

Algebraic Fault Detection and Identification for Rigid Robots

Alexander Lomakin and Joachim Deutscher

Abstract—This paper presents a method for algebraic fault detection and identification of nonlinear mechanical systems, describing rigid robots, by using an approximation with orthonormal Jacobi polynomials. An explicit expression is derived for the fault from the equation of motion, which is decoupled from disturbances and only depends on measurable signals and their time derivatives. Fault detection and identification is then achieved by polynomial approximation of the determined fault term. The results are illustrated by a simulation for a faulty SCARA.

I. INTRODUCTION

Due to increasing demands on mechanical systems, e.g. robots, in terms of availability and safety and the simultaneously increasing complexity, there is a continuous need of detecting and identifying potentially occurring faults in real time such that the general safety of the entire system can be guaranteed. This is why fault detection and identification methods are also becoming more and more important in industrial applications. Based on model knowledge of the system and a FMEA (Failure Mode and Effects Analysis), faults as well as other disturbances can be considered as unknown inputs in the motion equations. This fact is often used for observer-based fault detection and identification (see e.g. [1]) which is well established especially for linear systems (see [2]). Problems with these methods, however, lie in the selection of residuals and in the high implementation and parameterization effort for the mostly nonlinear systems. Furthermore, the high effort required for disturbance decoupling (see e.g. [3]) and the nonlinear observer design render these approaches uninteresting for practical applications. To overcome this problem, alternative, model-free approaches exist, which are for example based on fuzzy logic (see e.g. [4]) or neural networks (see e.g. [5]). Another possible solution to integrate model knowledge into the fault detection and at the same time reduce the design effort is the use of algebraic fault detection methods (see e.g. [6] and [7]). By reconstructing the time derivatives of the input and output variables with the help of derivative estimators the fault can be determined algebraically from the equation of motion.

In this paper an algebraic method for fault detection and identification is introduced. It is based on an algebraic polynomial approximation of the fault by Jacobi polynomials (see [8]). The properties of the approximation are introduced and then used to successively eliminate the time derivatives such that the fault reconstruction can be determined solely from the existing measurement signals, independent of the

characteristics of the input and output signals and the disturbance.

In the next section a formulation of the considered fault detection problem is given. Then, the polynomial approximation and its properties are presented in Section 3. In Sections 4 and 5 these results are used for a general fault detection and fault identification. The proposed method is demonstrated by means of a simulation for a faulty SCARA (Selective Compliance Assembly Robot Arm).

II. PROBLEM FORMULATION

Consider a robot as a general nonlinear mechanical system, which has n fully actuated rigid joints. The motion of the robot can be described by the generalized coordinates $q \in \mathbb{R}^n$, such as link positions that are measured and whose associated time derivatives are $\dot{q}(t) \in \mathbb{R}^n$ and $\ddot{q}(t) \in \mathbb{R}^n$. Taking the fault f and the disturbance d into account, the dynamic behaviour of the robot can be described by

$$\begin{aligned} \mathcal{M}(q)\ddot{q} + \mathcal{C}(q, \dot{q})\dot{q} + \mathcal{G}(q) &= u + \mathcal{F}(q, \dot{q})f \\ &+ \mathcal{D}(q, \dot{q})d \quad (1a) \\ y &= q \quad (1b) \end{aligned}$$

with the initial condition $q(0), \dot{q}(0) \in \mathbb{R}^n$. In this formulation, $\mathcal{M}(q) \in \mathbb{R}^{n \times n}$ corresponds to the generalized inertia matrix and contains mass, inertia and geometric parameters of the joints. The vector $\mathcal{C}(q, \dot{q})\dot{q} \in \mathbb{R}^n$ represents the Coriolis and centripetal components and $\mathcal{G}(q) \in \mathbb{R}^n$ is regarded as the influence of gravitational force. The output $y \in \mathbb{R}^n$ of (1) is available for measurement and equal to the link positions q . The presented method implicitly reduces the influence of measurement noise (see, e.g. [9]). Hence, the influence of noise on the system is not discussed in this paper. The right-hand side of (1) contains all the non-conservative forces that affect the robot. These forces include the input torque $u \in \mathbb{R}^n$, as well as the unknown fault vector $f(t) = \text{col}(f_1(t), f_2(t), \dots, f_{n_f}(t)) \in \mathbb{R}^{n_f}$, containing the particular faults $f_i, i = 1, \dots, n_f$, and unknown disturbance $d = \text{col}(d_1, d_2, \dots, d_{n_d}) \in \mathbb{R}^{n_d}$. The corresponding matrices

$$\mathcal{F}(q, \dot{q}) = [\mathcal{F}_1(q, \dot{q}) \dots \mathcal{F}_{n_f}(q, \dot{q})] \in \mathbb{R}^{n \times n_f} \quad (2)$$

and

$$\mathcal{D}(q, \dot{q}) = [\mathcal{D}_1(q, \dot{q}) \dots \mathcal{D}_{n_d}(q, \dot{q})] \in \mathbb{R}^{n \times n_d} \quad (3)$$

represent the influence of the fault and the disturbance, respectively, and are assumed to be known, based on knowledge from the specific application and the FMEA. It is also assumed that $\mathcal{D}(q, \dot{q})$ has full column rank and therefore $\text{rank } \mathcal{D}(q, \dot{q}) = n_d$. Otherwise, disturbances with the same effect can be combined to form a new disturbance. In this

Alexander Lomakin and Joachim Deutscher are with the Lehrstuhl für Regelungstechnik, Friedrich-Alexander-Universität, Erlangen-Nürnberg, Germany {alexander.lomakin, joachim.deutscher}@fau.de

paper, faults are assumed to be additive only, e.g., actuator faults.

The fault detection and identification problem addressed in this paper is interpreted analogously to [10] as the subsequent design problem.

