FIR-FILTER DESIGN FOR DERIVATIVE ESTIMATION IN A NANOPOSITIONING SYSTEM

Johann Reger[†], *Arvid Amthor*[⋆], *Bernd Schmidt*[†]

[†]Control Engineering Group, Ilmenau University of Technology
[⋆]System Analysis Group, Ilmenau University of Technology

ABSTRACT

The paper is concerned with the real-time estimation of time derivatives with respect to signals subject to measurement noise. To this end, an algebraic counterpart of a weighted least squares algorithm is reformulated in order to find a tunable balance between proper noise attenuation and acceptable estimation delay. The estimator derived in this paper has the structure of an FIR filter whose coefficients may be calculated in an offline manner. Hence, the advocated model-free approach is well-apt for the derivative estimation under real-time conditions. The serviceability of the approach is demonstrated on a nanopositioning system where the carriage velocity needs to be reconstructed out of the noisy position signal.

***Index Terms*—** FIR filter design, algebraic derivative estimation, nanopositioning

1. INTRODUCTION

The accurate reconstruction of unknown system quantities, as for example system parameters and states, is substantially based on the disposability of appropriate process models. In case such models are available, there are numerous observer or optimization based approaches exploiting the structural properties of the process model in order to generate a set of linearly independent equations that admit to solve for the unknown system quantities in a computationally tractable way. Naturally, the real-time processability of these procedures is highly dependent on the numerical complexity of the respective systems of equations; see [8, 9, 13] for classical textbooks on the matter.

Whenever appropriate process models are not available, facilities are significantly limited in estimation. Under certain conditions, however, system quantities may be determined directly by filtering the measurement signals; for example, if some time-derivatives of a measurement signal are to be reconstructed. Filter design then has to be thoroughly tailored for the process, e.g. to match with the signals' noise properties. For real-time usage, also, abundant calculations should be avoided. Typical more recent solutions to the problem of derivative estimation are to employ discrete time differentiators [6], in particular of finite impulse response (FIR) type [12]. Since there is an inherent trade-off between noise attenuation and estimation delay, some of the algorithms allow for adjustment, see [14]. Numerical differentiators that are real-time capable may be obtained in various styles. Amongst the classical approaches, receding horizon estimators [4, 7] receive increasing attention due to lower memory requirements.

In this paper, we modify an estimation technique presented in [5, 10, 11] to obtain a receding horizon discrete FIR differentiator that is adjustable to a balance betweeen noise attenuation and estimation delay. The underlying application is a trajectory tracking control of a nanopositioning system, where the time-derivative of the noisy carriage position signal is required for the feedback control law.

The organization of the paper is as follows: After briefly recalling basics from algebraic time-derivative estimation in Section 2, Section 3 presents our main result on adjustable receding horizon FIR differentiation. Detailed steps of derivation are relegated to the Appendix. The experimental setup, the control scheme and the results that we obtained on the nanopositioning stage are presented in Section 4.

2. ALGEBRAIC FIR FILTER

For better readability, we recall an exposition of the algebraic estimation technqiue from [11]; for preliminary results also see the references therein.

Consider a real-valued, N-th order Taylor-polynomial function of time

$$y(t) = \sum_{i=0}^{N} \frac{y^{(i)}(0)}{i!}\, t^i \tag{1}$$

with unknown constant coefficients $y^{(i)}(0)$. The goal is to obtain estimates of the time-derivatives of $y(t)$ in terms of $y(t)$ which may be assumed available through measurement, for example.

