Model Predictive Control for Fractional-order System
A Modeling and Approximation Based Analysis

Mandar M. Joshi[1], Vishwesh A. Vyawahare[2] and Mukesh D. Patil[2]

[1]*Department of Electrical Engineering, Government College of Engineering, Pune, India*

[2]*Department of Electronics Engineering, Ramrao Adik Institute of Technology, Nerul, Navi Mumbai, India*

Keywords: Predictive Control, Fractional Calculus, Fractional Systems, Integer-order Approximation, Internal Model Control.

Abstract: A widely recognized advanced control methodology model predictive control is applied to solve a classical servo problem in the context of linear fractional-order (FO) system with the help of an approximation method. In model predictive control, a finite horizon optimal control problem is solved at each sampling instant to obtain the current control action. The optimization delivers an optimal control sequence and the first control thus obtained is applied to the plant. An important constituent of this type of control is the accuracy of the model. For a system with fractional dynamics, accurate model can be obtained using fractional calculus. One of the methods to implement such a model for control purpose is Oustaloup's recursive approximation. This method delivers equivalent integer-order transfer function for a fractional-order system, which is then utilized as an internal model in model predictive control. Analytically calculated output equation for FO system has been utilized to represent process model to make simulations look more realistic by considering current and initial states in process model. The paper attempts to present the effect of modeling and approximations of fractional-order system on the performance of model predictive control strategy.

1 INTRODUCTION

Model predictive control (MPC) is an optimal control theory based on numerical optimization. Future control efforts and future plant responses are predicted using a system model and optimized at regular intervals with respect to a performance index. It is a computational technique for improving control performance in applications and processes in chemical and petrochemical industries. Predictive control has become arguably the most widespread advanced control methodology currently in use in industry (Muske and Rawlings, 1993; Rawlings, 2000; Morari and Lee, 1999; Bemporad, 2006; Garcia et al., 1989).

The basic MPC concept can be explained as follows. In MPC, model of the system is used to predict the future output and control efforts required to attain the reference trajectory. Hence, the accuracy of the model determines the control and delivers precise future input trajectory in order to follow the reference signal. This is the basic philosophy of any MPC. MPC is more a methodology and not a single technique and it is known by many different nomenclatures such as Model Predictive Control (MPC), Model based predictive control (MBPC), Receding horizon control (RHC), Moving horizon control (MHC), Internal Model Control (IMC) etc.

The literature survey reveals that MPC has made outstanding contributions for solving control problems in industry (J.M.Maciejowski,). It has so far been applied mainly in the petrochemical industry, but is currently being increasingly applied in other sectors of the process industry. The main advantages of using MPC in these applications are:

1. Inherent handling of multivariable control problems.
2. Actuator limitations are considered.
3. Operation closer to constraints is possible for more profitability.
4. Applications with relatively low control update rates provides sufficient time for the necessary on-line computations.

Fractional calculus is used for modeling real-world and man-made systems more compactly and faithfully. Fractional calculus has been known from 1695, when Leibnitz introduced the symbol for the n^{th} derivative $d^n y/dx^n$, and L'Hopital discussed the

Joshi M., Vyawahare V. and Patil M..
Model Predictive Control for Fractional-order System - A Modeling and Approximation Based Analysis.
DOI: 10.5220/0005038203610372
In *Proceedings of the 4th International Conference on Simulation and Modeling Methodologies, Technologies and Applications* (SIMULTECH-2014), pages 361-372
ISBN: 978-989-758-038-3

possibility that ‘n’ could be $1/2$. Nonetheless, this branch of mathematical analysis was really developed in the 19^{th} century by Liouville, Riemann, Letnikov and others (Monje et al., 2010). One of the most popular examples of fractional-order modeling of a physical system is the semi-infinite lossy transmission line (Boudjehem and Boudjehem, 2012). Other systems with fractional-order dynamics are viscoelasticity, dielectric polarization and electromagnetic waves (Boudjehem and Boudjehem, 2012). Fractional-order control can be regarded as the generalization of the conventional integer-order control theory. Fractional-order models represent a system in more accurate ways (Boudjehem and Boudjehem, 2012). It is worth exploring the introduction of fractional theory in MPC. There are some systems which are of fractional nature and hence a fractional-order model can be used to represent such systems instead of integer-order models.

For implementation of MPC, a state-space model is required and approximation method helps to obtain the same. A fractional-order Transfer Function (FOTF) can not be converted to a state-space model through simulation softwares like MATLAB. Therefore, we need to get an approximation that is, an integer-order equivalent (transfer function) which can represent that FOTF and also can be converted to state-space model using MATLAB. Moreover, an approximation has to be made for a fractional operator s^{α} or $s^{-\alpha}$ with $0 < \alpha < 1$.

In this paper, MPC is applied to an FO system with different fractional-order α for which Oustaloup’s recursive approximation has been used to obtain respective models. Obtained models are then converted to acquire strictly proper natured discrete state-space models by adding a pole at far away from the origin. Authenticity of these models were checked with the original one through frequency and step response. Finally, these models were used for prediction purposes in MPC. In order to make MPC realistic, analytically calculated output equation for FO systems has been utilized to represent process output.

The paper is organized as follows. In Section 2, definition of fractional calculus and dynamics of FO systems are reviewed; Section 3 presents the Oustaloup’s recursive approximation method and its application for fractional operators. Philosophy behind MPC and formulation of MPC problem is covered in section 4. Section 5 presents the results obtained for MPC with FO systems and section 6 concludes this work with some observations.

2 FRACTIONAL CALCULUS

Fractional Calculus can be defined as integration and differentiation of non-integer order. In the development of the theory of fractional calculus, there are different definitions available for fractional-order integration and differentiation such as Riemann-Liouville fractional differentiation and integration, Caputo fractional differentiation, Grunwald-Letnikov fractional differentiation (YaLi and Ruikun, 2010). Caputo fractional differentiation is defined as (Podlubny, 1998)

$$ {}_{0}^{C}D_t^{\alpha} f(t) = \frac{1}{\Gamma(m-\alpha)} \int_0^t (t-\tau)^{(m-\alpha-1)} D_{\tau}^{m} f(\tau) d\tau, \quad (1) $$

where α is the order of the differentiation and m is the nearest integer on higher side of α.

