

Numerical simulations for fractional variable-order equations [★]

Dorota Mozyrska ^{*} Piotr Oziabło ^{**}

^{*} Faculty of Computer Science, Białystok University of Technology, Wiejska 45A, Białystok, Poland (e-mail: d.mozyrska@pb.edu.pl)

^{**} Faculty of Computer Science, Białystok University of Technology, Wiejska 45A, Białystok, Poland (e-mail: p.oziabło@student.pb.edu.pl)

Abstract: The objective of the paper is to present the method of fitting finding constant λ coefficient and a parameter of an order function of the processes described by variable-, fractional-order backward difference of the Grünwald-Letnikov-type. As a qualitative criterion of the estimation the Coefficient of Determination (which we mark as R^2) and the Mean Square Error are used. All the numerical experiments were done with MATLAB.

Keywords: Difference equations, Dynamic systems, Eigenfunction, Fractional variable-order

1. INTRODUCTION

Fractional order calculus became an useful tool in modelling, which could be successfully used in both: continuous- and discrete-time systems. Models which involve fractional derivatives and fractional difference operators can very often describe real phenomena in more accurate way than it is possible with integer order models. The theory and applications of fractional calculus can be reviewed in (Hilfer, 2000; Kaczorek, 2009; Mozyrska & Wyrwas, 2015, 2017; Ostalczyk, 2016; Podlubny, 1999). For variable-order applications the reader can see more in Mozyrska & Ostalczyk (2017). In the paper we investigate discrete-time operators with variable-orders. One of the most important issues in modeling is finding a function which fits the experimental data, Garcia & Aguirre & Surez (2008). In the paper we deal with the problem of identifying parameters of a variable-order in discrete-time fractional equations by a variety of techniques, mainly based on MATLAB routines. Variable-order derivatives and differences are new directions in the research of fractional systems. Moreover, now there are research activities that are focused on developing new analysis and closed-loop system synthesis methods for fractional-order controllers being an extension of classical control theory. An appropriate choice of the order functions in fractional PID controllers is still one of the open problems.

In this work we calculate values of scalar eigenvalue functions for fractional variable-order initial value problem and then we are fitting back the noised eigenvalue function. The fitting process is based on assumption that the class of order function is known. There are considered functions of different class of monotonicity. Moreover, the constant λ coefficient of the linear equation with variable-, fractional-order operator and the parameter of order function should be estimated during fitting procedure. The idea of using particular MATLAB routine (lscurvefit) to estimate

fractional order parameters was inspired by Almeida & Bastos & Monteiro (2018). The paper is the first attempt to wider research of fitting values of fractional-, variable-order function. The method is planned to be used in fractional PID-like controllers. The important goal is to state what type of order functions can be considered as better in simulations of measurements. Moreover, the methods of fitting parameters of equations and order-functions are the core issue in possible modelling of controllers in closed-loop systems.

2. PRELIMINARIES

Definition 1. (Mozyrska & Ostalczyk (2017))

For $k, l \in \mathbb{N}$ and a given order function $\nu(\cdot)$ we define the oblivion function, as a discrete function of two variables, by its values $a^{[\nu(l)]}(k)$ given for $k > 0$ as

$$a^{[\nu(l)]}(k) = (-1)^k \frac{\nu(l) [\nu(l) - 1] \cdots [\nu(l) - k + 1]}{k!}, \quad (1)$$

and $a^{[\nu(l)]}(0) = 1$.

Formula (1) in Definition 1 is equivalent to the following recurrence with respect to $k \in \mathbb{N}$

$$a^{[\nu(l)]}(0) = 1, \\ a^{[\nu(l)]}(k) = a^{[\nu(l)]}(k-1) \left[1 - \frac{\nu(l) + 1}{k} \right] \text{ for } k \geq 1. \quad (2)$$

Definition 2. (Mozyrska & Ostalczyk (2017))

The Grünwald-Letnikov variable-, fractional-order backward difference (GL-VFOBD) with an order function $\nu : \mathbb{Z} \rightarrow \mathbb{R}_+ \cup \{0\}$ of function $x(\cdot)$ is defined as a finite sum

$$\begin{aligned} (\Delta^{[\nu(k)]}x)(k) &= \sum_{i=0}^k a^{[\nu(k)]}(i)x(k-i) \\ &= [1 \ a^{[\nu(k)]}(1) \ \cdots \ a^{[\nu(k)]}(k)] \begin{bmatrix} x(k) \\ x(k-1) \\ \cdots \\ x(1) \\ x(0) \end{bmatrix}. \end{aligned} \quad (3)$$

[★] The work was supported by Polish funds of National Science Center, granted on the basis of decision DEC-2016/23/B/ST7/03686.