In order to determine the coefficients $y^{(i)}(0)$, we could premultiply (1) by powers of t and integrate by parts. So as to ease notation we take an alternative: We rephrase (1) in the Laplace domain, i.e.

$$Y(s) = \sum_{i=0}^{N} \frac{y^{(i)}(0)}{s^{i+1}} \tag{2}$$

which is equivalent to

$$s^{N+1}\,Y(s) = \sum_{i=0}^{N} y^{(i)}(0)\,s^{N-i}\,. \qquad (3)$$

Due to the simple polynomial form of (3) we may differentiate the equation with respect to the operator s so as to isolate the term $y^{(i)}(0)$. This idea leads to

$$\frac{d^j}{ds^j}\left(\frac{1}{s}\frac{d^{N-j}}{ds^{N-j}}\left(s^{N+1}Y(s)\right)\right) = \frac{(-1)^j j!\,(N-j)!}{s^{j+1}}y^{(j)}(0). \qquad (4)$$

In view of the present $N+1$ fold time differentiation of $y(t)$, the equation is integrated $N+1$ times, hence premultiplied by $\frac{1}{s^{N+1}}$ as per

$$\frac{1}{s^{N+1}}\frac{d^j}{ds^j}\left(\frac{1}{s}\frac{d^{N-j}}{ds^{N-j}}\left(s^{N+1}Y(s)\right)\right) = \frac{(-1)^j j!\,(N-j)!}{s^{N+j+2}}y^{(j)}(0). \qquad (5)$$

Following the steps in [11], equation (5) can be transformed back into the time domain. The result reads:

Theorem 1 *Let $y(t)$ be a polynomial signal of time as in (1) and $y^{(j)}(t)$ its j-th order time derivative. For any $T > 0$ and $t \geq T$, the derivative $y^{(j)}(t)$, $j = 0, 1, \ldots, N$, satisfies the convolution*

$$\hat{y}^{(j)}(t) = \int_0^T H_j(T,\tau)\,y(t-\tau)\,d\tau \qquad (6)$$

where

$$H_j(T,\tau) = \frac{(N+j+1)!\,(N+1)!}{T^{N+j+1}}\times \sum_{\kappa=0}^{N-j}\sum_{\nu=0}^{j}\frac{(T-\tau)^{\kappa+\nu}\,(-\tau)^{N-\kappa-\nu}}{\kappa!\,\nu!\,(N-j-\kappa)!\,(j-\nu)!\,(N-\kappa-\nu)!\,(\kappa+\nu)!\,(N-\kappa+1)}\,.$$

There are several remarks in order here: The theorem clearly states that the time derivatives of the polynomial function y are determined in an exact way after an arbitrary small amount of time T. Consequently, the presented approach exhibits a deadbeat property that, when discretizing the integral, yields a finite impulse response filter (FIR filter) scheme as shown in [15]. Of course, for non-polynomial but sufficiently smooth signals z, the function y may be considered as its Taylor polynomial approximation. In this case, the theorem yields an estimate for the time derivatives of z, locally on $[t-T, t]$. Finally, the calculation of the time derivatives is based on the values $y(t)$ on the fixed width but receding time horizon $[t-T, t]$, only. Therefore, the approach is memoryless in the sence that perturbations on measurements of $y(t)$ that occur outside the horizon do not interfer with the accuracy of the estimate.

3. ALGEBRAIC LOW PASS FILTERING

In order to introduce an adjustable low pass filter (instead of taking integrators, only) the central equation (4) may also be premultiplied by some first order low pass filter series, where the cut-off frequency s_1 may be specified arbitrarily and the entire filter order may be varied by means of ν. Thus, for $j = 0, 1, \ldots, N$

$$\underbrace{\left(\frac{1}{s_1+s}\right)^{N+\nu+1}\frac{d^j}{ds^j}\left(\frac{1}{s}\frac{d^{N-j}}{ds^{N-j}}\left(s^{N+1}Y(s)\right)\right)}_{=:\,\mathrm{lhs}} = \underbrace{\frac{(-1)^j\,j!\,(N-j)!}{(s_1+s)^{N+\nu+1}\,s^{j+1}}}_{=:\,\mathrm{rhs}}\,y^{(j)}(t)\,. \qquad (7)$$