For a given system (1), find a *residual signal* r

$$r = \Phi(q, \dot{q}, \ddot{q}, u) \in \mathbb{R}^{n_r} \quad (4)$$

such that the conditions

$$\begin{aligned} \text{I. } & \lim_{t \rightarrow \infty} \|r\| = 0, & \forall f = 0 \\ \text{II. } & r \neq 0, & \forall f \neq 0 \end{aligned}$$

are satisfied for any input $u \in \mathbb{R}^n$, any disturbance $d \in \mathbb{R}^{n_d}$ and any initial states $q(0), \dot{q}(0) \in \mathbb{R}^n$. Then the fault f can be detected by the residual signal r . If additionally to I and II any two particular faults f_i and f_j , with $f_i, f_j \in \mathbb{R}^{n_f}$ and $f_j \neq f_i$, the corresponding residual signals r_i and r_j can be distinguished for any finite time interval \mathcal{I}_t , the fault f can also be isolated by the residual signal r . Furthermore, if f_i and f_j can be additionally estimated by r for any finite time interval \mathcal{I}_t , the fault is assumed to be identifiable. Henceforth, the addressed problem will be regarded as the residual generation problem for fault detection (RGP-FD) and fault identification (RGP-FDI), respectively. It should be noted that a general definition of RGP-FD and RGP-FDI in literature are often referred to concepts of differential algebraic detectability and isolability (see e.g. [7]). In this paper these definitions are defined and interpreted specifically with respect to the equations of motion.

III. ALGEBRAIC FAULT DETECTION AND IDENTIFICATION

To solve the presented fault detection problem, it is necessary to calculate the residual signal r . Since the derivatives of q are not available for measurement, a computational method to determine the time derivatives \dot{q}, \ddot{q} and thus the residual, which only needs the available signals u and q , will be presented in this section.

A. Polynomial approximation

As already shown in [8] and [11], a polynomial approximation of a function can be determined to calculate the derivatives without differentiation. In this paper an analogous method is used by defining a polynomial approximation operator $\mathcal{P}\{\cdot\}$, which calculates the polynomial approximation of a function by an integral transformation. Therefore, the derivation and definition of $\mathcal{P}\{\cdot\}$ will be described first.

Consider the function $x \in L_2([t-T, t])$ defined on the sliding time window $\mathcal{I}_{t,T} = [t-T, t]$, $T > 0$. In order to only take a constant time independent approximation interval $\tilde{\mathcal{I}} = [-1, 1]$ into account, the bijective transformation $\phi_T : \tilde{\mathcal{I}} \mapsto \mathcal{I}_{t,T}$, which maps the interval $\tilde{\mathcal{I}} = [-1, 1]$ to the given time window $\mathcal{I}_{t,T}$ is introduced. This transformation reads

$$\phi_T(\tau') = t + \frac{\tau' - 1}{2}T, \quad \tau' \in [-1, 1], \quad (5)$$

with the inverse mapping $\phi_T^{-1} : \mathcal{I}_{t,T} \mapsto \tilde{\mathcal{I}}$ given by

$$\phi_T^{-1}(t') = 1 + 2\frac{t' - t}{T}, \quad t' \in [t-T, t]. \quad (6)$$

The transformed function $\bar{x} = x \circ \phi_T$ is then defined on a Hilbert space $\mathcal{H} = L_2([-1, 1])$ with the inner product

$$\langle \varphi_i, \varphi_j \rangle = \int_{-1}^1 \varphi_i(\tau) \varphi_j(\tau) w^{(\alpha, \beta)}(\tau) d\tau, \quad \forall \varphi_i, \varphi_j \in \mathcal{H}, \quad (7)$$

and the induced norm

$$\|\varphi\| = \sqrt{\langle \varphi, \varphi \rangle}, \quad \forall \varphi \in \mathcal{H}. \quad (8)$$

The weight function $w^{(\alpha, \beta)}$, which allows to consider Jacobi polynomials as an orthonormal basis for \mathcal{H} , is given by

$$w^{(\alpha, \beta)}(\tau) = \begin{cases} (1-\tau)^\alpha (1+\tau)^\beta, & \tau \in [-1, 1], \\ 0, & \tau \notin [-1, 1], \end{cases} \quad (9)$$

with the real exponential coefficients $\alpha, \beta > -1$ as a degree of freedom. They can be chosen to achieve special blocking properties of individual frequencies in the signal (see, e.g., [11]). Then, it is possible to introduce an orthonormal basis $\{P_i^{(\alpha, \beta)}\}_{i=0}^\infty$ for \mathcal{H} by the normalized Jacobi polynomials $P_i^{(\alpha, \beta)}$ (see [12, Sec. 4.3]). According to the projection theorem (see, e.g. [13]) the best fitting (in the least squares sense) approximation of N -th order $\hat{x} \in \mathcal{H}$ of \bar{x} always exists unambiguously, and can be calculated by

$$\hat{x}(\tau) = \sum_{i=0}^N \underbrace{\langle \bar{x}, P_i^{(\alpha, \beta)} \rangle}_{c_i} P_i^{(\alpha, \beta)}(\tau), \quad \tau \in [-1, 1], \quad (10)$$

while c_i is considered as the corresponding expansion coefficient. Furthermore, the approximation \hat{x} is orthogonal, i.e. $\langle \hat{x}, p \rangle = 0$, to any polynomial $p \in \bar{\pi}_N := \text{span}\{P_i^{(\alpha, \beta)}\}_{i=N+1}^\infty$ and is exactly equal to \bar{x} , if $\bar{x} \in \pi_N := \text{span}\{P_i^{(\alpha, \beta)}\}_{i=0}^N$.