In the paper we consider the basic form of linear equation with variable-order

$$\left(\Delta^{[\nu(k)]}x\right)(k) = \lambda x(k-1) + u(k-1), \quad k \geq 1 \quad (4)$$

with initial condition $x(0) = x_0$, constant coefficient $\lambda \in \mathbb{R}$. The function $u(\cdot)$ is given input signal. Then we can solve equation (4) by recurrence:

$$x(k) = - \sum_{i=1}^k a^{[\nu(k)]}(i)x(k-i) + u(k-1), \quad k \geq 1. \quad (5)$$

If in equation (4) we use constant order function equals zero, then we receive classical linear recurrence of first order. In other cases we include memory into the system.

Let us define the following matrix, see for example Mozyrska & Ostalczyk (2017)

$$\mathbf{A}^{[\nu(k)]} = \begin{bmatrix} 1 & a^{[\nu(k)]}(1) & \dots & a^{[\nu(k)]}(k) \\ 0 & 1 & \dots & a^{[\nu(k-1)]}(k-1) \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}. \quad (6)$$

Instead of working with recurrence, we can use a matrix form of defined matrices $\mathbf{A}^{[\nu(k)]}$. Moreover, let us use the following notation:

$$\mathbf{x}(k) = \begin{bmatrix} x(k) \\ x(k-1) \\ \vdots \\ x(1) \\ x(0) \end{bmatrix} \quad (7)$$

and similarly for input signal

$$\mathbf{u}(k) = \begin{bmatrix} u(k) \\ u(k-1) \\ \vdots \\ u(1) \\ u(0) \end{bmatrix}. \quad (8)$$

Then, equation (4) can be written in a matrix form

$$\mathbf{A}^{[\nu(k)]}\mathbf{x}(k) = \begin{bmatrix} \lambda \mathbf{x}(k-1) \\ x(0) \end{bmatrix} + \mathbf{u}(k), \quad k \geq 1$$

and it looks like the series of algebraic solutions

$$\mathbf{x}(k) = \left(\mathbf{A}^{[\nu(k)]}\right)^{-1} \left(\begin{bmatrix} \lambda \mathbf{x}(k-1) \\ x(0) \end{bmatrix} + \mathbf{u}(k) \right), \quad k \geq 1. \quad (9)$$

3. ANALYTICAL APPROXIMATION

During the first stage of our research we are trying to find constant coefficient λ in an analytical way. Our GL-VFOBD calculations are based on (9) matrix definition. We assume that the order function is known and because of that we also know the matrices given by formula (6). As a qualitative criterion of the fitting process we are using Mean Squared Error (which in our case would be a function of λ which we mark as $S_k(\lambda)$) and the Coefficient of Determination which we mark as R^2 .

We are doing numerical experiment by taking the values of exact solution to (9). Then, we add random values with mean equal to 0 and standard deviation equal to 20% of the mean of the given values of function $x(k)$. This simulates real time measurement data. In this case the simulated

data can be described as $\mathbf{y}_k = \mathbf{x}(k) + \epsilon$, where $\epsilon \sim N(0, \sigma)$ is added random value. The simplest criterion to fit λ would be in this case the minimisation of the following function

$$S_k(\lambda) := E_k^T E_k, \quad (10)$$

where

$$E_k = \mathbf{y}(k) - \left(\mathbf{A}^{[\nu(k)]}\right)^{-1} \left(\begin{bmatrix} \lambda \mathbf{y}(k-1) \\ y(0) \end{bmatrix} + \mathbf{u}(k) \right). \quad (11)$$

To simplify the formulation of minimiser we slightly changed formula (11), which enables easier calculations and (based on the experimental results) should not have a big impact on the final results. The updated formula which we use is the following

$$E_k = \mathbf{y}(k) - \lambda \left(\mathbf{A}^{[\nu(k)]}\right)^{-1} \begin{bmatrix} \mathbf{y}(k-1) \\ y(0) \end{bmatrix} + \left(\mathbf{A}^{[\nu(k)]}\right)^{-1} \mathbf{u}(k). \quad (12)$$

The similar situation has been considered in the paper, where we investigated approximations of λ for equations without $u(\cdot)$ function, in Oziablo (2018).