Further manipulation on lhs and rhs as is done in the Appendix, i.e. introducing a fixed receding horizon time window $T > 0$, allowing for causality and using a zero-order hold for discretization, leads to the filter representation formula

$$\sum_{k=1}^{M} y(t-\tfrac{k}{M}T)\,\overline{\mathrm{lhs}} = y^{(j)}(t)\,\overline{\mathrm{rhs}}\,. \qquad (8)$$

which results from equating lhs and rhs with $\overline{\mathrm{lhs}}$, $\overline{\mathrm{rhs}}$ from (17),(21). Now, we may define

$$\hat{\Pi}_{j,k} = \frac{\overline{\mathrm{lhs}}}{\overline{\mathrm{rhs}}}. \qquad (9)$$

For simplicity, we use a time window T that is a multiple of the sampling time T_S, i.e. $T = M\,T_\mathrm{S}$, Finally, the estimate of the j-th time-derivative of $y(t)$ results in

$$y^{(j)}(t) = \sum_{k=1}^{M} y(t-kT_\mathrm{s})\,\hat{\Pi}_{j,k}\,. \qquad (10)$$

In the following, we abbreviate the estimator by ADES (= Algebraic Derivative Estimator Scheme).

For characterization of the estimator, consider the Bode-plot of the first derivative in Fig. 1. As can be seen, it is possible to adjust the cutoff frequency s_1 so as to alter the noise attenuation of the filter.

4. NANOPOSITIONING SYSTEM

4.1. Experimental setup

The experimental setup is a two dimensional fine positioning stage (see Fig. 2). It was developed at the Collaborative Research Centre “Nanopositioning and Nanomeasuring Machines” at Ilmenau University of Technology [1]. As illustrated in Fig. 2, every axis is driven by two linear voice coil actuators of IDAM[1]. The motors are powered by proprietary developed analog amplifiers, which provide the needed current with the required precision. The control algorithm uses magnetic field intensity measurements provided online by integrated Hall-sensors in order to commutate the motors. The operating range of this positioning stage is

[1] http://www.ina-dam.de

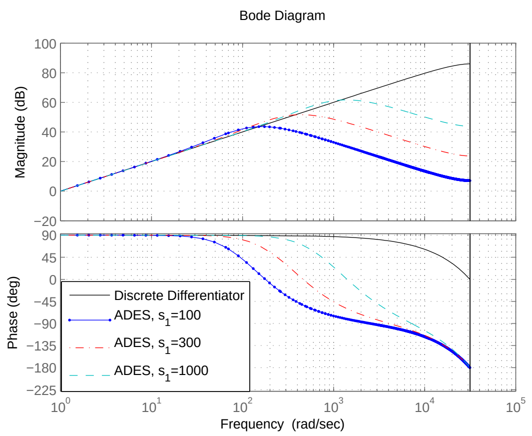

Fig. 1. Bode plot of ADES with $N = 2, M = 1001, \nu = 0, T_S = 10^{-4}$sec for several values of s_1 for $j = 1$ (first time-derivative), and comparison with an ideal discrete differentiator $G(z) = 10^4 \frac{z-1}{z}$.

200×200 mm^2. Each axis is supported by two linear guide ways of SCHNEEBERGER[2]. The position is measured by a stabilized NeHe-laser interferometer of type SP 2000 (manufactured by SIOS Messtechnik GmbH[3]) with a resolution of less than 0.1 nm. For data acquisition and control a modular dSpace® [4] real-time hardware system in combination with Matlab/ Simulink® is utilized. The position is provided by the SIOS interferometer unit as a 32-bit digital signal and is sampled by the dSpace system at a rate of 10 kHz (i.e. $T_S = 10^{-4}$sec). Also the control algorithm uses this sampling rate and operates on the analog amplifiers with a 16 bit resolution. For the presented study only the outer axis of the demonstrator is used. The inner axis is mechanically jammed at the position shown in Fig. 2.