2.1 Fractional-order Systems

Fractional-order systems are best represented by fractional-order models. Also, for integer-order systems, fractional-order models give better predictions (Boudjehem and Boudjehem, 2012; Hortelano et al., 2010; Romero et al., 2010). Fractional-order systems are theoretically of infinite order (Caponetto et al.,). A fractional-order system can be represented mathematically by following fractional-order differential equation (FDE) (Podlubny, 1998)

$$
\begin{aligned}
a_n D^{\alpha_n} y(t) \quad &+ \quad a_{n-1} D^{\alpha_{n-1}} y(t) + \cdots \\
&+ \quad a_1 D^{\alpha_1} y(t) + a_0 y(t) \\
&= \quad b_m D^{\beta_m} u(t) + b_{m-1} D^{\beta_{m-1}} u(t) + \cdots \\
&+ \quad b_1 D^{\beta_1} u(t) + b_0 u(t), \qquad (2)
\end{aligned}
$$

where D^{α_i} and D^{β_i} represents Caputo or Riemann Liouville Fractional Derivative (RLFD).

Using Laplace transform, one can obtain the transfer function for fractional system. The order of the fractional-order Transfer Function (FOTF) is theoretically infinite since a fractional-order (FO) system is basically infinite dimensional system (Podlubny, 1998). But for convention we assume that the order of a FOTF is equal to the highest degree of denominator which will be a real number and not necessarily an integer. In an FOTF, the numerator and denominator are not polynomials but are pseudo-polynomials.

3 APPROXIMATION METHOD

Approximation methods are based on different techniques. In continuous domain there are three types of techniques on which approximations are based.

Curve fitting/frequency response matching, continued fraction expansion (CFE) and regular Newton process for iterative approximation (Vinagre et al., 2000). Only Curve fitting/frequency response matching is briefly explained below since it is the principle on which Oustaloup's recursive approximation is based.

3.1 Curve Fitting/ Frequency Response Matching

In this technique, frequency domain identification is used to obtain the rational function (integer-order function) of a given fractional operator. For example, we will try to get an integer-order function of s^{α} by matching their frequency response. This can be written as an optimization problem, where the cost function will be of the form (Vinagre et al., 2000)

$$J = \int W(w)|G(w) - \hat{G}(w)|^2 dw, \tag{3}$$

where $W(w)$ is the weighting function and $G(w)$ represents the frequency response of original function which is in this case is s^{α}. $\hat{G}(w)$ represents the frequency response of integer-order approximation of s^{α}. We will minimize the cost function in order to achieve the best approximation. This is the basic idea behind approximation methods based on curve fitting (Vinagre et al., 2000).

3.2 Oustaloup Recursive Approximation

It gives the approximation for a fractional-order differentiator s^{α} and is widely used. This approximation is given by following equations (Monje et al., 2010)(Valerio and Costa, 2011):

$$s^{\alpha} \approx K \prod_{k=1}^{N} \frac{s + w'_k}{s + w_k}, \tag{4}$$

where the poles, zeros and gain can be evaluated as (Monje et al., 2010):

$$w'_k = w_l w_u^{(2k-1-\alpha)/N}, \tag{5}$$

$$w_k = w_l w_u^{(2k-1+\alpha)/N}, \tag{6}$$

$$K = w_h^{\alpha}, \tag{7}$$

$$w_u = \sqrt{w_h / w_l}, \tag{8}$$

where w_l and w_h are lower and higher frequency values. It should be noted that small values of N obviously result in low-order approximations which are simple in nature but at the ripples in both Bode gain and phase plots. To practically eliminate such a ripple, N has to be increased. A higher value of N makes the computation heavier (Boudjehem and Boudjehem, 2012).

3.3 Comparison of FO System and Its Approximation

This section contains bode plots and step response for the FO system for which the main results of this paper are shown. The FO plant given below is used for five different values of $\alpha = 0.1,\ 0.3,\ 0.5,\ 0.7\ and\ 0.9$; $(a = b = 1)$ thus giving five different FO systems.

$$G(s) = \frac{a}{s^{\alpha} + b} \quad a, b \in \Re. \tag{9}$$

The FO system shown above is used here since it represents model of some physical systems and has been used as a benchmark system in (Poinot and Trigeassou, 2003) (Poinot and Trigeassou, 2004). Few examples of physical systems represented by (9) are: model of beam heating process (Dzieliński and Sierociuk, 2010), model of thermal systems (Gabano and Poinot, 2011) and an explicit systems model for electrical networks (Enacheanu et al., 2006). Above FO system (9) was also used for approximation of high order integer systems (Mansouri et al., 2010).

In the plots (shown below) the original FO system (9) is compared with its obtained approximation via frequency as well as step response. FOTF represents the original FO system and Oustaloup represents approximation obtained for the FO system. To obtain frequency response of original system (9), s is replaced by jw and then the system is simulated for a fixed frequency range (in this case 10^{-2} to 10^{6} rad/sec) to obtain the magnitude and phase vectors. The bode plot is obtained by multiplying suitable factors to the magnitude and phase vector. To obtain the approximation of fractional operator s^{α} in (9), Oustaloup recursive approximation (4) was used. An example of approximation for one of the FO systems is shown below.

3.3.1 Approximation of $\frac{1}{s^{0.5}+1}$

Approximation was carried out for $s^{0.5}$ and following results were obtained:

$$s^{0.5} = \frac{N(s)}{D(s)}, \tag{10}$$

where

$$
\begin{aligned}
N(s) = {} & 1000s^{10} + 2.985e^{8}s^{9} + 1.219e^{13}s^{8} \\
& + 7.722e^{16}s^{7} + 7.727e^{19}s^{6} + 1.225e^{22}s^{5} \\
& + 3.076e^{23}s^{4} + 1.224e^{24}s^{3} + 7.691e^{23}s^{2} \\
& + 7.498e^{22}s + 1e^{21}, \\
D(s) = {} & s^{10} + 7.498e^{5}s^{9} + 7.691e^{10}s^{8} \\
& + 1.224e^{15}s^{7} + 3.076e^{18}s^{6} + 1.225e^{21}s^{5} \\
& + 7.727e^{22}s^{4} + 7.722e^{23}s^{3} + 1.219e^{24}s^{2} \\
& + 2.985e^{23}s + 1e^{22}.
\end{aligned}
$$