Let us use simplified E_k , then we can calculate the first derivative of $S_k(\lambda)$ in the following way

$$S'_k(\lambda) = E_k'^T E_k + E_k^T E_k'. \quad (13)$$

From that the critical point of $S_k(\lambda)$ is for $S'_k(\lambda) = 0$, we receive λ_0 which minimises S_k for the set of $k+1$ values of measurements. It is given by the formula

$$\lambda_0 = \frac{1}{2d} \left(\mathbf{y}_k^T A \begin{bmatrix} \mathbf{y}_{k-1} \\ y_0 \end{bmatrix} + \begin{bmatrix} \mathbf{y}_{k-1}^T y_0 \end{bmatrix} (A^T) \mathbf{y}_k - \begin{bmatrix} \mathbf{y}_{k-1}^T y_0 \end{bmatrix} A^T \mathbf{A} \mathbf{u}_k - \mathbf{u}_k^T A^T A \begin{bmatrix} \mathbf{y}_{k-1} \\ y_0 \end{bmatrix} \right), \quad (14)$$

where

$$d = \begin{bmatrix} \mathbf{y}_{k-1}^T y_0 \end{bmatrix} A^T A \begin{bmatrix} \mathbf{y}_{k-1} \\ y_0 \end{bmatrix} \quad (15)$$

and $A = \left(\mathbf{A}^{[\nu(k)]}\right)^{-1}$. We also use the abbreviations that $\mathbf{y}_k := \mathbf{y}(redk)$.

Then, theoretical values $\hat{\mathbf{y}}(k)$ we have from

$$\hat{\mathbf{y}}(k) = \left(\mathbf{A}^{[\nu(k)]}\right)^{-1} \left(\begin{bmatrix} \lambda_0 \mathbf{y}(k-1) \\ y(0) \end{bmatrix} + \mathbf{u}(k) \right). \quad (16)$$

In our research we used four different order functions: $\nu_1(k) = e^{-0.2k}$, $\nu_2(k) = 1 - e^{-0.2k}$, $\nu_3(k) = 1 - \frac{0.5}{k+1}$, $\nu_4(k) = \sin^2(0.5k)$. The results of analytical lambda calculations are shown in Figures 1, 2, 3, 4. The invocation results summary are shown in Table 1.

Order function	Estimated λ	R^2	$S_k(\lambda)$
$\nu_1(k)$	0.26897	0.95305	0.023426
$\nu_2(k)$	0.26458	0.65547	7.816600
$\nu_3(k)$	0.24310	0.37761	499.2796
$\nu_4(k)$	0.25784	0.83715	0.027874

Table 1. Analytical λ estimation results.

The analytical method of λ estimation returned the best result for the order function $\nu_1(k)$. Even when the estimated λ value 0.26897 significantly differed from the λ of

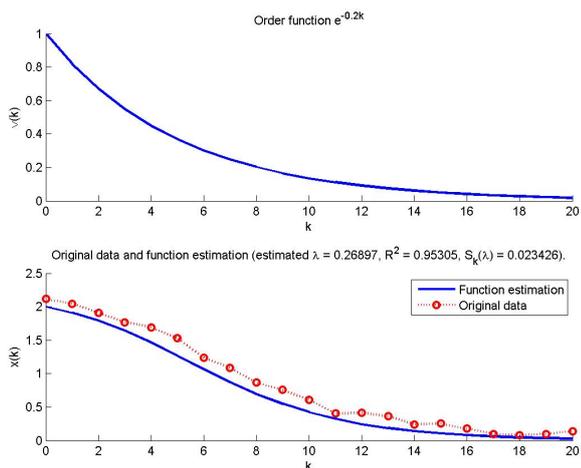


Fig. 1. Analytical estimation of λ for order function $\nu_1(k) = e^{-0.2k}$.