By applying the transformation (5), (6) the approximation (10) is valid in the time window $\mathcal{I}_{t,T}$ and can be evaluated at any time $t' \in \mathcal{I}_{t,T}$. It is reasonable to choose the evaluation at the time t in order to approximate values at current time. However, by adding a delay $t_d \geq 0$ as a zero $p_{N+1}^{(\alpha, \beta)}$ of the Jacobi polynomial $P_{N+1}^{(\alpha, \beta)}$, the order of the approximation error $\tilde{x} = x - \hat{x}$ can be reduced by one (see [9]). For this reason it makes sense to introduce a delay t_d if the delay is justifiable with regard to fault detection. The delayed polynomial approximation of x based on (10) can thus be written as

$$\begin{aligned} \hat{x}(t-t_d) &= \sum_{i=0}^N \langle x \circ \phi_T, P_i^{(\alpha, \beta)} \rangle (P_i^{(\alpha, \beta)} \circ \phi_T^{-1}(t-t_d)) \\ &= \langle x \circ \phi_T, R_{N, t_d}^{(\alpha, \beta)} \rangle, \end{aligned} \quad (11)$$

with

$$R_{N, t_d}^{(\alpha, \beta)}(\tau) = \sum_{i=0}^N P_i^{(\alpha, \beta)}(\tau) (P_i^{(\alpha, \beta)} \circ \phi_T^{-1}(t-t_d)). \quad (12)$$

The definition of the inner product (7) can be used to represent $\hat{x}(t - t_d)$ by the integral

$$\hat{x}(t - t_d) = \int_{-1}^1 (x \circ \phi_T(\tau)) g_{N,t_d}(\tau) d\tau, \quad (13)$$

with the kernel

$$g_{N,t_d}(\tau) = R_{N,t_d}^{(\alpha,\beta)}(\tau) w^{(\alpha,\beta)}(\tau). \quad (14)$$

To evaluate the integral within the original time window $\mathcal{I}_{t,T}$, the substitution $\bar{\tau} = t - \phi_T(\tau)$ is performed. The approximation can therefore be written as

$$\hat{x}(t - t_d) = \int_0^T x(t - \bar{\tau}) g_{N,t_d}(\bar{\tau}) d\bar{\tau} =: \mathcal{P}_{N,t_d}\{x\}(t), \quad (15)$$

with the kernel

$$g_{N,t_d}(\bar{\tau}) = \frac{2}{T} (R_{N,t_d}^{(\alpha,\beta)} w^{(\alpha,\beta)}) \circ \phi_T^{-1}(t - \bar{\tau}), \quad (16)$$

which is independent of t , since $\phi_T^{-1}(t - \bar{\tau}) = 1 - \frac{2}{T}\bar{\tau}$. Based on these assumptions, for any $x \in L_2([t - T, t])$ the operator $\mathcal{P}_{N,t_d}\{x\}$ can now be defined as the time-delayed polynomial approximation based on (15).

Assume $x \in L_2([t - T, t]) \cap C^{k-1}([t - T, t])$ and the k -th derivative $x^{(k)}$ exists and is Lebesgue integrable. Since the kernel (16) and its $k - 1$ derivatives have a compact support due to (9) in $[0, T]$, if $\alpha, \beta \geq k$, the polynomial approximation $x^{(k)}$ can be calculated by successive application of integration by parts

$$\begin{aligned} \mathcal{P}_{N,t_d}\{x^{(k)}\}(t) &= \int_0^T x^{(k)}(t - \tau) g_{N,t_d}(\tau) d\tau \\ &= \int_0^T x(t - \tau) g_{N,t_d}^{(k)}(\tau) d\tau \\ &=: \mathcal{P}_{N,t_d}^{(k)}\{x\}(t) \end{aligned} \quad (17)$$

with the derivative of the kernel given by

$$g_{N,t_d}^{(k)}(\tau) = (-1)^k \frac{2}{T} (R_{N,t_d}^{(\alpha,\beta)} w^{(\alpha,\beta)})^{(k)} \circ \phi_T^{-1}(t - \tau). \quad (18)$$

The polynomial approximation of $x^{(k)}$ can thus be calculated by applying the differentiation approximation operator $\mathcal{P}_{N,t_d}^{(k)}\{\cdot\}$ to x .

According to the previously established definition of the polynomial approximation operator, specific properties can be deduced, which are listed below:

- **linearity:** For any two functions $x_1, x_2 \in L_2([t - T, t])$, and any constants $c_1, c_2 \in \mathbb{R}$, the equivalence

$$\begin{aligned} \mathcal{P}_{N,t_d}\{c_1 x_1 + c_2 x_2\}(t) \\ = c_1 \mathcal{P}_{N,t_d}\{x_1\}(t) + c_2 \mathcal{P}_{N,t_d}\{x_2\}(t), \end{aligned} \quad (19)$$

is valid.

- **composition:** For a function $x \in L_2([t - T, t])$ and a second Lipschitz continuous function $\psi : \mathbb{R} \mapsto \mathbb{R}$ the commutation of the composition of ψ and the polynomial Approximation $\mathcal{P}_{N,t_d}\{x\}(t)$ are approximately equal:

$$\mathcal{P}_{N,t_d}\{\psi(x)\}(t) \approx \psi(\mathcal{P}_{N,t_d}\{x\}(t)) = \psi(\hat{x}), \quad (20)$$

if x is at least $C^0([t - T, t])$ and the Lipschitz constant L of ψ is sufficiently small. The resulting approximation error is then bounded by

$$\|\psi(x) - \psi(\hat{x})\| \leq L \|x - \hat{x}\|. \quad (21)$$

- **reproduction:** For any function $x \in L_2([t - T, t])$, with $x \in \pi_N$, the polynomial approximation is equal to the approximated function:

$$\begin{aligned} \mathcal{P}_{N,t_d}\{x\}(t) &= x(t - t_d) \\ &= \sum_{i=0}^N c_i P_i^{(\alpha,\beta)} \circ \phi_T^{-1}(t - t_d). \end{aligned} \quad (22)$$

The coefficients c_i can be determined by the integral

$$c_i = \int_0^T x(t - \tau) g_{c_i}(\tau) d\tau =: \mathcal{P}_{N,0,c_i}\{x\}(t), \quad (23)$$

with the kernel

$$g_{c_i}(\tau) = \frac{2}{T} (P_i^{(\alpha,\beta)} w^{(\alpha,\beta)}) \circ \phi_T^{-1}(t - \tau), \quad (24)$$

which is independent of t , since $\phi_T^{-1}(t - \bar{\tau}) = 1 - \frac{2}{T}\bar{\tau}$. Thus, the representation of the function x as a linear combination of Jacobi polynomials to the N -th order is equivalent up to a delay t_d to x if $x \in \pi_N$. The coefficients c_i can be calculated directly from x by (23).