[2] http://www.schneeberger.com
[3] http://www.SIOS.de
[4] http://www.dspace.de

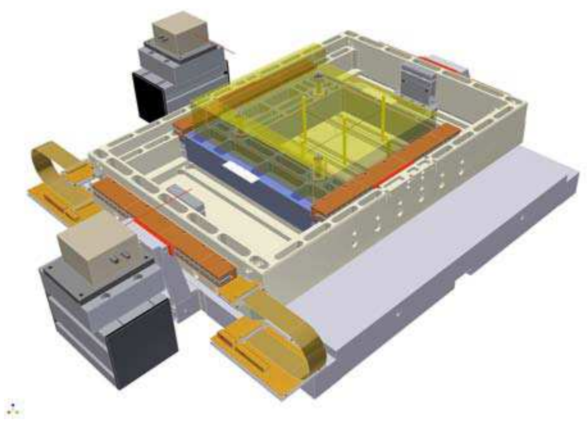

Fig. 2. xy-fine positioning stage

4.2. Control Scheme

The mentioned trajectory tracking control scheme is shown in Fig. 3. As can be seen, this approach is composed of the components trajectory generation, feed-forward friction compensation, feedback controller, disturbance observer and state reconstruction. The dynamic set points for the position and its derivatives are generated by a trajectory generation algorithm. The friction compensation is realized by an adaptive inverse system model in the feed-forward path of the trajectory tracking controller [1]. The disturbances caused by sound waves, ground motion, etc. are compensated by a disturbance observer [2]. A PI state space controller in the feedback path accounts for model uncertainties between the adaptive feed-forward friction compensator and the real system [3]. Due to the fact that the immeasurable velocity is needed as input for the feedback controller, an elaborated derivation algorithm is needed in order to reconstruct the velocity out of the noisy position signal.

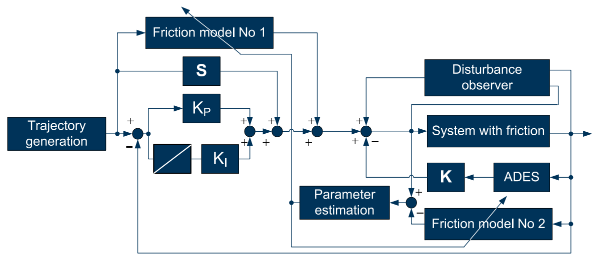

Fig. 3. Trajectory tracking control scheme

4.3. Results

The experimental task is to track a sinusoidal reference signal with amplitude 10^4 nm and frequency 1 Hz, i.e. $r(t) = 10^4 \sin(2\pi t)$ for the carrier position, and the expected carrier velocity would be its derivative $v(t) = \dot{r}(t) = 2\pi 10^4 \cos(2\pi t)$.

Extensive experiments showed that the system behavior of the considered nanopositioning machine is dominated by friction [1]. We refrained from modeling the dynamic friction behavior on nanometer scale, for complexity reasons, with the consequence that the carrier velocity could not be reconstructed by means of an observer. Instead, we used ADES to calculate the carrier velocity of the nanopositioning system from the noisy position data itself, i.e. estimate its first derivative. The values for ADES were $N = 2$, $M = 1001$, $\nu = 0$, $T_S = 10^{-4}$sec, $s_1 = 800$. For comparison of the result, we considered the optimal Savitzky-Golay filter with $N = 2$, $M = 1001$, $T_S = 10^{-4}$sec. Savitzky-Golay filters (as most standard filters) are not suitable for real-time estimation and have a delay of half a window length for its first derivative. As is well-known, there is a trade-off between estimation delay and noise attenuation. Fig. 4 depicts this trade-off. ADES shows less delay but more noise (dependent on the cutoff frequency s_1), while the Savitzky-Golay filter has more delay but less noise.