Adding one to $s^{0.5}$ and then inverting it will lead to an expression for $1/(s^{0.5}+1)$, which is:

$$\frac{1}{s^{0.5}+1} = \frac{N1(s)}{D1(s)}, \tag{12}$$

where

$$
\begin{aligned}
N1(s) = {} & s^{10} + 7.498e^{5}s^{9} + 7.691e^{10}s^{8} \\
& + 1.224e^{15}s^{7} + 3.076e^{18}s^{6} + 1.225e^{21}s^{5} \\
& + 7.727e^{22}s^{4} + 7.722e^{23}s^{3} + 1.219e^{24}s^{2} \\
& + 2.985e^{23}s + 1e^{22}, \\
D1(s) = {} & 1001s^{10} + 2.992e^{8}s^{9} + 1.227e^{13}s^{8} \\
& + 7.844e^{16}s^{7} + 8.034e^{19}s^{6} + 1.347e^{22}s^{5} \\
& + 3.849e^{23}s^{4} + 1.996e^{24}s^{3} + 1.988e^{24}s^{2} \\
& + 3.735e^{23}s + 1.1e^{22}.
\end{aligned}
$$

A pole at far end from the left hand side of imaginary axis was added to convert the above obtained system into strictly proper form; explanation for this conversion is given in subsection 4.4. The expression for $1/(s^{0.5}+1)$ becomes:

$$\frac{1}{s^{0.5}+1} = \frac{N2(s)}{D2(s)}, \tag{14}$$

where

$$
\begin{aligned}
N2(s) = {} & s^{10} + 7.498e^{5}s^{9} + 7.691e^{10}s^{8} \\
& + 1.224e^{15}s^{7} + 3.076e^{18}s^{6} + 1.225e^{21}s^{5} \\
& + 7.727e^{22}s^{4} + 7.722e^{23}s^{3} + 1.219e^{24}s^{2} \\
& + 2.985e^{23}s + 1e^{22}, \\
D2(s) = {} & 1.586e^{-13}s^{11} + 1001s^{10} + 2.992e^{8}s^{9} \\
& + 1.227e^{13}s^{8} + 7.844e^{16}s^{7} + 8.034e^{19}s^{6} \\
& + 1.347e^{22}s^{5} + 3.849e^{23}s^{4} + 1.996e^{24}s^{3} \\
& + 1.988e^{24}s^{2} + 3.735e^{23}s + 1.1e^{22}.
\end{aligned}
$$

Similarly approximation for other four FO plants was evaluated. Once, an integer-order equivalent (transfer function) is obtained for FO plant, traditional methods can be used to plot frequency and step response (Nise, 2007).

3.3.2 Frequency response of $\frac{1}{s^{\alpha}+1}$

Figures below show the frequency domain validation of the IO approximations for different values of α (plot for $\alpha = 0.3$ is not shown due to space constraints).

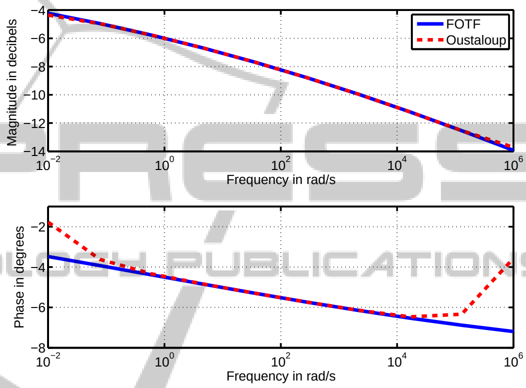

Figure 1: Bode Plot for $G_1(s) = \frac{1}{s^{0.1}+1}$.

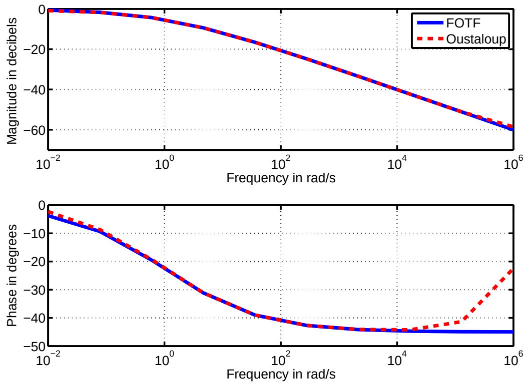

Figure 2: Bode Plot for $G_3(s) = \frac{1}{s^{0.5}+1}$.

It can be observed that for the frequency range of 10^0 to 10^4 rad/sec we get an exact match for the frequency response, which advocates that the obtained approximation can replace the original FOTF for the said frequency range.

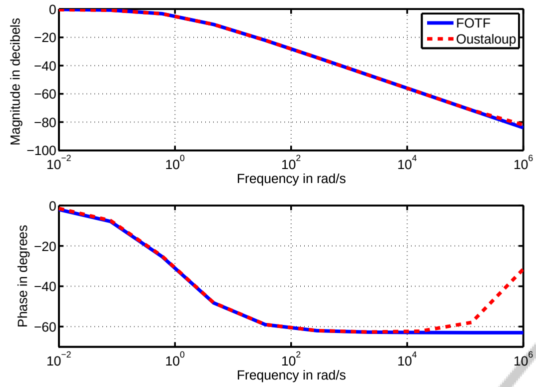

Figure 3: Bode Plot for $G_4(s) = \frac{1}{s^{0.7}+1}$.

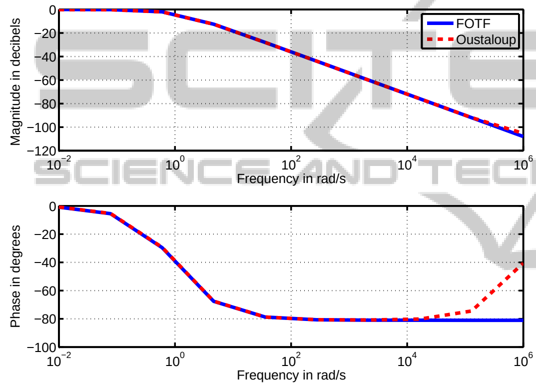

Figure 4: Bode Plot for $G_5(s) = \frac{1}{s^{0.9}+1}$.

3.3.3 Step Response of $\frac{1}{s^{\alpha}+1}$

To obtain the expression for step response of FO plant, the transfer function (9) is rewritten such that an expression can be obtained in terms of $Y(s)$ and $U(s)$, where the first is output and the later is the input.

$$Y(s) = \frac{U(s)}{s^{\alpha}+1}. \tag{16}$$

Now for step input $U(s) = 1/s$ and then applying inverse Laplace transformation (Monje et al., 2010), gives:

$$Y(t) = 1 - E_{\alpha}(-t^{\alpha}), \tag{17}$$

where $E_{\alpha}(-t^{\alpha})$ is one parameter Mittag-Leffler Function (MLF) (Podlubny et al., 2002). Figures below show the time domain validation of the IO approximations for different values of α (plot for $\alpha = 0.3$ is not shown due to space constraints).