- **differentiation:** For each k -times differentiable function $x \in L_2([t - T, t])$, assuming that the corresponding time derivative of k -th degree $x^{(k)} \in L_2([t - T, t])$ exists and is Lebesgue integrable, the polynomial approximation of $x^{(k)}$ is equivalent to the differentiation approximation of x :

$$\mathcal{P}_{N,t_d}\{x^{(k)}\}(t) = \mathcal{P}_{N,t_d}^{(k)}\{x\}(t). \quad (25)$$

- **partial approximation:** For any two functions $x_1, x_2 \in L_2([t - T, t])$ and $x_1 \in \pi_{N^*}$ with $N^* \in \mathbb{N}$, the polynomial approximation of the product $x_1 x_2$ can be defined by

$$\begin{aligned} \mathcal{P}_{N,t_d}\{x_1 x_2\}(t) &= \mathcal{P}_{N,t_d}\{\mathcal{P}_{N^*,0}\{x_1\} x_2\}(t) \\ &= \mathcal{P}_{N,t_d}\left\{\sum_{i=0}^{N^*} c_i (P_i^{(\alpha,\beta)} \circ \phi_T^{-1}) x_2\right\}(t) \\ &= \sum_{i=0}^{N^*} c_i \mathcal{P}_{N,t_d}\{(P_i^{(\alpha,\beta)} \circ \phi_T^{-1}) x_2\}(t). \end{aligned} \quad (26a)$$

The components of the polynomial approximation of the product of $x \in L_2([t - T, t])$ and the Jacobi polynomial $P_i^{(\alpha,\beta)} \circ \phi_T^{-1} \in L_2([t - T, t])$ can be determined by

$$\begin{aligned} \mathcal{P}_{N,t_d}\{(P_i^{(\alpha,\beta)} \circ \phi_T^{-1}) x\}(t) \\ = \int_0^T x(t - \tau) \tilde{g}_{N,t_d,i}(\tau) d\tau =: \tilde{\mathcal{P}}_{N,t_d,i}\{x\}(t), \end{aligned} \quad (26b)$$

with the kernel

$$\tilde{g}_{N,t_d,i}(\tau) = \frac{2}{T} (P_i^{(\alpha,\beta)} R_{N,t_d}^{(\alpha,\beta)} w^{(\alpha,\beta)}) \circ \phi_T^{-1}(t - \tau). \quad (27)$$

Thus, a polynomial approximation of the product $x_1 x_2$ can be obtained by the sum of the products of the coefficients c_i of the approximation $\mathcal{P}_{N^*,0}\{x_1\}$ and the modified polynomial approximation $\tilde{\mathcal{P}}_{N,t_d,i}\{x_2\}$. It should also be noted that the order N^* of the partial approximation of x_1 does not have to be related to the order N of the approximation of the product. It only needs to be large enough to map x_1 to the $\mathcal{I}_{t,T}$ interval. Furthermore it is also possible to transfer the differentiation of x_1 or x_2 to the kernel $\tilde{g}_{N,t_d,i}$ or g_{c_i} , respectively, analogous to the differentiation (17) because the property of the compact support has not changed by the multiplication with the Jacobi polynomial of i -th degree.

Based on the presented properties, the polynomial approximation of a complex function with unavailable time derivatives can also be obtained and successively calculated solely by the available signals and the application of the presented integral transformations.

B. Real-time Implementation

The polynomial approximation can be calculated from the history of the signal x in the interval $\mathcal{I}_{t,T}$. Since the continuous time signal x has to be represented for the real-time implementation as a sequence of samples $x[k] = x(kT_s)$ of the length $L = \frac{T}{T_s} \in \mathbb{N}$ with the sampling time $T_s \in \mathbb{R}$, a discrete approximation of the integral must be realized. Therefore, the midpoint rule is used, under the assumption that the measured signal can be considered constant within one sampling period to convert the integral to the weighted sum

$$\begin{aligned} \mathcal{P}_{N,t_d}\{x\}[k] &= \sum_{j=0}^{L-1} \underbrace{\int_{jT_s}^{(j+1)T_s} g_{N,t_d}(\tau) d\tau}_{w[j]} x[k-j] \\ &= \sum_{j=0}^{L-1} w[j] x[k-j], \end{aligned} \quad (28)$$

with the weights $w[j] \in \mathbb{R}$, $j \in \{1, \dots, L-1\}$. It should also be mentioned that this approximation method achieves compensation by half a sampling period. Since the weights $w[j]$ are constant and therefore independent of the time and the measured signals, they can be determined in advance and do not have to be calculated during evaluation. An analogous procedure to (28) can also be defined for the differentiation approximation operator (17), to determine the coefficients (23) and the modified approximation operator (26b), since in all cases it is an integral transformation with a time independent kernel. For this reason, the polynomial approximation can be executed directly on a controller in real-time, by evaluating the individual weighted sums at runtime.