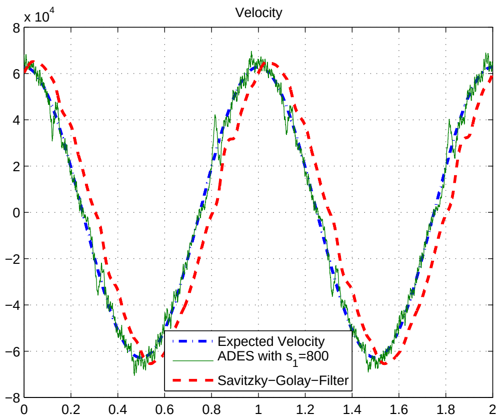

Fig. 4. Expected velocity, velocity estimate from ADES for $N = 2$, $M = 1001$, $\nu = 0$, $T_S = 10^{-4}$sec, $s_1 = 800$ and velocity estimate from Savitzky-Golay filter with $N = 2$, $M = 1001$, $T_S = 10^{-4}$sec.

5. CONCLUSIONS

In this contribution, an algebraic derivative estimation technique was extended to better adapt to measurement noise and estimation delay. The experimental validity of the algorithm was demonstrated on a nanopositioning system where the carrier velocity was estimated out of noisy position data. In contrast to model-based observers and other standard estimation techniques which turned out unsuitable, the incorporation of ADES ensured to meet the control objective.

6. ACKNOWLEDGEMENTS

The work of AA was done within the Collaborative Research Centre "Nanopositioning and Nanomeasuring Machines" at Ilmenau University of Technology, which is supported by the German Research Foundation (DFG) and the Thuringian Ministry of Science.

7. REFERENCES

[1] A. Amthor, S. Zschäck, and C. Ament, "High precision position control using an adaptive friction compensation approach," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 274–278, 2010.

[2] A. Amthor, S. Zschäck, and C. Ament, "Position Control on Nanometer Scale based on an Adaptive Friction Compensation Scheme," *Annual Conference of the IEEE Industrial Electronics Society*, Orlando, USA, pp. 2568–2573, 2008.

[3] M. Müller, A. Amthor, W. Fengler, and C. Ament, "Model-driven Development and Multi-processor Implementation of a Dynamic Control Algorithm for Nanopositioning and Nanomeasuring Machines," *Journal of Systems and Control Engineering*, vol. 223, no. 3, pp. 417–429, 2009.

[4] S. Fuksa and W. Byrski, "General approach to linear optimal estimator of finite number of parameters," *IEEE Transactions on Automatic Control*, vol. 29, no. 5, pp. 470–472, 1984.

[5] M. Fliess and H. J. Sira-Ramírez, "State reconstructors: a possible alternative to asymptotic observers and Kalman filters," *Proc. of CESA*, 2003.

[6] I. R. Khan and R. Ohba, "New design of full band differentiators based on Taylor series," *IEE Proceedings Vision, Image & Signal Processing*, vol. 146, pp. 185–189, 1999.

[7] W. H. Kwon, P. S. Kim and P. G. Park, "A receding horizon Kalman FIR filter for linear continuous-time systems," *IEEE Transactions on Automatic Control*, vol. 44, no. 11, pp. 2115–2120, 1995.

[8] D. Luenberger, *Optimization by Vector Space Methods*, Wiley, 1969.

[9] J. O'Reilly, *Observers for Linear Systems*, Academic Press, 1983.

[10] J. Reger and J. Jouffroy, "Algebraische Ableitungsschtzung im Kontext der Rekonstruierbarkeit," [in German], *at-Automatisierungstechnik*, vol. 56, no. 6, pp. 324–331, 2008.

[11] J. Reger and J. Jouffroy, "On algebraic time-derivative estimation and deadbeat state reconstruction," *IEEE Conf. on Decision and Control*, Shanghai, China, pp. 1740–1745, 2009.