From both frequency and step response it is observed that the dynamics of original FO system is still retained in the obtained IO approximation and hence using this approximation for analysis and simulation in model predictive control as a finite-dimensional model of original FO system is justified.

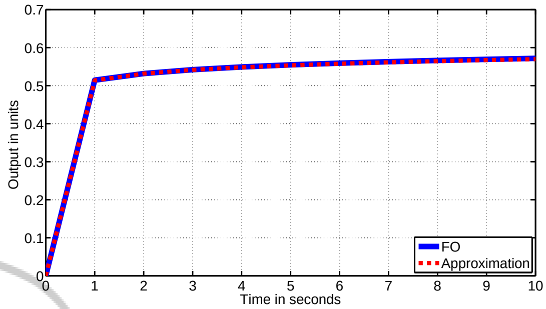

Figure 5: Step response of $G_1(s) = \frac{1}{s^{0.1}+1}$.

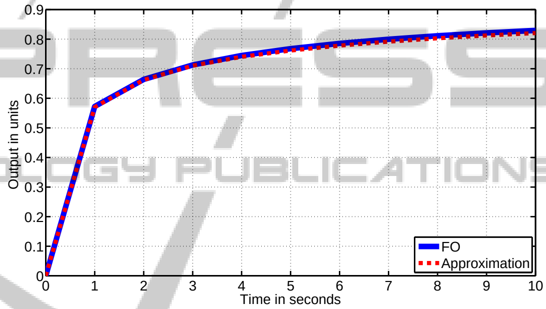

Figure 6: Step response of $G_3(s) = \frac{1}{s^{0.5}+1}$.

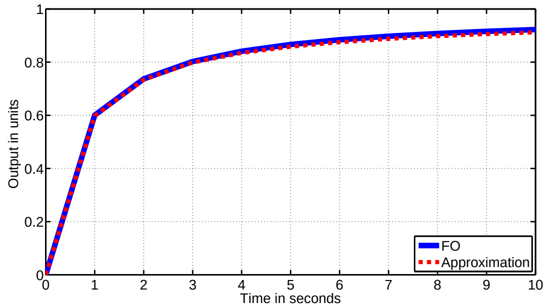

Figure 7: Step response of $G_4(s) = \frac{1}{s^{0.7}+1}$.

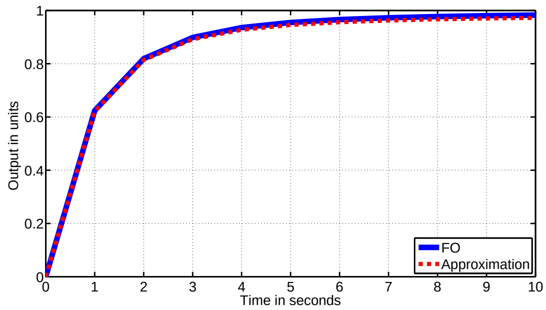

Figure 8: Step response of $G_5(s) = \frac{1}{s^{0.9}+1}$.

4 MODEL PREDICTIVE CONTROL

In this section, the theory of MPC is discussed in detail.

4.1 Principle of Model Predictive Control

For a system, the reference trajectory r(k) is defined as the ideal trajectory along which a plant should return to the set-point trajectory w(k), at any instant k. The current error between the output signal y(k) and the setpoint is an error represented as:

$$err(k) = w(k) - y(k). \tag{18}$$

Then the reference trajectory is chosen such that if the output followed it exactly, then the error after i steps would be

$$err(k+i) = exp^{-iT_s/T_{ref}} err(k), \tag{19}$$

where T_{ref} is the time constant of exponential assuming that reference trajectory approaches the set-point exponentially and T_s is the sampling interval. Therefore, the reference trajectory is defined as

$$r(k+i|k) = w(k+i) - y(k+i). \tag{20}$$

The notation $r(k+i|k)$ implies that the reference trajectory depends on the conditions at time k (J.M.Maciejowski,) (Boudjehem and Boudjehem, 2012).

Now, the predictive controller uses an internal model to estimate the future behavior of the plant. This process starts at the current time k and carries over a future prediction horizon H_P. This predicted behavior depends on the assumed control effort trajectory $\hat{u}$ that is to be applied over the prediction horizon and we have to select the input which promises the best predicted behavior. The notation $\hat{u}$ represents the predicted value input at time $k+i$ at instant k; the actual input $u(k+i)$ may be different than $\hat{u}$. After a future trajectory is decided, only the first element of the trajectory is required as the control input to the process. Hence $u(k)$ becomes $\hat{u}(k|k)$. The tasks of prediction, measurement of output and determination of input trajectory are repeated at every sampling instant. This is known as the receding horizon strategy, since prediction horizon slides along by one sampling interval at each step while its length remains the same.

Formulation of MPC problem needs few basic elements such as model, cost function, prediction equations and control law. These are covered in the next section.

4.2 Problem Formulation

In MPC, the first element of paramount importance is the model of the process/plant, which is required for prediction of the future states. This model must show the dependance of the output on the current measured variable and the current inputs. A precise model will deliver accurate predictions, but for MPC we do not always need precise model though they are always appreciated. Since the decisions that is, the optimal control effort is updated regularly, any model uncertainty can be dealt within a fair range (Rossiter, 2003). Hence, accuracy of the model used for the prediction can be compensated with regular updating of states. In this paper, FO system is used as a plant/process. Transfer function of the FO plant is:

$$G(s) = \frac{1}{s^{\alpha}+1}, \tag{21}$$

where α is varied between 0 and 1.
Second element is the cost function, which is required to evaluate the input trajectory such that it minimizes/maximizes the cost function. Selection of the cost function is a crucial issue and involves engineering and theocratical expertise. The cost function should be as simple as one can get away with for the desired performance. Choice of cost function affects the complexity of the implied optimization and hence it should taken into consideration that the cost function should be simple enough to have a straightforward optimization (Rossiter, 2003). For this reason 2-norm measures are popular and the cost function used in this paper is also a 2-norm measure.