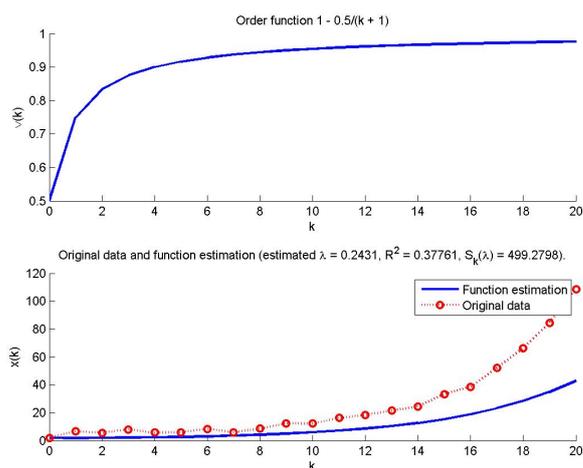


Fig. 3. Analytical estimation of λ for order function $\nu_3(k) = 1 - \frac{0.5}{k+1}$.

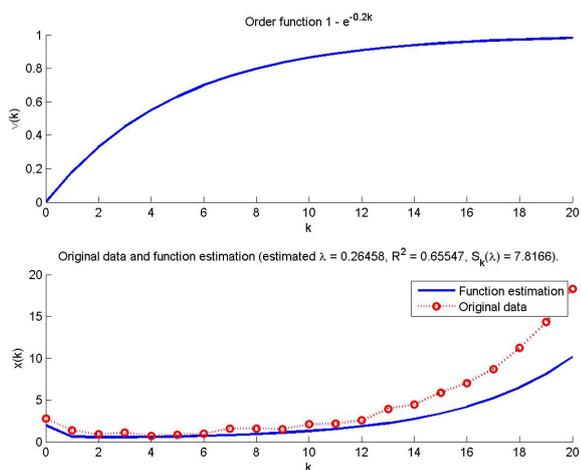


Fig. 2. Analytical estimation of λ for order function $\nu_2(k) = 1 - e^{-0.2k}$ (Levenberg-Marquardt).

the original, noised function (which was 0.3) Coefficient of Determination (R^2) higher than 0.9 proves that the function was properly fitted. Also for the order function $\nu_4(k)$ the fitting it gives a good result (R^2 equal to 0.83715), even though the estimated λ value is equal to 0.25784 (in comparison to original 0.3). For the order function $\nu_2(k)$ the values of R^2 is equal to 0.65547 and we get the satisfactory fit (R^2 value between 0,6 - 0,8 usually is considered as satisfactory). What is interesting, that the estimation error of λ was in this case lower than e.g. for $\nu_3(k)$ order function. The worst fitting result we get for order function $\nu_3(k)$. Then, R^2 equals to 0.37761, means that the fitting is not sufficient (R^2 value between 0 - 0.5 usually is considered as not sufficient).

4. APPROXIMATION OF UNKNOWN λ BY MATLAB ROUTINE

During the second part of our research we are trying to find a constant coefficient λ using MATLAB routines. We assume that the order function is known.

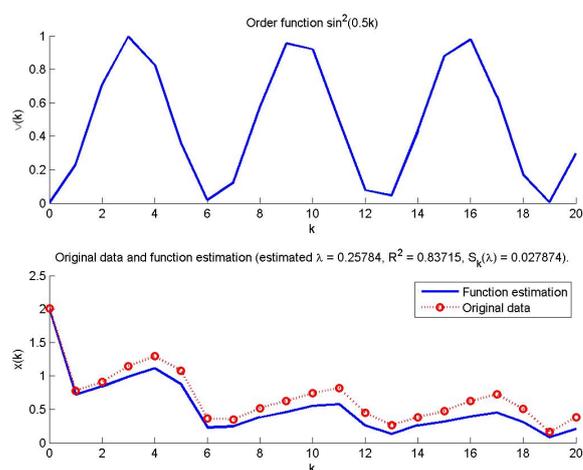


Fig. 4. Analytical estimation of λ for order function $\nu_4(k) = \sin^2(0.5k)$.