[12] S. Samadi and A. Nishihara, "Optimal Control and Estimation," *IEICE Transactions*, vol. E90-A, no. 8, pp. 1511–1518, 2007. Springer, 1998.

[13] R. Stengel, *Optimal Control and Estimation*, Dover Publications, 1986.

[14] S. Valiviita and O. Vainio, "Delayless differentiation algorithm and its efficient implementation for motion control applications," *IEEE Transactions on Instrumentation and Measurement*, vol. 48, no. 5, pp. 967–971, 1999.

[15] J. Zehetner, J. Reger, and M. Horn, "Echtzeit-Implementierung eines algebraischen Ableitungsschtzverfahrens, " [in German], *at-Automatisierungstechnik*, vol. 55, no. 11, pp. 553–560, 2007.

Appendix

Taking the left hand side (lhs) of (7) and using the general Leibniz rule, one my show that

$$\begin{aligned}\mathrm{lhs} &= \left(\frac{1}{s_1+s}\right)^{N+\nu+1}\left(\frac{d^j}{ds^j}\left(\frac{1}{s}\frac{d^{N-j}}{ds^{N-j}}\left(s^{N+1}Y(s)\right)\right)\right)\\ &= \sum_{\kappa_1=0}^{N-j}\sum_{\kappa_2=0}^{j}\binom{N-j}{\kappa_1}\binom{j}{\kappa_2}\frac{(N+1)!}{(N-\kappa_1-\kappa_2)!\,(N-\kappa_1+1)}\frac{s^{N-\kappa_1-\kappa_2}}{(s_1+s)^{N+\nu+1}}\frac{d^{N-\kappa_1-\kappa_2}}{ds^{N-\kappa_1-\kappa_2}}Y(s)\,.\end{aligned}$$

Use the correspondence

$$\frac{s^{N-\kappa_1-\kappa_2}}{(s_1+s)^{N+\nu+1}}\frac{d^{N-\kappa_1-\kappa_2}}{ds^{N-\kappa_1-\kappa_2}}Y(s) \bullet\!\!-\!\!\circ \int_0^{t_1}\frac{d^{N-\kappa_1-\kappa_2}}{dt^{N-\kappa_1-\kappa_2}}\left(\frac{t^{N+\nu}}{(N+\nu)!}e^{-s_1 t}\right)\Bigg|_{t_1\to t_1-\tau}(-\tau)^{N-\kappa_1-\kappa_2}\,y(t-\tau)\,d\tau \tag{11}$$

set $t_1=-T$ with fixed time window $T>0$ and flip $\tau\to-\tau$, for causality, such that (11) is equivalent to

$$\begin{aligned}\int_0^{T}\frac{d^{N-\kappa_1-\kappa_2}}{dt^{N-\kappa_1-\kappa_2}}\left(\frac{t^{N+\nu}}{(N+\nu)!}e^{-s_1 t}\right)\Bigg|_{t=\tau-T}(-1)\,\tau^{N-\kappa_1-\kappa_2}y(-\tau)\,d\tau &= \sum_{\kappa_3=0}^{N-\kappa_1-\kappa_2}\binom{N-\kappa_1-\kappa_2}{\kappa_3}\times\\ \frac{(-1)^{\nu+\kappa_1+\kappa_2+1}s_1^{\kappa_3}}{(\nu+\kappa_1+\kappa_2+\kappa_3)!}\int_0^{T}e^{s_1(T-\tau)}\,(T-\tau)^{\nu+\kappa_1+\kappa_2+\kappa_3}\tau^{N-\kappa_1-\kappa_2}\,y(t-\tau)\,d\tau\,. \end{aligned}\tag{12}$$