$$J = \sum_{i=1}^{n_y} \|r_{k+i} - y_{k+i}\|_2^2 + \lambda \sum_{i=0}^{n_u-1} \|\Delta u_{k+i}\|_2^2 \tag{22a}$$

$$= \sum_{i=1}^{n_y} \|e_{k+i}\|_2^2 + \lambda \sum_{i=0}^{n_u-1} \|\Delta u_{k+i}\|_2^2, \tag{22b}$$

where λ is the weighting scalar; first term represents the sum of squares of the predicted tracking errors from an initial horizon to an output horizon n_y and the second term represents the sum of squares of the control over the control horizon n_u. It is assumed that control increments are zero beyond the control horizon, that is

$$\Delta u_{k+i|k} = 0, i \geq n_u.$$

The third element is the optimization, which is used to evaluate the control law. The control law is computed from the minimization of the cost J (22a) with respect to n_u future control moves, that is $\Delta \underset{\rightarrow}{u}$. It is denoted as

$$\min_{\underset{\rightarrow}{\Delta u}} J = \| \underset{\rightarrow}{r} - \underset{\rightarrow}{y} \|_2^2 + \lambda \| \Delta \underset{\rightarrow}{u} \|_2^2, \tag{23a}$$

$$= \| \underset{\rightarrow}{e} \|_2^2 + \lambda \| \Delta \underset{\rightarrow}{u} \|_2^2, \tag{23b}$$

where

$$\underset{\rightarrow}{r} = \begin{bmatrix} r(k+1) & r(k+2) & \cdots \end{bmatrix}^T. \tag{24}$$

$$\underset{\rightarrow}{y} = \begin{bmatrix} y(k+1) & y(k+2) & \cdots \end{bmatrix}^T. \tag{25}$$

$$\Delta \underset{\rightarrow}{u} = \begin{bmatrix} u(k+1) & u(k+2) & \cdots \end{bmatrix}^T. \tag{26}$$

Once the model and the cost function is defined, next step is to get prediction equations and these are obtained from discrete state-space model of the plant.

4.3 Prediction with State-space Models

Prediction with a state-space model is straightforward, provided the nature of the model should be strictly proper. Proof for prediction equation is as follows: Consider the state-space model which gives the one step ahead prediction:

$$x_{k+1} = Ax_k + Bu_k; \quad y_{k+1} = Cx_{k+1}, \tag{27a}$$

and at instant $k+2$

$$x_{k+2} = Ax_{k+1} + Bu_{k+1}; \quad y_{k+2} = Cx_{k+2}. \tag{27b}$$

Substitute (22a) into (22b) to eliminate x_{k+1}, we get

$$x_{k+2} = A^2 x_k + ABu_k + Bu_{k+1}; \quad y_{k+2} = Cx_{k+2}, \tag{27c}$$

at instant $k+3$ using (22c)

$$x_{k+3} = A^2 x_{k+1} + ABu_{k+1} + Bu_{k+2}; \quad y_{k+3} = Cx_{k+3}. \tag{27d}$$

Substitute (22a) to eliminate x_{k+1}

$$x_{k+3} = A^2 [Ax_k + Bu_k] + ABu_{k+1} + Bu_{k+2}; \quad y_{k+3} = Cx_{k+3}. \tag{27e}$$

Hence a generalized equation for n-step ahead predictions will be

$$x_{k+n} = A^n x_k + A^{n-1} Bu_k + A^{n-2} Bu_{k+1} + \ldots + Bu_{k+n-1}, \tag{27f}$$

$$y_{k+n} = C[A^n x_k + A^{n-1} Bu_k + A^{n-2} Bu_{k+1} + \ldots + Bu_{k+n-1}]. \tag{27g}$$

Hence, one can form the whole vector of future predictions up to a horizon n_y as follows:

$$\begin{bmatrix} x_{k+1} \\ x_{k+2} \\ x_{k+3} \\ \vdots \\ x_{k+n_y} \end{bmatrix} = \begin{bmatrix} A \\ A^2 \\ A^3 \\ \vdots \\ A^{n_y} \end{bmatrix} x_k + H_x \begin{bmatrix} u_k \\ u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+n_y-1} \end{bmatrix}, \tag{28a}$$

$$\begin{bmatrix} y_{k+1} \\ y_{k+2} \\ y_{k+3} \\ \vdots \\ y_{k+n_y} \end{bmatrix} = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{n_y} \end{bmatrix} x_k + H \begin{bmatrix} u_k \\ u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+n_y-1} \end{bmatrix}, \tag{28b}$$

$$H_x = \begin{bmatrix} B & 0 & 0 & \ldots \\ AB & B & 0 & \ldots \\ A^2B & AB & B & \ldots \\ \vdots & \vdots & \vdots & \vdots \\ A^{n_y-1}B & A^{n_y-2}B & A^{n_y-3}B & \ldots \end{bmatrix}, \tag{28c}$$

$$H = \begin{bmatrix} CB & 0 & 0 & \ldots \\ CAB & CB & 0 & \ldots \\ CA^2B & CAB & CB & \ldots \\ \vdots & \vdots & \vdots & \vdots \\ CA^{n_y-1}B & CA^{n_y-2}B & CA^{n_y-3}B & \ldots \end{bmatrix}. \tag{28d}$$

4.4 Discrete State-space Model for FO Plants

In this paper, discrete state-space model is used in MPC for tracking problem. Discrete state-space model is obtained using Oustaloup recursive approximation. As discussed in the above section, the state-space model considered for prediction equations was of strictly proper nature. However, the state-space model obtained for five FO systems were not of strictly proper nature and hence, a pole was added at a far-away location in the left half of s-plane to make the system strictly proper in nature. A state-space model thus obtained was verified using frequency and step response (refer subsection 3.3) and it was observed that the addition of pole affects the frequency response at higher frequencies. Thus, for a specific region of frequency plot (from 10^0 to 10^4 rad/sec) the FO system retains its original form. Finally, obtained state-space model were discretized at a step size of 1 second. One example of discrete state-space model obtained using approximation method is shown subsequently.

4.4.1 Discrete State-space Model for $\frac{1}{s^{0.5}+1}$

Expression (14) was used to obtain the discrete state-space model (using MATLAB) with step size of one second. The obtained A,B,C and D matrices are as follows:

$$A = \begin{bmatrix} A1 & -6.934e^{34} \\ I & 0 \end{bmatrix},$$

where, I is 10X10 identity matrix and A1 is single row of 10 elements given by:

$$A1 = [-6.31e^{15}, -1.886e^{21}, -7.732e^{25}, -4.944e^{29}, -5.064e^{32}, -8.492e^{34}, -2.426e^{36}, -1.258e^{37}, -1.253e^{37}, -2.354e^{36}],$$

$$B = [1,0,0,0,0,0,0,0,0,0]^T,$$

$$C = [6.303e^{12}, 4.726e^{18}, 4.848e^{23}, 7.714e^{27}, 1.939e^{31}, 7.72e^{33}, 4.87e^{35}, 4.867e^{36}, 7.683e^{36}, 1.882e^{36}, 6.303^{e}34],$$

$$D = 0.$$

Similarly, discrete state-space models for other four FO plants were obtained. Once again the frequency and step response of obtained state-space model were compared with that of the original FOTF and it was observed that conversion of transfer function to state-space does not affect frequency and step response. Hence, use of state-space model (obtained through approximation) instead of FO system is further justified.