C. Fault detection

For fault detection, a residual signal r must be calculated, which fulfils the conditions I and II. To obtain such a residual signal, one possible approach is to rearrange the equation

(1a) to isolate the known signals q and u from the fault f . For notational convenience, the dependencies of q and \dot{q} in the matrices $\mathcal{M}(q)$, $\mathcal{C}(q, \dot{q})$, $\mathcal{G}(q)$, $\mathcal{F}(q, \dot{q})$ and $\mathcal{D}(q, \dot{q})$ are not displayed. The generated residual r_d , which still depends on d , is thus introduced explicitly by

$$r_d = \mathcal{M}\ddot{q} + \mathcal{C}\dot{q} + \mathcal{G} - u = \Phi(q, \dot{q}, \ddot{q}, u) = \mathcal{F}f + \mathcal{D}d \quad (29)$$

with $n_r = n$ and meets the condition I of the RGP-FD if d is neglected. The second condition is satisfied if $\mathcal{F}(q, \dot{q})$ is non-zero for all q and \dot{q} . If disturbances affect the system, it is necessary to decouple the residual from the unknown disturbance d . In order to eliminate this dependency, determine the left annihilator $\mathcal{D}^\perp \in \mathbb{R}^{n \times n}$ of \mathcal{D} , i.e., $\mathcal{D}^\perp \mathcal{D} = 0$, by

$$\mathcal{D}^\perp = I - \mathcal{D}\mathcal{D}^\dagger, \quad (30)$$

with the Moore-Penrose generalized inverse $\mathcal{D}^\dagger \in \mathbb{R}^{n_d \times n}$ given by

$$\mathcal{D}^\dagger = (\mathcal{D}^\top \mathcal{D})^{-1} \mathcal{D}^\top, \quad (31)$$

since \mathcal{D} has full column rank (see, e.g., [14]). By premultiplication of the residual with the annihilator $\mathcal{D}^\perp \in \mathbb{R}^{n \times n}$, the influence of the disturbance on r_d is eliminated yielding the desired residual

$$r = \mathcal{D}^\perp (\mathcal{M}\ddot{q} + \mathcal{C}\dot{q} + \mathcal{G} - u) = \mathcal{D}^\perp \mathcal{F}f. \quad (32)$$

It is unequal to the zero vector if any fault f is present, as long as $\mathcal{D}^\perp \mathcal{F}_i \neq 0, \forall i \in \{1, \dots, n_f\}, \forall q, \dot{q} \in \mathbb{R}^n$. Then r solves the RGP-FD and can be used for fault detection. The only remaining problem is that the determined residual also depends on time derivatives of q and therefore cannot be evaluated. Now the estimates of \dot{q} and \ddot{q} could simply be inserted into the equation (32) to evaluate the residual r . However, since \ddot{q} does not have to be continuous, it can happen that the estimation error caused by the polynomial approximation of \ddot{q} and the further multiplication with other terms does not consistently reproduce the residual and thus excites it. For this reason, a polynomial approximation of the whole residual r is performed, by using the polynomial approximation operator $\mathcal{P}_{N,t_d}\{\cdot\}$ instead of replacing \dot{q} and \ddot{q} by the derivative estimates $\hat{\dot{q}}$ and $\hat{\ddot{q}}$ within the residual. The polynomial approximation does not affect the properties of the residual w.r.t. the decoupling of the input u and the disturbance d and can be determined sufficiently well according to the choice of the polynomial degree N . By applying the operator to (32) and using its linearity, the polynomial approximation of r can be determined by

$$\begin{aligned} \hat{r} = \mathcal{P}_{N,t_d}\{r\} &= \mathcal{P}_{N,t_d}\{\mathcal{D}^\perp (\mathcal{M}\ddot{q} + \mathcal{C}\dot{q} + \mathcal{G} - u)\} \\ &= \mathcal{P}_{N,t_d}\{\mathcal{D}^\perp \mathcal{M}\ddot{q}\} + \mathcal{P}_{N,t_d}\{\mathcal{D}^\perp \mathcal{C}\dot{q}\} \\ &\quad + \mathcal{P}_{N,t_d}\{\mathcal{D}^\perp \mathcal{G}\} - \mathcal{P}_{N,t_d}\{\mathcal{D}^\perp u\} \\ &= \mathcal{P}_{N,t_d}\{\mathcal{D}^\perp \mathcal{F}f\}. \end{aligned} \quad (33)$$

Every single term can therefore be determined individually by applying the partial approximation (26) and the differentiation approximation operator (25). Then, the unknown

derivatives of q can be successively eliminated from (33) and the approximation of the residual is given by

$$\begin{aligned} \hat{r} = & \sum_{j=0}^{N^*} \mathcal{P}_{N,0,c_j} \{ \mathcal{D}^\perp(q, \hat{q}) \mathcal{M}(q) \} \tilde{\mathcal{P}}_{N,t_d,j}^{(2)} \{ q \} \\ & + \mathcal{D}^\perp(\hat{q}, \hat{q}) \mathcal{C}(\hat{q}, \hat{q}) \hat{q} + \mathcal{P}_{N,t_d} \{ \mathcal{D}^\perp(q, \hat{q}) \mathcal{G}(q) \} \\ & - \sum_{j=0}^{N^*} \mathcal{P}_{N,0,c_j} \{ \mathcal{D}^\perp(q, \hat{q}) \} \tilde{\mathcal{P}}_{N,t_d,j} \{ u \} \end{aligned} \quad (34)$$

with $\hat{q} = \mathcal{P}_{N,t_d} \{ q \}$, $\hat{q} = \mathcal{P}_{N,t_d}^{(1)} \{ q \}$. Considering that q and \hat{q} are assumed to be at least $C^0([t-T, t])$ and that \mathcal{C} , \mathcal{D} , \mathcal{F} and \mathcal{G} are Lipschitz, the composition rule (20) can be applied to the Coriolis and gravitational terms. Since (34) meets criteria I and II of RGP-FD, it can be used for fault detection.