Then, substitute $\tau=\mu T$ and use the binomial formula. In addition, assume that the y-values are sampled, say with M sampling intervals in the time window T. On the assumption of a zero order hold we may write

$$\begin{aligned}&\int_0^{T}e^{s_1(T-\tau)}\,(T-\tau)^{\nu+\kappa_1+\kappa_2+\kappa_3}\tau^{N-\kappa_1-\kappa_2}\,y(-\tau)\,d\tau\\ &= \sum_{\kappa_4=0}^{\nu+\kappa_1+\kappa_2+\kappa_3}\binom{\nu+\kappa_1+\kappa_2+\kappa_3}{\kappa_4}(-1)^{\kappa_4}\,T^{N+1+\nu+\kappa_3}\int_0^{1}\mu^{N-\kappa_1-\kappa_2+\kappa_4}e^{(1-\mu)s_1T}\,y(t-\mu T)\,d\mu\\ &= \sum_{\kappa_4=0}^{\nu+\kappa_1+\kappa_2+\kappa_3}\sum_{k=1}^{M}\binom{\nu+\kappa_1+\kappa_2+\kappa_3}{\kappa_4}(-1)^{\kappa_4}\,T^{N+1+\nu+\kappa_3}y(t-\tfrac{k}{M}T)\int_{\frac{k-1}{M}}^{\frac{k}{M}}\mu^{N-\kappa_1-\kappa_2+\kappa_4}e^{(1-\mu)s_1T}\,d\mu\,.\end{aligned}\tag{13}$$

We use the integral

$$\int_{t_1}^{t_2}t^n\,e^{a\,t}dt=\left[e^{a\,t}\sum_{i=0}^{n}\frac{(-1)^i}{a^{i+1}}\frac{n!}{(n-i)!}\,t^{n-i}\right]_{t=t_1}^{t=t_2} \tag{14}$$

in order to calculate

$$\begin{aligned}\int_{\frac{k-1}{M}}^{\frac{k}{M}}\mu^{N-\kappa_1-\kappa_2+\kappa_4}e^{(1-\mu)s_1T}\,d\mu &= e^{s_1T}\int_{\frac{k-1}{M}}^{\frac{k}{M}}\mu^{N-\kappa_1-\kappa_2+\kappa_4}e^{-s_1T\mu}\,d\mu\\ &= \sum_{\kappa_5=0}^{N-\kappa_1-\kappa_2+\kappa_4}\frac{(-1)^{\kappa_5}}{(-s_1T)^{\kappa_5+1}}\frac{(N-\kappa_1-\kappa_2+\kappa_4)!}{(N-\kappa_1-\kappa_2+\kappa_4-\kappa_5)!}\times\\ &\left(\left(\tfrac{k}{M}\right)^{N-\kappa_1-\kappa_2+\kappa_4-\kappa_5}e^{\left(1-\frac{k}{M}\right)s_1T}-\left(\tfrac{k-1}{M}\right)^{N-\kappa_1-\kappa_2+\kappa_4-\kappa_5}e^{\left(1-\frac{k-1}{M}\right)s_1T}\right)\,.\end{aligned}\tag{15}$$

Collecting these results, the left hand side of equation (7), at arbitrary time instants t, reads

$$\mathrm{lhs}=\sum_{k=1}^{M}y(t-\tfrac{k}{M}T)\,\overline{\mathrm{lhs}} \tag{16}$$