Once the model, cost function and the prediction equations are defined for MPC; the next step is to evaluate the control law.

4.5 Control Law

As stated above, the control law is determined from the minimization of a 2-norm measure of predicted performance (23). True optimal control uses $n_y = n_u = \infty$ in performance index (Rossiter, 2003); but there is a limitation on MPC algorithms and hence this assumption cannot be validated, making MPC suboptimal. Though an upper limit on n_y and n_u cannot be ∞ but the limits should be selected such that they are always higher than the settling time of the system. For large n_y and n_u MPC gives near identical control to an optimal control law with the same weights. While on the other hand, very small n_y and n_u will result in severely suboptimal control law (Rossiter, 2003)(Camacho and Bordons, 2004). Hence, proper choice of n_y and n_u is required for MPC. Selection of n_y and n_u will affect the control law, since they are calculated ones and used each time when the control law is updated. The control law for discrete state-space model and unconstrained condition is (Rossiter, 2003):

$$\Delta u_k = P_r \underset{\rightarrow}{r} - Kx - P_r[y_k - \hat{y}_k], \qquad (30)$$

where y_k is the process output (real output) and $\hat{y}_k$ is the model output. P_r and K are given by

$$\begin{aligned} P_r &= (H^T H + \lambda I)^{-1} H^T M, \\ K &= (H^T H + \lambda I)^{-1} H^T P, \\ P &= \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{n_y} \end{bmatrix}, \end{aligned}$$

and M is a weighting vector.

Expression (30) is used to evaluate optimal control effort u^* to be given to the plant/process and to the model. At each instant the optimal control effort u^* is updated by evaluating the control law (30) by considering revised process output y_k, model output $\hat{y}_k$ and the states x. Fig. (9) represents block diagram of MPC with FO plant and a model, obtained using approximation method.

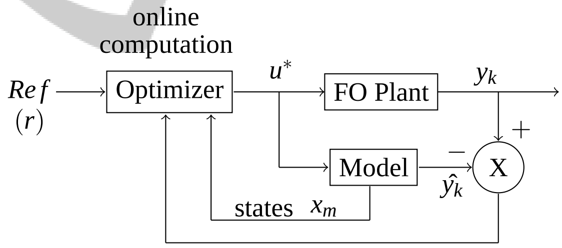

Figure 9: Block Diagram for MPC.

4.6 Process Output Evaluation

When MPC is applied to a real process, its output y_k is measured in real time at every instant for evaluation of optimal control effort. In case of simulations, generally the model of the process/plant is used to evaluate the process output. Hence, the bias $[y_k - \hat{y}_k]$ is redundant as y_k is equal to $\hat{y}_k$ at each instant. In this paper, the process output y_k has been evaluated by formulating an output equation in time domain for the input given at each instant to the FO plant mentioned in (21) for respective α values. Expression of y_k in time domain was obtained by applying inverse Laplace transformation ((Monje et al., 2010)) to the expression obtained using transfer function of FO plant (9) for respective α values. Laplace transform of Caputo fractional differentiation (1) and a special function known

as Mittag-Leffler function (MLF) is used to evaluate the expression of process output in time domain. For the evaluation of process output expression, definitions of 1-parameter MLF, 2-parameter MLF and Laplace transform of Caputo derivative is required and they are as follows:
Definition for Laplace transform of Caputo fractional differentiation is (Podlubny et al., 1997)

$$L[{}_0^C D_t^{\alpha} f(t)] = s^{\alpha}F(s) - \sum_{k=0}^{m-1} s^{\alpha-k-1}[\frac{d^k}{dt^k} f(t)]_{t=0}, \tag{31}$$

where $0 < \alpha < 1$, $m-1 < \alpha \leq m$ and $m \in \mathbb{N}$.
Definition for 1-parameter MLF is (Podlubny et al., 2002)

$$E_{\alpha} = \sum_{k=0}^{\infty} \frac{t^k}{\Gamma(\alpha k + 1)}, \tag{32}$$

where $\alpha \in \mathbb{R}^+$.
Definition for 2-parameter MLF is (Podlubny et al., 2002)

$$E_{\alpha,\beta} = \sum_{k=0}^{\infty} \frac{t^k}{\Gamma(\alpha k + \beta)}, \tag{33}$$

where $\alpha, \beta \in \mathbb{R}^+$.

Analytical expression for process output y_k is obtained by using inverse Laplace transformation on the expression by rewriting transfer function (9) in terms of $Y(s)$ and $U(s)$ using definition of Caputo fractional differentiation (1) and using Laplace transform of Caputo fractional differentiation (31) as follows:

$$\frac{Y(s)}{U(s)} = \frac{b}{s^{\alpha}+a}. \tag{34a}$$

Above expression is a general form of (9)

$$s^{\alpha}Y(s) + aY(s) = bU(s). \tag{34b}$$

The actual FDE with Caputo differentiation is

$${}_0^C D_t^{\alpha} y(t) + ay(t) = bu(t), \tag{34c}$$

for $0 < \alpha < 1$. Taking Laplace transform as given in (31) for Caputo differentiation, we get

$$s^{\alpha}Y(s) - Y(0) + aY(s) = bU(s), \tag{34d}$$

$$Y(s) = \frac{bU(s)}{s^{\alpha}+a} + \frac{Y(0)}{s^{\alpha}+a}. \tag{34e}$$

Taking inverse Laplace transform we get

$$L^{-1}\left(\frac{bU(s)}{s^{\alpha}+a}\right) = L^{-1}\left(\frac{1}{s^{\alpha}+a}\right) * L^{-1}\left(bU(s)\right). \tag{34f}$$

Note that u(t) is a constant value at any k^{th} instant and hence $U(s) = \frac{u_{k-1}}{s}$ whereas u_{k-1} is the constant value obtained as the optimal control effort at the previous instant. Using the standard inverse Laplace transform expressions as given in (Monje et al., 2010), (34f) becomes

$$L^{-1}\left(\frac{bU(s)}{s^{\alpha}+a}\right) = b \times u_{k-1}[1 - E_{\alpha}(-at^{\alpha})]. \tag{34g}$$

