Implementation of a Fractional Model–Based Fault Detection Algorithm into a PLC Controller

Ryszard Kopka
Opole University of Technology, Sosnkowskiego 31, 45-272 Opole, POLAND
E-mail: r.kopka@po.opole.pl

Abstract. This paper presents results related to the implementation of model–based fault detection and diagnosis procedures into a typical PLC controller. To construct the mathematical model and to implement the PID regulator, a non–integer order differential/integral calculation was used. Such an approach allows for more exact control of the process and more precise modelling. This is very crucial in model–based diagnostic methods. The theoretical results were verified on a real object in the form of a supercapacitor connected to a PLC controller by a dedicated electronic circuit controlled directly from the PLC outputs.

1. Introduction
Improving the functionality of modern equipment and systems requires the use of very advanced fault detection mechanisms or procedures. Most of them are based on mathematical models of processes, built by differential equations, neural networks or fuzzy logic. But the efficiency of model–based fault detection methods are related to the precision of the model [7, 8]. On the other hand, technology development generates new materials, new equipment or systems for which the use of traditional mathematical calculus is insufficient. An example could be activated carbon technology used in supercapacitors. The differential equation model of non–integer order, so–called fractional order, allows for the described phenomena in porous materials to yield more detail than the integer one.

There are many examples in the use of fractional calculus [5, 10]. However, these are generally used as experimental or laboratory solutions. The paper presents results related to the implementation of a fractional model and fractional regulator into a typical, industrial PLC controller. The main constraints in such systems are computational speed and insufficient memory.

The paper presents the results of fault detection procedures based on a plant model implemented into a PLC controller. In a fractional plant simulator, the supercapacitor charged by voltage–controlled voltage source was used. The fractional order model of the process and two fractional PI$^\lambda$D$^\mu$ regulators were all implemented into a PLC controller. The minimal sampling time and maximal length of data blocks were determined.

This article is organized as follows. In Section 2, the basics of model-based fault detection mechanism implementation are presented. Section 3 describes the principles of fractional order calculus, its discretization scheme and problems with practical implementation. The experimental test setup, its configuration, software implementation into the PLC controller and model–based fault detection procedure implementation are described in detail in Section 4. Next,
the results of model identification and PI$^1$D$^\mu$ regulator configuration are shown. The last two sections are test results and a conclusion.

2. Model–Based Fault Detection

There are different approaches for fault detection using mathematical models. The goal is to detect faults in the process using a residual signal, expressed as the difference between measurable process signals and mathematical models. Figure 1 shows the basic structure of model–based fault detection. Based on the measured input signal $u(t)$ and the output signal $y(t)$, the detection module generates a difference signal $d(t)$. By comparison with nominal values, the analysis module detects changes leading to faults [2, 4, 9]. Model–based methods require the knowledge of process dynamics in the form of mathematical structure and parameters. For process dynamics, the basic input/output models in the form of differential equations are used. But in most practical cases, the process parameters are not known. They must be determined with parameter estimation methods by measuring input and output signals.

3. Fractional Calculus

Fractional differential/integral calculus operator $aD_t^\alpha$ is a generalization of the integration and differentiation to the non–integer order, in which $a$ and $t$ denote the limits of the operation and $\alpha$ denotes the fractional order such that

$$aD_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \text{for } \alpha > 0 \\ 1 & \text{for } \alpha = 0 \\ \int_a^t d\tau^{-\alpha} & \text{for } \alpha < 0, \end{cases}$$

where it is assumed that $\alpha \in \mathbb{R}$. In the literature, there are a few definitions of the fractional order differ/integral operator that may be useful in a specific situation [1, 3]. One of the most widely used is the Grünwald–Letnikov. According to it, the fractional operator $aD_t^\alpha$ is defined as

$$GLaD_t^\alpha f(t) = \lim_{h \to 0} \frac{1}{h^\alpha} \sum_{j=0}^{[\frac{t-a}{h}]} (-1)^j \binom{\alpha}{j} f(t-jh),$$

where $h$ is the time period, $m-1 < \alpha \leq m$ while $m \in \mathbb{N}$, and $[\kappa]$ is the integer part of $\kappa$. The binomial $\binom{\alpha}{j}$ is calculated as

$$\binom{\alpha}{j} = \begin{cases} 1 & \text{for } j = 0 \\ \frac{\alpha(\alpha-1)\ldots(\alpha-1+j)}{j!} & \text{for } j > 0. \end{cases}$$