To fit our simulated experimental data with equation given by (9) we use lscurvefit MATLAB routine. The routine takes as an argument the function, which in our case calculates GL-VFOBD values, the set of experimental data and the initial value of searched parameter (in our case the initial value of λ). As the result the routine returns the parameter value for which the Mean Squared Error between function values (which calculates in our case GL-VFOBD) and experimental data is the lowest. The tests were done for the same set of four order functions as in the previous paragraph. Also the input data were exactly the same to make it easier to compare the results with the analytical solution described in the previous paragraph.

We checked the results for two different searching algorithms supported by lscurvefit routine which are Trust-Region-Reflective Least Squares and Levenberg-Marquardt Method. In all the cases both methods gave exactly the same results. The results of the invocations are shown in Figures 5, 6, 7, 8 (with marked Coefficient of Determination and Mean Squared Error values).

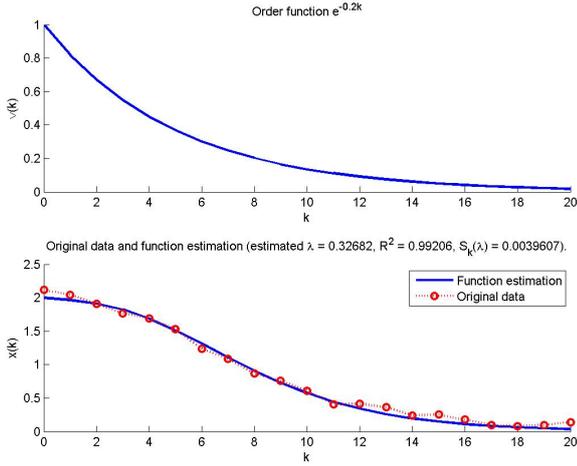


Fig. 5. Estimation (lsqcurvefit) of λ for order function $\nu_1(k) = e^{-0.2k}$.

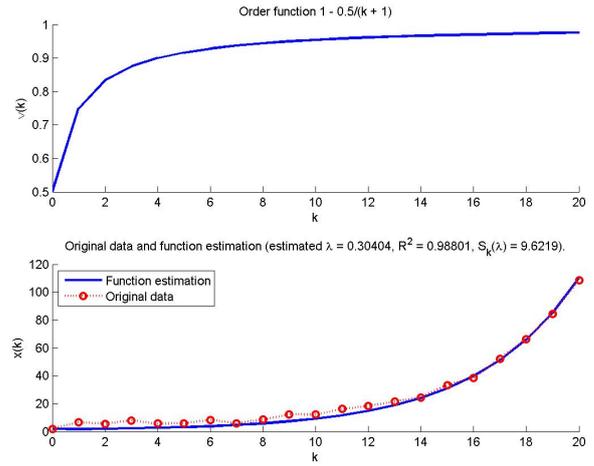


Fig. 7. Estimation (lsqcurvefit) of λ for order function $\nu_3(k) = 1 - \frac{0.5}{k+1}$.

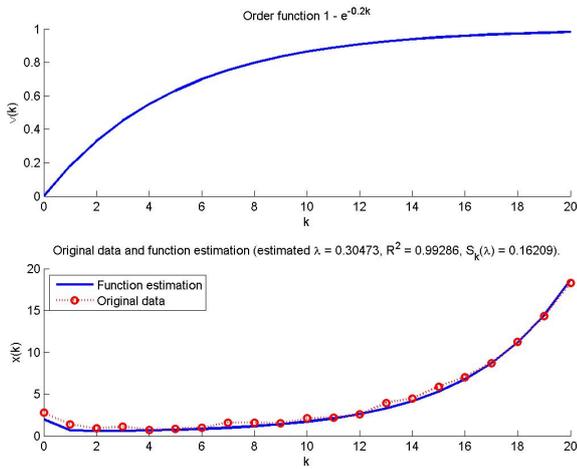


Fig. 6. Estimation (lsqcurvefit) of λ for order function $\nu_2(k) = 1 - e^{-0.2k}$ (Levenberg-Marquardt).