Again using (Monje et al., 2010) for Laplace inverse of second term

$$L^{-1}\left(\frac{Y(0)}{s^{\alpha}+a}\right) = y(o) \times [(t^{\alpha-1})E_{\alpha,\alpha}(-at^{\alpha})]. \tag{34h}$$

Hence, y_k is obtained for using (34g) and (34h) as:

$$y_k = b \times u_{k-1}[1 - E_{\alpha,1}(-at_k^{\alpha})] + y_{k-1}[t_k^{\alpha-1}E_{\alpha,\alpha}(-at_k^{\alpha})]. \tag{35}$$

Now, by substituting $a = 1$ and $b = 1$ in (35), an expression for y_k can be obtained for FO plant described by (9) and this expression takes into account intial/current conditions of plant.

$$y_k = u_{k-1}[1 - E_{\alpha,1}(-t_k^{\alpha})] + y_{k-1}[t_k^{\alpha-1}E_{\alpha,\alpha}(-t_k^{\alpha})] \tag{36}$$

In the following section, results obtained for simulation of unconstrained MPC with FO plant with different α values are shown, where $(0 < \alpha < 1)$.

5 RESULTS

In the previous sections of this paper, the control law and the process output has been evaluated which were used to obtain these results. MPC is applied to an FO plant defined at (21) for five different values of $\alpha = 0.1, 0.3, 0.5, 0,7$ and 0.9. Simulations are carried out to obtain output and input values for unconstrained condition. Output plot shows reference, process output y_k and model output $\hat{y_k}$ for respective FO plants. Subsequent plot shows the optimal control effort u^* at each instant for respective plants. For all plots y-axis represents amplitude in units and x-axis represents sampling instances. Sampling time of 1 second is used for these results. Though here results of simulations for one sampling time are shown, it was observed that the simulation of MPC for these FO plants is also possible for other step sizes. The range of step size for which simulations can be carried out was found to be from 0.5 to 1 seconds. This becomes one of the advantages of using continuous approximation methods for MPC of FO systems.

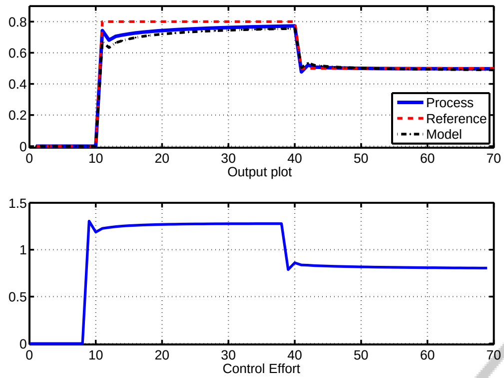

Figure 10: Output and control effort for $G_1(s) = \frac{1}{s^{0.1}+1}$.

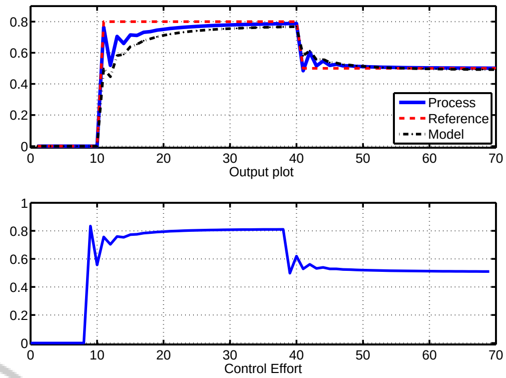

Figure 13: Output and control effort for $G_4(s) = \frac{1}{s^{0.7}+1}$.

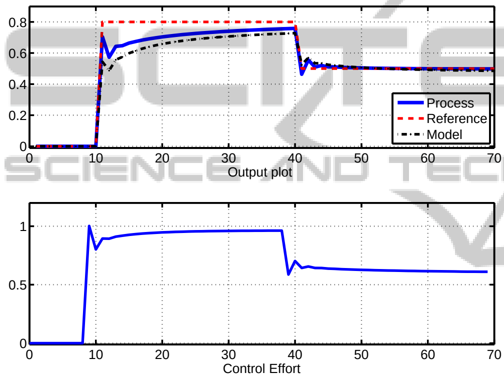

Figure 11: Output and control effort for $G_2(s) = \frac{1}{s^{0.3}+1}$.

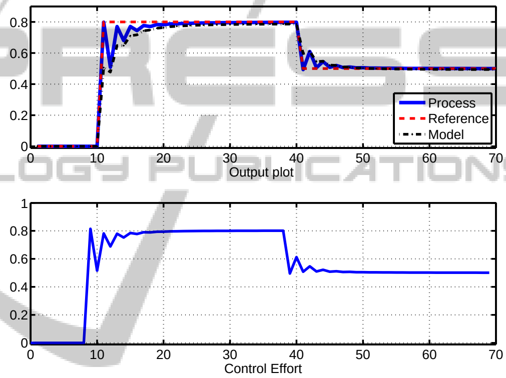

Figure 14: Output and control effort for $G_5(s) = \frac{1}{s^{0.9}+1}$.

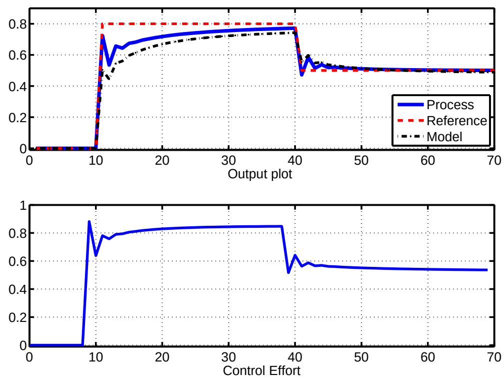

Figure 12: Output and control effort for $G_3(s) = \frac{1}{s^{0.5}+1}$.

Process output and model output are seen to have different values initially because both are calculated using different equations, though same initial conditions are fed to both the models . Process output is calculated using (36) with the help of MATLAB routine `mlf()` developed by Podlubny. Model output is calculated using state-space model using (27) obtained through Oustaloup recursive approximation. It was observed that the difference between process output and model output eventually becomes negligible. Results clearly show that the optimal control effort is obtained so that the process output can track the reference signal. In real scenario, there will be a actual plant running and on that MPC will be applied. It will be possible to measure real output at every instant and that will be fed to the optimizer along with the calculated output of internal model (see Fig. (9)). In most of the cases there will be a small amount of difference in the actual output and the model output at initial stages due to the errors in the model of the process. Here in simulations, the same scenario was created by using process output expression instead of using the same state-space model which was used as an internal model.