Figure 1. General scheme of model–based fault detection.
To obtain the fractional model and controller in discrete time, the definition of Grünwald–Letnikov in discrete form was introduced as

$$\Delta_h^\alpha f(t)_{|t=kh} = \frac{1}{h^\alpha} \sum_{j=0}^{k} (-1)^j \binom{\alpha}{j} f(t - jh),$$  \hspace{1cm} (4)

where $t = kh$ is the discrete sample in continuous time. Converting continuous time samples into a discrete domain $k = 0, 1, ..., \text{equation (4) can be written as}$

$$\Delta_h^\alpha f(t) = \frac{1}{h^\alpha} \sum_{j=0}^{t} (-1)^j \binom{\alpha}{j} f(t - j).$$  \hspace{1cm} (5)

There are a few discretization schemes of the Grünwald–Letnikov equation. The most commonly used are the backward difference (Euler), trapezoidal (Tustin) rule, and the Al–Alaoui operator. Taking the Euler rule, the fractional order difference in the discrete time domain for sequential time sample $k$ can be calculated as

$$\Delta_h^\alpha x_k = \frac{1}{h^\alpha} \sum_{j=0}^{k} (-1)^j \binom{\alpha}{j} x_{k-j}.$$  \hspace{1cm} (6)

The fractional order result according to (6) depends on all earlier samples. This is a major advantage of the fractional differ/integral calculus in relation to the integer order. But the infinite summation all previous measurements according to equation (6), in real devices must be limited to finite samples $L + 1$. This is constrained by both the limited amount of memory and the calculation time. According to this, equation (6) takes the form

$$\Delta_h^\alpha x_k = \frac{1}{h^\alpha} \sum_{j=0}^{L} (-1)^j \binom{\alpha}{j} x_{k-j}.$$  \hspace{1cm} (7)

The limitation of sample number reduces the accuracy expressed by equation (7). This is why the number of samples needs to be as large as possible. But in industrial controllers, which operate in real-time and have a limited memory, this is a serious problem.

4. Experimental Test Setup

The experimental test setup consists of an electronic circuit that simulates part of a real plant. A special electronic element known as a supercapacitor makes proper use of fractional calculus. The plant simulator is connected to a typical PLC controller by dedicated electronic circuits with voltage–controlled voltage and current sources. A fractional order $\text{PI}^\lambda \text{D}^\mu$ regulator and fractional plant model were both implemented into the PLC controller.

4.1. Fractional order $\text{PI}^\lambda \text{D}^\mu$ regulator

The scheme of the fractional $\text{PI}^\lambda \text{D}^\mu$ regulator is shown in Fig. 2. It has a typical PID configuration, except that integration and differentiation are of fractional order. The regulator transfer function is defined as

$$G_R(s) = K_p + \frac{K_i}{s^\lambda} + K_ds^\mu,$$  \hspace{1cm} (8)

where integration and differentiation are of the orders $\lambda$ and $\mu$, respectively, with $\lambda, \mu \in \mathbb{R}$. To implement fractional calculus, according to equation (7), the arrays of the binomials $\binom{\lambda}{j}$ and $\binom{\mu}{j}$ of length $L + 1$ have to be calculated first.
4.2. Fractional order plant model

The plant was simulated by an electronic circuit built with resistors and a supercapacitor, as shown in Fig. 3(a). To implement the model into the PLC controller, the plant in the form of differential equations was defined as

\[
y(t) + T_1 \frac{d^\alpha y(t)}{dt^\alpha} = u(t) + T_2 \frac{d^\alpha u(t)}{dt^\alpha},
\]

where \( T_1 = (R_1 + R_2)C \) and \( T_2 = R_2C \). The resistance \( R_2 \) represents the series connection of \( R_{I1} \) and \( r_{ESR} \) (\( r_{ESR} \) represents the equivalent series resistance of the supercapacitor, see Fig. 3(b) and [6]). After simple calculations it can be presented as

\[
\frac{d^\alpha y(t)}{dt^\alpha} = \frac{1}{(R_1 + R_2)C}(u(t) - y(t)) + \frac{R_2}{R_1 + R_2} \frac{d^\alpha u(t)}{dt^\alpha}.
\]

Based on (10), the mathematical fractional model of the plant was implemented into the PLC controller.