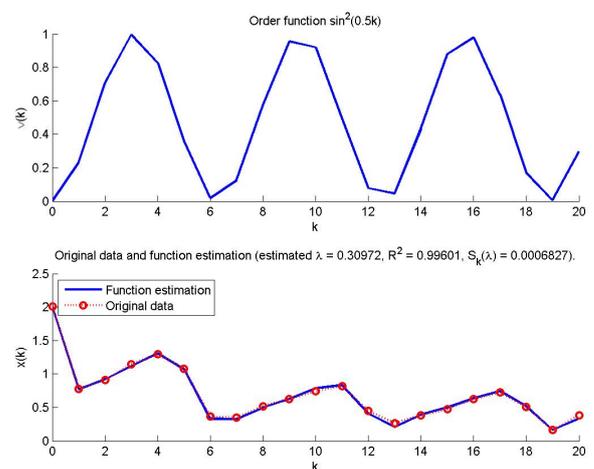


Fig. 8. Estimation (lsqcurvefit) of λ for order function $\nu_4(k) = \sin^2(0.5k)$.

The invocation results summary is shown in Table 2.

Order function	Estimated λ	R^2	$S_k(\lambda)$
$\nu_1(k)$	0.32682	0.99206	0.0039607
$\nu_2(k)$	0.30473	0.99286	0.1620900
$\nu_3(k)$	0.30404	0.98801	9.6219000
$\nu_4(k)$	0.30972	0.99601	0.0006827

Table 2. λ estimation with lsqcurvefit routine.

Using lscurvefit MATLAB routine we received very good results. For all the order functions Coefficient of Determination (R^2) significantly exceeded 0.9 (in most of the cases it was higher than 0.99) which means very good fitting. Also estimated λ values were in most of the cases very close to the original λ value of the noised signal. The lscurvefit MATLAB routine returned the best result for the order function $\nu_4(k)$ (R^2 equal to 0.99601). What is interesting is that while the Coefficient of Determination was the highest for $\nu_4(k)$, the estimation error of λ was the lowest for $\nu_3(k)$. At the same time for the order function $\nu_3(k)$ the value of R^2 was the lowest one (but still significantly above 0.9),

lower even than for the order function $\nu_1(k)$ for which the estimation error was relatively high λ . The possible reason of getting such results is the noise added to the original function (with $\lambda = 0.3$) which may increase the calculated Coefficient of Determination.

5. APPROXIMATION OF UNKNOWN λ AND ORDER FUNCTION VALUES

In the last step of our research we tried to fit the constant coefficient λ and the parameter p of the order function assuming that the general class of the order function is known. For the tests we took given in previous section formulas of the order functions (where p is searched parameter). As in the previous steps we used lscurvefit, but in this case the routine was provided with the initial values of λ and p and as a result it returned the optimal values of these two parameters which minimize Mean Squared Error between experimental data and GL-VFOBD values. For the order functions $\nu_1(k)$ and $\nu_2(k)$ the original p parameter value was -0.2. For the order functions $\nu_3(k)$

and $\nu_4(k)$ as the original p parameter value we took 0.5. For lscurvefit routine in most of the cases both (Levenberg-Marquardt and Trust Region) algorithms gave the same results. Just for the order function $\nu_2(k) = 1 - e^{-0.2k}$ the results were different. The results of lscurvefit routine invocations are shown in Figures 9, 10, 11, 12, 13.

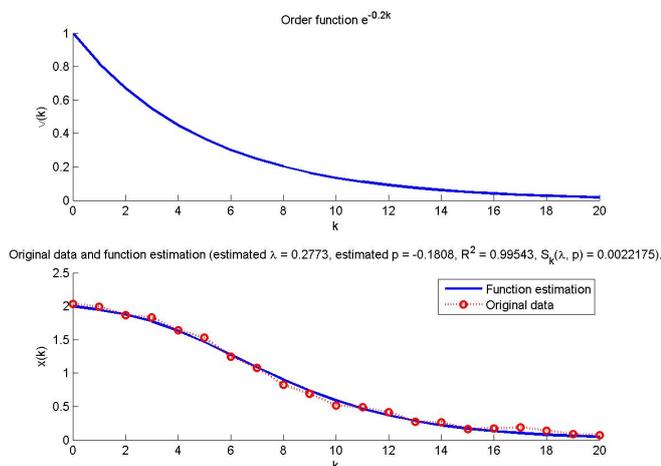


Fig. 9. Estimation (lscurvefit) of λ and p parameter for order function $\nu_1(k) = e^{-0.2k}$.