Few observations are noted from these results. Magnitude of control effort goes on reducing as the fractional-order α goes on increasing. Also, the process output becomes more oscillatory with increasing

α value, when the reference signal changes. There are two different reasons which may cause these anomalies. As stated earlier, control effort is obtained so that the process output can track the reference signal and hence process output expression is the reason behind this reduction. Process output expression uses two MLFs for its computation (see (36), (32) and (33)). MLF takes a long time for computation and the nature of MLF changes very rapidly for small α values. The other reasons behind irregularity can be the infinite memory nature of the FO system. For small α values a larger amount of past values are considered for computation.

Both these observations are not seen as soon as constraints are applied on the input as well as output. In this paper, unconstrained case was considered since for an optimal solution one needs to first confirm that solution exists for unconstrained case before going to constrained situation. In optimal theory, it is said that if the control law does not deliver optimal solution and robustness in the unconstrained case then it will not give optimal solution in the constrained case (Rossiter, 2003). Unfortunately the opposite does not hold true and hence it is worthwhile to check the performance of the system for unconstrained case first.

From these results it is evident that FO system's model can be used in its true form using approximation method for MPC. Furthermore, the obtained process and model output depicts the behaviour of the original FO system in terms of nature of plot and the required settling time. Systems with small values of α tend to take large time to settle. The reference trajectory is followed faithfully. The reference signal used for the simulation contains both positive and negative steps and hence it can be inferred that MPC for FO systems can work for all types of limiting reference signals.

6 CONCLUSION

Fractional-order systems are represented using FO model and for MPC, the model is required for prediction purposes. A discrete state-space model delivers straightforward prediction equations with one condition that the model should be of strictly proper nature. Control law was obtained for unconstrained MPC problem and was utilized to obtain results.

It was observed that MPC can be utilized for FO plants once the model of the FO plant is obtained and a unique analytical expression can be used to depict process output for FO plants. Approximation method was used to obtain model of the FO plant and its authenticity was verified using frequency and step response. Obtained results reveal that MPC can be used for FO plants and the model used works fine. Moreover, use of systematic expression obtained for process output makes it realistic. Hence, classical servo problem can be solved for a class of linear FO systems with the help of approximation method using MPC.

Use of other approximation methods like modified Oustaloup, Carlson will be interesting for MPC of FO systems. Furthermore, a comparison of these approximation methods for a single FO system might reveal some motivating results and hence the study of MPC with constraints for FO systems would be quite interesting and worth exploring as a future scope.

REFERENCES

Bemporad, A. (2006). Model predictive control design: New trends and tools. *IEEE Conference on Decision and Control*.

Boudjehem, D. and Boudjehem, B. (2012). A fractional model predictive control for fractional order systems. *Fractional Dynamics and Control Springer*, pages 59–71.

Camacho, E. and Bordons, C. (2004). *Model Predictive control*, volume 2. Springer.

Caponetto, R., Dongola, G., and Fortuna, L. *Fractional Order Systems Modeling and Control Applications*, volume 72. World Scientific Publishing Co. Pvt. Ltd., second edition.

Dzieliński, A. and Sierociuk, D. (2010). Fractional order model of beam heating process and its experimental verification. In *New trends in nanotechnology and fractional calculus applications*, pages 287–294. Springer.

Enacheanu, O., Riu, D., Retière, N., Enciu, P., et al. (2006). Identification of fractional order models for electrical networks. In *Proceedings of IEEE Industrial Electronics IECON 2006–32nd Annual Conference on*, pages 5392–5396.

Gabano, J.-D. and Poinot, T. (2011). Fractional modelling and identification of thermal systems. *Signal Processing*, 91(3):531–541.

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

Hortelano, M. R., de Madrid y Pablo, A. P., Hierro, C., and Berlinches, R. H. (2010). Generalized predictive control of arbitrary real order. *New trends in nanotechnology and fractional calculus applications, Springer*.

J.M.Maciejowski. *Predictive Control with Constraints*. Prentice Hall.

Mansouri, R., Bettayeb, M., and Djennoune, S. (2010). Approximation of high order integer systems by fractional order reduced-parameters models. *Mathematical and Computer Modelling*, 51(1):53–62.

Monje, C. A., Chen, Y. Q., Vinagre, B. M., Xue, D., and Feliu, V. (2010). *Fractional Order Systems and Control*. Springer.

Morari, M. and Lee, J. H. (1999). Model predictive control: past, present and future. *The Journal of Computers and Chemical Engineering*.

Muske, K. R. and Rawlings, J. B. (1993). Model predictive control with linear models. *Process Systems Engineering*, 39(2):262–287.

Nise, N. S. (2007). *Control Systems Engineering*. John Wiley & Sons.

Podlubny, I. (1998). *Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications*, volume 198. Academic press.

Podlubny, I., Dorcak, L., and Kostial, I. (1997). On fractional derivatives, fractional-order dynamic systems and pi$^{\lambda}$d$^{\mu}$-controllers. volume 5, pages 4985–4990. 36th IEEE Conference on Decision and Control.

Podlubny, I., Petras, I., Vinagre, B. M., O'leary, P., and Dorcak, L. (2002). Analogue realizations of fractional-order controllers. *J.Nonlinear Dynamics*, 29.

Poinot, T. and Trigeassou, J.-C. (2003). A method for modelling and simulation of fractional systems. *Signal processing*, 83(11):2319–2333.

Poinot, T. and Trigeassou, J.-C. (2004). Identification of fractional systems using an output-error technique. *Nonlinear Dynamics*, 38(1-4):133–154.

Rawlings, J. B. (2000). Tutorial overview of model predictive control. *IEEE Control Systems Magazine*.

Romero, M., de Madrid, A. P., Maoso, C., and Vinagre, B. M. (2010). Fractional order generalized predictive control: Formulation and some properties. *Int. Conf. Control, Automation, Robotics and Vision Singapore*.

Rossiter, J. A. (2003). *Model - Based Predictive Control: A Practicle Approach*. CRC Press.

Valerio, D. and Costa, J. (2011). Variable order fractional order derivatives and their numerical approximations. *J. Signal Processing*, 91.

Vinagre, B. M., Podlubny, I., Hernandez, A., and Feliu, V. (2000). Some approximation of fractional order operators used in control theory and applications. *Fractional calculus and applied analysis*, 3(3):210–213.

YaLi, H. and Ruikun, G. (2010). Application of fractional order model reference adaptive control on industry boiler burning system. *Int. Conf. on intelligent Computation Technology and Automation*.