4.3. Realization of fractional calculus into PLC controller

The fractional calculus was implemented into the PLC controller according to the scheme shown in Fig. 4. In the main memory, seven real-number arrays of length \( L + 1 \) were defined, where
\( L = 900 \). All binomials and sign arrays were calculated in the Startup Block. After that, the Main Block was started. The *Time Interrupt* was generated with sampling time \( t_s = 250 \text{ ms} \), independent of the Main Block \([11]\). During every *Time Interrupt* procedure, the control signals for both plant and model output were calculated. According to equation (7), the calculations of the fractional PI\( ^{\lambda}D^{\mu} \) regulator and the fractional plant model outputs need to provide six times the \( L + 1 \) summation: two for each regulator and two for the model output. In every *Time Interrupt* the actual value of the plant output was measured. Next, the error signals for plant and model PI\( ^{\lambda}D^{\mu} \) regulators were calculated. These values were introduced into two error arrays (DB6 and DB7). Based on them, the integral and differential calculations for PI\( ^{\lambda}D^{\mu} \) regulators were provided. The control signal for plant steering was written to the analog output \( (A_{\text{Out}0}) \), while the control signal for the model was used to calculate the model output. All calculations were made during \( t_C \approx 230 \text{ ms} \). The analog output signal from the PLC was steering the voltage source with a current intensity high enough to charge the supercapacitor. The actual voltage level was measured on the analog input \( (A_{\text{In}0}) \), while the supercapacitor current was measured by the voltage drop on the \( R_2 \) resistor \( (A_{\text{In}1}) \). The fault was simulated by a voltage-controlled current source discharging supercapacitor. The current source intensity was steered by voltage changes on the analog output \( (A_{\text{Out}1}) \). The detailed wiring diagram is shown in Fig. 3(b).

4.4. Configuration of model-based fault detecting structure

Based on the scheme in Fig. 1, a detailed configuration of the implemented model–based fault detection mechanism is shown in Fig. 5. As a fault indicator, the difference signal between the PI\( ^{\lambda}D^{\mu} \) regulators was taken. This solution allows one to keep the plant and model output signals at the same levels regardless of the faults. This guarantees the same dynamic behaviour and set point.
5. Model Parameters Identification and PID Configuration

There are several kinds of identification methods of plant model parameters. The goal of the time-domain identification process used in this work, was to obtain a fractional model parameters vector in the form of (10). An output error method was used with a non-linear least-square approach. The optimization criterion was the output error norm $\| \epsilon_k \|_2^2$ given by

$$\epsilon_k = y_k - \hat{y}_k,$$

where $y_k$ was the plant output signal and $\hat{y}_k$ was obtained by simulation of the identified model under the input signal $u_k$. The best method to estimate the parameters so as to minimize the output error is numerical optimization by means of iterative procedures. $\theta$ denotes the vector of the identified parameters and $y(t)$ and $\hat{y}(t)$ represent the vectors of measured and calculated data, during simulations. The identification problem is to find a vector $\theta \in \Theta_{ad}$ that minimizes the quadratic criterion $J$, that is

$$\min_{\theta \in \Theta_{ad}} J,$$

where

$$J = \int_0^t e(\tau)^T W e(\tau) d(\tau).$$

$\Theta_{ad}$ stands for the set of admissible parameters, $W = W^T$ is the positive definite weighting matrix and $t$ denotes the simulation time. The problem (12) can be solved using many optimization algorithms. In this paper, the identification procedure was implemented in MATLAB® environment using the Nelder–Mead simplex method. The results of the model identification are presented in Tab. 1.