D. Fault identification and estimation

As shown before, it is possible to use the residual to determine the occurrence of a fault. However, in many cases it is still necessary to know which fault has occurred and to estimate it accordingly. This can be achieved by solving (32) for f , which is possible for $\text{rank}(\mathcal{D}^\perp \mathcal{F}) = n_f$ and leads to the fault identification

$$f = (\mathcal{D}^\perp \mathcal{F})^\dagger \mathcal{D}^\perp (\mathcal{M} \ddot{q} + \mathcal{C} \dot{q} + \mathcal{G} - u). \quad (35)$$

Therein, $(\mathcal{D}^\perp \mathcal{F})^\dagger$ is the Moore-Penrose generalized inverse of $\mathcal{D}^\perp \mathcal{F}$. If $\text{rank}(\mathcal{D}^\perp \mathcal{F}) = \tilde{n}_f < n_f$ fault identification can still be partially realized. In this case the solution of (32) is given by

$$f = (\mathcal{D}^\perp \mathcal{F})^\dagger \mathcal{D}^\perp (\mathcal{M} \ddot{q} + \mathcal{C} \dot{q} + \mathcal{G} - u) + (\mathcal{D}^\perp \mathcal{F})_R^\perp l, \quad (36)$$

in which $(\mathcal{D}^\perp \mathcal{F})_R^\perp \in \mathbb{R}^{n_f \times (n_f - \tilde{n}_f)}$ is the full column rank right annihilator of $\mathcal{D}^\perp \mathcal{F}$, i.e., $\mathcal{D}^\perp \mathcal{F} (\mathcal{D}^\perp \mathcal{F})_R^\perp = 0$ and any vector $l \in \mathbb{R}^{n_f - \tilde{n}_f}$. Hence, f cannot be determined uniquely anymore. However, any linear combination $\tilde{f} = T f$ of f with $T \in \mathbb{R}^{\tilde{n}_f \times n_f}$ and $T (\mathcal{D}^\perp \mathcal{F})_R^\perp = 0$ can still be computed unambiguously, which follows from premultiplication of (36) by T . Therein, T is parametrizable by $T = N (\mathcal{D}^\perp \mathcal{F})$ with an arbitrary matrix $N \in \mathbb{R}^{\tilde{n}_f \times n}$ satisfying $\text{rank} N (\mathcal{D}^\perp \mathcal{F}) = \tilde{n}_f$ to obtain linear independent faults to be identified. The degrees of freedom in the latter matrix can be utilized to select faults of interest. A solution for N that fulfils $\text{rank} N (\mathcal{D}^\perp \mathcal{F}) = \tilde{n}_f$ obviously always exists, since $\text{rank}(\mathcal{D}^\perp \mathcal{F}) = \tilde{n}_f$. Subsequently, however, the case that $\text{rank}(\mathcal{D}^\perp \mathcal{F}) = n_f, \forall q, \dot{q} \in \mathbb{R}^n$ is always considered for fault diagnosis. Analogous to fault detection, by the polynomial approximation the fault estimate can be defined and represented by

$$\begin{aligned} \hat{f} = & \mathcal{P}_{N,t_d} \{ \mathcal{K} (\mathcal{M} \ddot{q} + \mathcal{C} \dot{q} + \mathcal{G} - u) \} \\ = & \mathcal{P}_{N,t_d} \{ \mathcal{K} \mathcal{M} \ddot{q} \} + \mathcal{P}_{N,t_d} \{ \mathcal{K} \mathcal{C} \dot{q} \} \\ & + \mathcal{P}_{N,t_d} \{ \mathcal{K} \mathcal{G} \} - \mathcal{P}_{N,t_d} \{ \mathcal{K} u \}, \end{aligned} \quad (37)$$

with $\mathcal{K} = (\mathcal{D}^\perp \mathcal{F})^\dagger \mathcal{D}^\perp$ and the fault estimate $\hat{f} \in \mathbb{R}^{n_f}$. The RGP-FDI can therefore be solved by (37) if f can be locally approximated within the interval $\mathcal{I}_{t,T}$ by a polynomial of N -th degree.

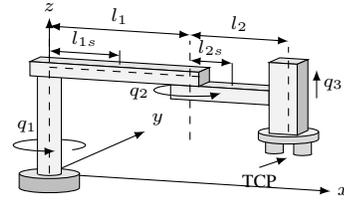


Fig. 1. Schematic illustration of a SCARA.

IV. EXAMPLE

To illustrate the results of this paper by a simulation, consider a SCARA as an example of a nonlinear mechanical system. This system has three degrees of freedom and consists of three movable segments with the masses m_1 , m_2 , m_3 and the corresponding inertia values J_1 , J_2 and J_3 (see Fig. 1). The lengths l_{1s} and l_{2s} describe the distance between the center of mass of the first and the second segment from the corresponding rotation axis. For (1) the matrices and vectors $\mathcal{M}(q)$, $\mathcal{C}(q, \dot{q})$ and $\mathcal{G}(q)$ are

$$\mathcal{M}(q) = \begin{bmatrix} \theta_1 + 2\theta_2 \cos(q_2) & \theta_3 + \theta_2 \cos(q_2) & 0 \\ \theta_3 + \theta_2 \cos(q_2) & \theta_3 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \quad (38a)$$

$$\mathcal{C}(q, \dot{q}) = \begin{bmatrix} -\dot{q}_2 \theta_2 \sin(q_2) & -(\dot{q}_1 + \dot{q}_2) \theta_2 \sin(q_2) & 0 \\ \theta_2 \dot{q}_1 \sin(q_2) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (38b)$$

$$\mathcal{G}(q) = [0 \quad 0 \quad m_3 g]^\top, \quad (38c)$$

with the constant parameters $\theta_1 = J_1 + J_2 + J_3 + l_{1s}^2 m_2 + l_1^2 m_3 + l_2^2 m_3 + l_{2s}^2 m_1 + l_{2s}^2 m_2$, $\theta_2 = l_1 l_2 m_3 + l_1 l_{2s} m_2$ and $\theta_3 = m_3 l_2^2 + m_2 l_{2s}^2 + J_2 + J_3$.