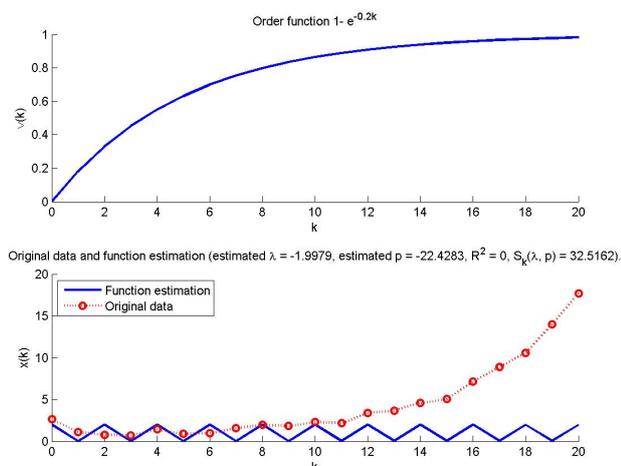


Fig. 10. Estimation (lscurvefit) of λ and p parameter for order function $\nu_2(k) = 1 - e^{-0.2k}$ (Levenberg-Marquardt).

The invocation results summary is shown in Table 3 (LM - Levenberg-Marquardt algorithm, TR - Trust Region algorithm). Using lscurvefit MATLAB routine to estimate

Order function	Est. λ	Est. p	R^2	$S_k(\lambda, p)$
$\nu_1(k)$	0.2773	-0.1808	0.9954	0.0022
$\nu_2(k)$ (LM)	-1.9979	-22.428	0	32.5162
$\nu_2(k)$ (TR)	0.2592	-0.2820	0.9943	0.1188
$\nu_3(k)$	0.2643	0.0336	0.9940	4.8367
$\nu_4(k)$	0.9535	0	0.7108	0.0516

Table 3. λ and p parameter estimation with lscurvefit routine.

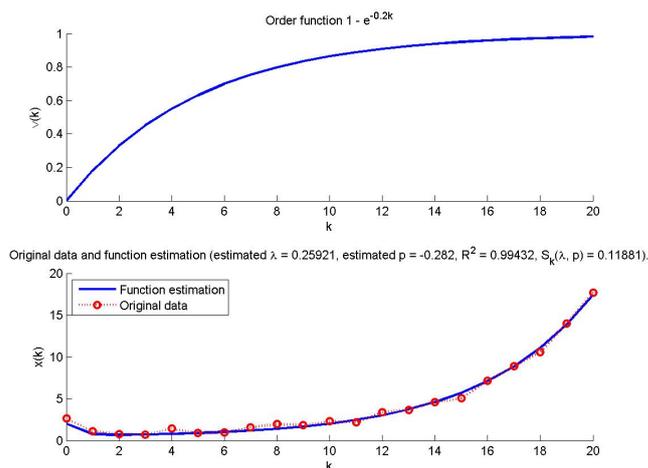


Fig. 11. Estimation (lscurvefit) of λ and p parameter for order function $\nu_2(k) = 1 - e^{-0.2k}$ (Trust Region).

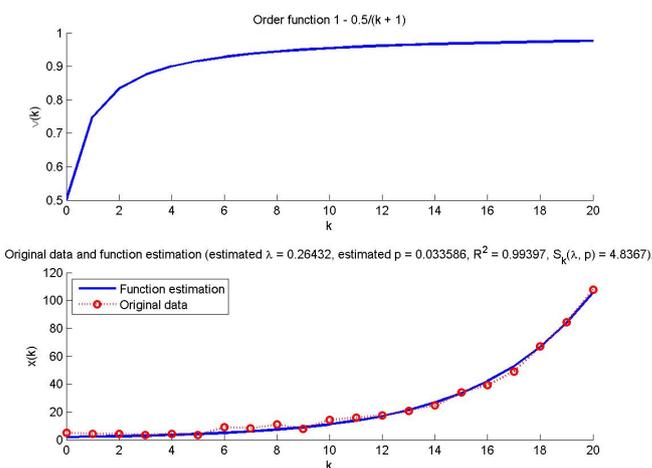


Fig. 12. Estimation (lscurvefit) of λ and p parameter for order function $\nu_3(k) = 1 - \frac{0.5}{k+1}$.