Several methods of tuning fractional PI$^\lambda$D$^\mu$ controllers are proposed. However, there is still no general one. Some authors propose optimization techniques applied to the tuning problem instead. But there are several problems in designing a fractional PI$^\lambda$D$^\mu$ controller via optimization. These include the type of plant to be controlled, optimization criteria, parameters to optimize or methods to obtain initial parameters. The results of the PI$^\lambda$D$^\mu$ regulator tuning procedure are presented in Tab. 2. Figure 6(a) shows the step response output signals of the plant and model for different step values. One can see the identity quite well. Figure 6(b), however, shows the closed-loop plant and model-output signals with regulator parameters presented in Tab. 2.

6. Test Results

The tests were carried out with and without fault events. The faults had abrupt or incipient form. Fig. 7 shows the plant $y_{Pk}$ and model $y_{Mk}$ output signals under the reference signal $r_k$. 

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{figure5.png}
\caption{Implementation of the fault detection algorithm based on plant and model – control difference signals.}
\end{figure}
Table 1. Model identification parameters.

| Parameter | Identified Value |
|-----------|------------------|
| C         | 0.31 F           |
| $R_1$     | 85.82 Ω          |
| $R_2$     | 71.07 Ω          |
| $\alpha$ | −0.96            |

Table 2. PI$^\lambda$D$^\mu$ configuration parameters.

| Parameter | Value   |
|-----------|---------|
| $K_p$     | 0.01    |
| $K_i$     | 8.3     |
| $K_d$     | 0.1     |
| $\lambda$| −0.9    |
| $\mu$     | 0.05    |

Figure 6. Process and model step responses for different reference values (a) and closed–loop output signals with PI$^\lambda$D$^\mu$ regulators (b).

The difference signal $d_k$ was still contained within the area bounded by two threshold levels. Figure 8 shows the same signals but the plant has been damaged by an abrupt and incipient fault signal $I_{f_k}$. Although the control process was still provided ($y_{P_k}$ followed the reference signal $r_k$), faults can be clearly seen on the difference signal $d_k$, which crossed the threshold line.

7. Conclusions

The growth in reliability demands result in developing new methods and procedures to more precisely and effectively diagnose and detect faults. In model–based methods, the crucial element is the precision of the mathematical model of the protected plant. It seems that fractional calculus meets these expectations. Related works show that scientific solutions developed in research laboratories can be implemented even in typical industrial controllers, although there are some limitations, such as calculation speed and memory size.

References

[1] Barbosa S. R., Tenreiro Machado J. A., Implementation of Discrete-Time Fractional-Order Controllers based on LS Approximations, Acta Polytechnica Hungarica, Vol. 3, No. 4, pp. 5–22 (2006)
[2] Blanke M., Kinnaert M., Lunze J., Staroswiecki M., Diagnosis and Fault-Tolerant Control, 2nd Edition, Springer-Verlag, Berlin Heidelberg (2006)
[3] Das S., Functional Fractional Calculus for System Identification and Controls, Springer-Verlag, Berlin Heidelberg (2008)
Figure 7. Difference signal $d(t)$ without faults.

Figure 8. Difference signal $d(t)$ with abrupt and incipient faults.

[4] Isermann R., Model-based fault-detection and diagnosis status and applications, Annual Reviews in Control, 29, pp. 71–85 (2005)
[5] Kaczorek T., Selected Problems of Fractional Systems Theory, Springer-Verlag, Berlin Heidelberg (2011)
[6] Kopka R., Tarczynski W., Measurement system for supercapacitors equivalent parameters determination, Metrology and Measurement Systems, Vol. XX, No. 4, pp. 581–590, (2013).
[7] Korbicz J., Kościelny J.M., Kowalczyk Z., Cholewa W. (Eds.), Fault Diagnosis, Springer-Verlag, Berlin Heidelberg (2004)
[8] Korbicz J., Kościelny J.M. (Eds.), Modeling, Diagnostics and Process Control, Springer-Verlag, Berlin Heidelberg (2010)
[9] Noura H., Theilliol D., Ponsart J-Ch., Chamseddine A., Fault-tolerant Control Systems, Springer-Verlag, London (2009)
[10] Podlubny I., Fractional-order Systems and Fractional-order Controllers. Slovak Academy of Sciences Institute of Experimental Physics, UEF SAV, Kosice (1994)
[11] Siemens Manual, Sinbis 3T-1200 Easy Book, www.automatyka.siemens.pl, (2012)