In addition, actuator faults f_1 and f_2 are considered for the first two joints, and a force acting in y - z -direction at the TCP (Tool Center Point) with an unknown value is regarded as the disturbance d . The matrices $\mathcal{F}(q, \dot{q})$ and $\mathcal{D}(q, \dot{q})$ are thus given by

$$\mathcal{F}(q, \dot{q}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^\top \quad (39a)$$

$$\mathcal{D}(q, \dot{q}) = \begin{bmatrix} l_2 \cos(q_1 + q_2) + l_1 \cos(q_1) \\ l_2 \cos(q_1 + q_2) \\ 1 \end{bmatrix}, \quad (39b)$$

with $\text{rank} \mathcal{D} = 1 = n_d$. The entire dynamic of the SCARA can thus be described as a nonlinear equation of motion. Since $\text{rank} \mathcal{D}^\perp \mathcal{F} = 2 = n_f$ the fault can be determined according to (35) by

$$\begin{aligned} f_1 = & \theta_1 \ddot{q}_1 + 2\theta_2 \cos(q_2) \ddot{q}_1 + \theta_3 \ddot{q}_2 - g l_2 m_3 \cos(q_1 + q_2) \\ & + l_1 \cos(q_1) u_3 - l_2 m_3 \cos(q_1 + q_2) \ddot{q}_3 + \theta_2 \cos(q_2) \ddot{q}_2 \\ & - l_1 m_3 \cos(q_1) \ddot{q}_3 - g l_1 m_3 \cos(q_1) - \dot{q}_2^2 \theta_2 \sin(q_1) \\ & - 2\dot{q}_1 \dot{q}_2 \theta_2 \sin(q_1) + l_2 \cos(q_1 + q_2) u_3 - u_1 \end{aligned} \quad (40a)$$

$$\begin{aligned} f_2 = & \theta_3 \ddot{q}_1 + \theta_2 \cos(q_2) \ddot{q}_1 + \theta_3 \ddot{q}_2 - u_2 + \dot{q}_1^2 \theta_2 \sin(q_1) \\ & + l_2 \cos(q_1 + q_2) u_3 - g l_2 m_3 \cos(q_1 + q_2) \\ & - m_3 l_2 \cos(q_1 + q_2) \ddot{q}_3. \end{aligned} \quad (40b)$$

In order to obtain a fault estimate \hat{f} , it is necessary to apply the polynomial approximation to (40) and then use the operator's introduced properties (19) to (26) in such a way that the dependencies on time derivatives of the input or output signals are eliminated. The estimated faults \hat{f}_1 and \hat{f}_2 then become

$$\begin{aligned} \hat{f}_1 = & \theta_1 \hat{q}_1 + 2\theta_2 \mathcal{P}_{N,t_d} \{ \cos(q_2) \ddot{q}_1 \} + \theta_3 \hat{q}_2 - \hat{q}_2^2 \theta_2 \sin(\hat{q}_1) \\ & - gl_2 m_3 \cos(\hat{q}_1 + \hat{q}_2) + l_1 \mathcal{P}_{N,t_d} \{ \cos(q_1) u_3 \} \\ & - l_2 m_3 \mathcal{P}_{N,t_d} \{ \cos(q_1 + q_2) \ddot{q}_3 \} + \theta_2 \mathcal{P}_{N,t_d} \{ \cos(q_2) \ddot{q}_2 \} \\ & - l_1 m_3 \mathcal{P}_{N,t_d} \{ \cos(q_1) \ddot{q}_3 \} - gl_1 m_3 \cos(\hat{q}_1) - \hat{u}_1 \\ & - 2\hat{q}_1 \hat{q}_2 \theta_2 \sin(\hat{q}_1) + l_2 \mathcal{P}_{N,t_d} \{ \cos(q_1 + q_2) u_3 \} \quad (41a) \end{aligned}$$

$$\begin{aligned} \hat{f}_2 = & \theta_2 \mathcal{P}_{N,t_d} \{ \cos(q_2) \ddot{q}_1 \} + \theta_3 \hat{q}_2 - gl_2 m_3 \cos(\hat{q}_1 + \hat{q}_2) \\ & + l_2 \mathcal{P}_{N,t_d} \{ \cos(q_1 + q_2) u_3 \} - \hat{u}_2 + \hat{q}_1^2 \theta_2 \sin(\hat{q}_1) \\ & - m_3 l_2 \mathcal{P}_{N,t_d} \{ \cos(q_1 + q_2) \ddot{q}_3 \} + \theta_3 \hat{q}_1, \quad (41b) \end{aligned}$$

with the estimated values $\hat{q}_1, \hat{q}_2, \hat{q}_1, \hat{q}_2, \hat{q}_1, \hat{q}_2, \hat{u}_1, \hat{u}_2$ and the partial approximation of the remaining products. The resulting fault estimate \hat{f} fulfils criteria I and II of RGP-FDI and is solely dependent on the available measurement signals.

For the simulation the masses m_1, m_2 and m_3 are chosen respectively as 10kg, 5kg and 2.35kg and the inertia J_1, J_2 and J_3 as 0.088kgm², 0.088kgm² and 0.088kgm². The lengths l_1, l_{1s}, l_2 and l_{2s} were parameterized to 0.325m, 0.1625m, 0.275m and 0.1375m and as gravitational acceleration $g = 9.81\text{ms}^{-2}$ was set. An exemplary pick and place movement with a quadratic spline interpolation between four positions was generated as the desired trajectory. In order to realize the trajectory tracking for the robot, a PI state controller with an additional feedforward control to compensate the nonlinearity was implemented. All eigenvalues of the state controller were set to $\lambda = -10$. The parameters for the polynomial approximation have been chosen to $\alpha = \beta = 3$ and $N = 1$, i.e., first order Jacobi polynomials are employed. Note that the approximation order $N = 1$ is sufficient to obtain a satisfactory reproduction of the fault signals. In order to improve the approximation accuracy the delay t_d was selected as the zero of the Jacobi polynomial of second order at $t_d = (LT_s)/3$ (see Sec. III-A) and correspondingly for discrete-time implementation of the simulation the sampling time of $T_s = 0.005\text{s}$ and $L = 20$ were set (see Sec. III-B). Furthermore, an additive gaussian distributed measurement noise $\bar{\omega}_y, \mathcal{N}_{\bar{\omega}_y}(0, 10^{-10})$ and process noise $\bar{\omega}_u, \mathcal{N}_{\bar{\omega}_u}(0, 10^{-3})$ for the variables q and u , respectively, were added for the simulation to verify the robustness of the method against noise. The disturbance d can be described by $d = \sigma(t - 0.5\text{s})(10\text{N} + 2\text{N} \sin(2t))$ with the Heaviside function $\sigma(t)$, whereby the faults f_1 and f_2 jump to the value $f_{1,\infty} = f_{2,\infty} = 10\text{Nm}$ at the time $t_{f1} = 1\text{s}$ and $t_{f2} = 3\text{s}$, respectively. The simulation results in Fig. 2 and Fig. 3 show the trajectories of q, u, f , and d for the exemplary pick and place movement. It becomes obvious that the reconstructed values \hat{f}_1 and \hat{f}_2 of the faults f_1 and f_2 are each independent of the input variable u and the disturbance d . The reconstruction of the fault thus depends solely on the parameters of the polynomial approximation. The simulation

