadrotor Actuator Fault Diagnosis and Accommodation Using Nonlinear Adaptive Estimators," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 6, pp. 2219–2226, nov 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7831361/

[15] R. J. Patton and J. Chen, "Review of parity space approaches to fault diagnosis for aerospace systems," *Journal of Guidance, Control, and Dynamics*, vol. 17, no. 2, pp. 278–285, Mar 1994.

[16] B. Buchberger, "Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory," in *Multidimensional Systems Theory*. Dordrecht: Springer Netherlands, 1985, pp. 184–232. [Online]. Available: http://www.springerlink.com/index/10.1007/978-94-009-5225-6{_}6

[17] M. Fravolini, G. Campa, and M. Napolitano, "Design of Redundancy Relations for Unmanned Aerial Vehicle FDI," *AIAA Guidance, Navigation and Control Conference and Exhibit*, no. August, pp. 1–12, 2008. [Online]. Available: http://arc.aiaa.org/doi/abs/10.2514/6.2008-7421http://arc.aiaa.org/doi/pdf/10.2514/6.2008-7421

[18] R. Izadi-Zamanabadi, "Structural analysis approach to fault diagnosis with application to fixed-wing aircraft motion," in *Proceedings of the 2002 American Control Conference*, vol. 5. IEEE, 2002, pp. 3949–3954 vol.5.

[19] M. Staroswiecki, J. Cassar, and V. Cocquempot, "Generation of Optimal Structured Residuals in the Parity Space," *IFAC Proceedings Volumes*, vol. 26, no. 2, pp. 535–542, jul 1993. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1474667017483248

[20] V. Puig, F. Schmid, J. Quevedo, and B. Pulido, "A new fault diagnosis algorithm that improves the integration of fault detection and isolation," *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, CDC-ECC '05*, vol. 2005, pp. 3809–3814, 2005.

[21] D. Jung and C. Sundstrom, "A Combined Data-Driven and Model-Based Residual Selection Algorithm for Fault Detection and Isolation," *IEEE Transactions on Control Systems Technology*, pp. 1–15, 2017.

[22] F. Amato, C. Cosentino, M. Mattei, and G. Paviglianiti, "A direct/functional redundancy scheme for fault detection and isolation on an aircraft," *Aerospace Science and Technology*, vol. 10, no. 4, pp. 338–345, may 2006.

[23] G. Zogopoulos-Papaliakos and K. J. Kyriakopoulos, "Generating semi-explicit DAEs with Structural Index 1 for fault diagnosis using structural analysis," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2017, pp. 5812–5817.

[24] J. Cassar and M. Staroswiecki, "A structural approach for the design of failure detection and identification systems," in *IFAC Control of Industrial Systems*, 1997.

[25] M. Krysander and J. Aslund, "An Efficient Algorithm for Finding Over-constrained Sub-systems for Construction of Diagnostic Tests," in *16th International Workshop on Principles of Diagnosis (DX-05)*, vol. 3, no. 2, 2005, pp. 1–17.

[26] J. Unger, A. Kröner, and W. Marquardt, "Structural analysis of differential-algebraic equation systems—theory and applications," *Computers & Chemical Engineering*, vol. 19, no. 8, pp. 867–882, aug 1995.

[27] K. E. Brenan, S. L. V. Campbell, and L. R. Petzold, "Numerical solution of initial-value problems in differential-algebraic equations," p. 256, 1996.

[28] M. Clerc, *Particle Swarm Optimization*. ISTE, 2006.

[29] K. E. Parsopoulos and M. N. Vrahatis, "Particle Swarm Optimization Method for Constrained Optimization Problems," in *Intelligent Technologies-Theory and Applications: New Trends in Intelligent Technologies*, 2002.

[30] G. Zogopoulos-Papaliakos, "Unatrap flight 2017-09-06," Mendeley Data, Sep 2017.