λ and p parameters gave very good results for order functions $\nu_1(k)$ and $\nu_3(k)$. In both cases we had Coefficient of Determination (R^2) higher than 0.99 which means a very good fitting. For the order function $\nu_2(k)$ lscurvefit routine returned good results only if we used Trust Region algorithm (R^2 higher than 0.99). Fitting did not work at all for order function $\nu_2(k)$ using lscurvefit routine with Levenberg-Marquardt algorithm (R^2 equal to 0). For order function $\nu_4(k)$ fitting results were not as good as for the rest of the functions but still one can be considered as satisfactory (R^2 value between 0.6 and 0.8). What is worth to be noticed, it is that estimated values of λ and p parameters were usually much different than the original parameter values, even while the Coefficient of Determination proved that the fitting process was correct and the fitting error was very low.

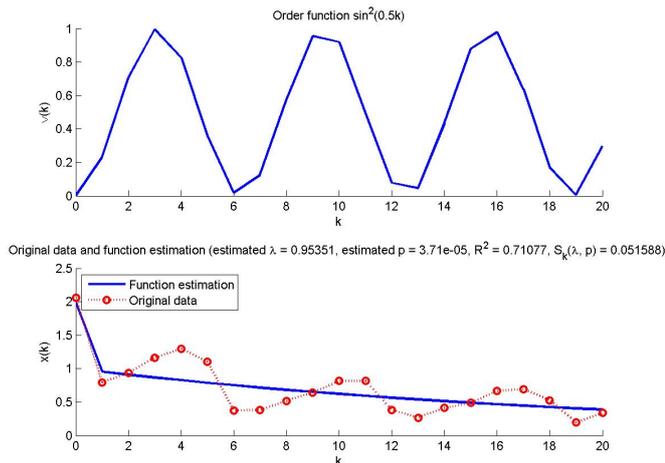


Fig. 13. Estimation (lsqcurvefit) of λ and p parameter for order function $\nu_4(k) = \sin^2(0.5k)$.

6. CONCLUSION

Both λ and parameter of order function can be successfully estimated using MATLAB. Function fitting for most of the examples was very high, taking as a qualitative criterion the Coefficient of Determination R^2 . Also presented analytical method to find λ gave good results in most of the examples, but in this cases the fitting quality depends on the order function.

REFERENCES

- Almeida, R. & Bastos, N.R.O. & Monteiro, M.T.T. (2018). A fractional Malthusian growth model with variable order using an optimization approach. *Published online in International Academic Press*.
- Garcia, R. & Aguirre, B. & Surez, R.. (2008). Stabilization of Linear Sampled-Data Systems by a Time-Delay Feedback Control. *Mathematical Problems in Engineering*. Vol. 2008, Art. ID 270518, 15 pages, ISSN: 1024-123X. DOI: 10.1155/2008/270518.
- Hilfer, R. (2000). *Applications of Fractional Calculus in Physics*. World Scientific Publishing Company, Singapore.
- Kaczorek, T. (2009). Fractional positive linear systems. *Kybernetes*, 38(7/8):1059-1078.
- D. Mozyrska and M. Wyrwas. Stability of discrete fractional linear systems with positive orders. *IFAC-PapersOnLine* 50 (1), 8115-8120 (2017).
- Mozyrska, D. & Wyrwas, M. (2015). The Z-transform method and delta type fractional difference operators. *Discrete Dynamics in Nature and Society*, 2015:25 pages.
- Mozyrska, D. & Ostalczyk, P. (2017). Generalized Fractional-Order Discrete-Time Integrator. *Complexity*, Volume 2017, Article ID 3452409, 11 pages.
- Ostalczyk, P. (2016). *Discrete fractional calculus. Applications in control and image processing*. World Scientific Publishing Co Pte Ltd, vol. Series in Computer Vision - Vol. 4.
- Oziabło, P. (2018). Numerical simulations for fractional variable-order difference eigenfunctions. *MATHMOD 2018 Extended Abstract Volume*, 9th Vienna Con-

ference on Mathematical Modelling, Vienna, Austria, February 21-23, 2018, p.85–86.
 Podlubny, I. (1999). *Fractional Differential Equations. Mathematics in Sciences and Engineering*. 198. Academic Press, San Diego.