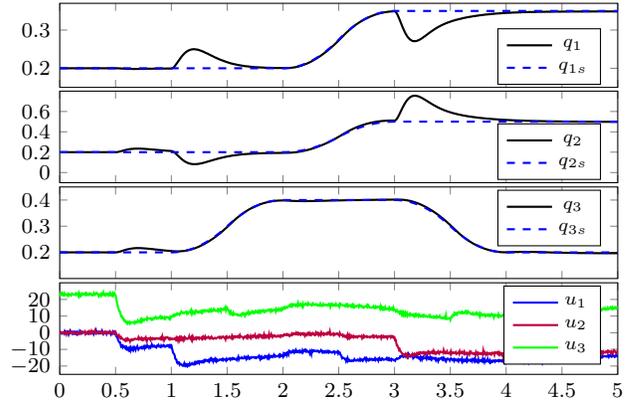


Fig. 2. Simulation results are the position values and the references as well as the corresponding control inputs for a pick and place movement of the SCARA.

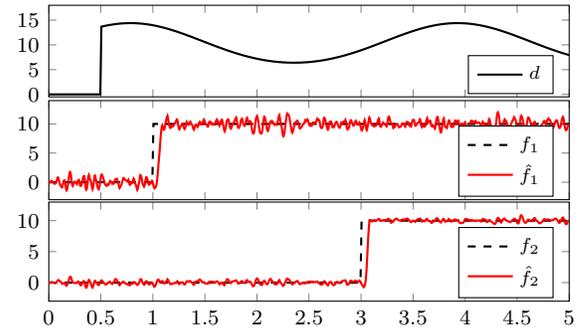


Fig. 3. Applied disturbance d , the obtained faults f_1, f_2 and their estimates \hat{f}_1, \hat{f}_2 in the presence of measurement and process noise.

results show that the faults were both reconstructed with a delay of 0.033s, which corresponds to the set delay t_d of the polynomial approximation. For the reconstruction it does not matter whether the respective measured variables or the disturbance can be approximated polynomially within the interval $\mathcal{I}_{t,T}$ or not. The only requirement is that the fault f can be sufficiently well approximated within the interval $\mathcal{I}_{t,T}$ by a polynomial of N -th degree.

V. CONCLUDING REMARKS

As discussed in this paper, faults in nonlinear mechanical systems can be detected or identified independently by the introduced polynomial approximation independent of the disturbance d and the form of the input or output signals. It should be noted that the presented method can be extended to other influences such as friction and elastic couplings in the joints. In addition to actuator faults, other faults, such as sensor and parameter faults, can also be detected and identified analogous to the presented method. In further research work, a more general class of nonlinear systems will also be investigated.

ACKNOWLEDGMENT

The authors kindly express their gratitude to the industrial research partner Siemens AG, Digital Industries Operating Company Erlangen for funding and supporting this project.

REFERENCES

- [1] J. Chen and R. J. Patton, *Robust Model-based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers, Norwell, 1999.
- [2] S. X. Ding, *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer, Berlin, 2008.
- [3] C. D. Persis and A. Isidori, "A geometric approach to nonlinear fault detection and isolation," *IEEE Trans. Automat. Contr.*, vol. 46, pp. 853–865, 2001.
- [4] P. M. Frank and B. Köppen-Seliger, "Fuzzy logic and neural network applications to fault diagnosis," *International Journal of Approximate Reasoning*, vol. 16, pp. 67 – 88, 1997.
- [5] S. Heo and J. H. Lee, "Fault detection and classification using artificial neural networks," *IFAC-PapersOnLine*, vol. 51, pp. 470 – 475, 2018.
- [6] P. Mai, C. Join, and J. Reger, "Flatness-based fault tolerant control of a nonlinear MIMO system using algebraic derivative estimation," in *3rd IFAC Symposium on System, Structure and Control, SSC'07, Foz do Iguassu, Brazil, 2007*, 2007.
- [7] M. Fliess, C. Join, and H. Sira-Ramírez, "Non-linear estimation is easy," *International Journal of Modelling Identification and Control*, vol. 4, pp. 12 – 27, 2007.
- [8] L. Kiltz, M. Mboup, and J. Rudolph, "Fault diagnosis on a magnetically supported plate," in *2012 1st International Conference on Systems and Computer Science (ICSCS), Lille, France*, pp. 1–6.
- [9] M. Mboup, C. Join, and M. Fliess, "Numerical differentiation with annihilators in noisy environment," *Numerical Algorithms*, vol. 50, pp. 439–467, 2009.
- [10] R. Isermann, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer, Berlin, 2005.
- [11] L. Kiltz and J. Rudolph, "Parametrization of algebraic numerical differentiators to achieve desired filter characteristics," in *Proceedings of the 52nd IEEE Conference on Decision and Control, CDC 2013, Firenze, Italy*, 2013, pp. 7010–7015.
- [12] G. Szegő, *Orthogonal Polynomials*, 23rd ed. American Mathematical Society, Providence, 1959.
- [13] D. G. Luenberger, *Optimization by Vector Space Methods*. John Wiley & Sons, Inc., New York, 1997.
- [14] R. Penrose, "A generalized inverse for matrices," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406–413, 1955.