with

$$\overline{lhs} = \sum_{\kappa_1=0}^{N-j} \sum_{\kappa_2=0}^{j} \sum_{\kappa_3=0}^{N-\kappa_1-\kappa_2} \sum_{\kappa_4=0}^{\nu+\kappa_1+\kappa_2+\kappa_3} \sum_{\kappa_5=0}^{N-\kappa_1-\kappa_2+\kappa_4} \binom{N-j}{\kappa_1}\binom{j}{\kappa_2}\binom{N-\kappa_1-\kappa_2}{\kappa_3} \times$$
$$\binom{\nu+\kappa_1+\kappa_2+\kappa_3}{\kappa_4}\binom{N-\kappa_1-\kappa_2+\kappa_4}{\kappa_5} \frac{\kappa_5!\,(N+1)!\,(-1)^{\nu+\kappa_1+\kappa_2+\kappa_4}}{(N-\kappa_1-\kappa_2)!\,(\nu+\kappa_1+\kappa_2+\kappa_3)!\,(N-\kappa_1+1)} \times$$
$$\frac{T^{N+1+\nu}}{(s_1T)^{\kappa_5-\kappa_3+1}} \left(\left(\tfrac{k}{M}\right)^{N-\kappa_1-\kappa_2+\kappa_4-\kappa_5} e^{\left(1-\frac{k}{M}\right)s_1T} - \left(\tfrac{k-1}{M}\right)^{N-\kappa_1-\kappa_2+\kappa_4-\kappa_5} e^{\left(1-\frac{k-1}{M}\right)s_1T} \right). \quad (17)$$

For transfroming the right hand side expression of equation (7) we use the correspondence

$$\frac{1}{(s_1+s)^{N+\nu+1}\, s^{j+1}} \quad \bullet\!\!-\!\!\circ \quad \int_0^t \frac{\tau^{N+\nu}\,(t-\tau)^j\, e^{-s_1\tau}}{j!\,(N+\nu)!}\, d\tau\,. \quad (18)$$

Therefore at arbitrary time instants t, with a small integration window $T > 0$ (integrating into the past) and the integration variable flip $\tau \to -\tau$ the right hand side of equation (7) reads

$$\mathrm{rhs} = y^{(j)}(t) \underbrace{\frac{(-1)^{N+\nu+1}\,(N-j)!}{(N+\nu)!} \int_0^T \tau^{N+\nu}\,(T-\tau)^j\, e^{s_1\tau}\, d\tau}_{=:\,\overline{\mathrm{rhs}}} \quad (19)$$

where the integral results from (14) along the lines

$$\int_0^T \tau^{N+\nu}\,(T-\tau)^j\, e^{s_1\tau}\, d\tau = \sum_{i=0}^{j} \binom{j}{i} (-1)^i T^{j-i} \int_0^T \tau^{N+\nu+i}\, e^{s_1\tau}\, d\tau =$$
$$\sum_{i=0}^{j} \binom{j}{i} (-1)^i\,(N+\nu+i)!\,T^{j-i} \left(\left(\tfrac{-1}{s_1}\right)^{N+\nu+i+1} + e^{s_1T} \sum_{l=0}^{N+\nu+i} \frac{(-1)^l}{s_1^{l+1}(N+\nu+i-l)!} T^{N+\nu+i-l} \right). \quad (20)$$

Consequently, we have that

$$\overline{\mathrm{rhs}} = \sum_{i=0}^{j} \binom{j}{i} \frac{(-1)^{N+\nu+i+1}\,(N-j)!\,(N+\nu+i)!}{(N+\nu)!}\, T^j\, T^{N+1+\nu} \times$$
$$\left(\left(\frac{-1}{s_1T}\right)^{N+\nu+i+1} + e^{s_1T} \sum_{l=0}^{N+\nu+i} \frac{(-1)^l}{(s_1T)^{l+1}(N+\nu+i-l)!} \right). \quad (21)$$

Therefore, setting $\mathrm{lhs} = \mathrm{rhs}$, we arrive at

$$\sum_{k=1}^{M} y(t - \tfrac{k}{M}T)\, \overline{\mathrm{lhs}} = y^{(j)}(t)\, \overline{\mathrm{rhs}} \quad (22)$$

and finally obtain

$$y^{(j)}(t) = \sum_{k=1}^{M} y(t - kT_\mathrm{s})\, \hat{\Pi}_{j,k} \quad (23)$$

with

$$\hat{\Pi}_{j,k} = \frac{\overline{\mathrm{lhs}}}{\overline{\mathrm{rhs}}}. \quad (24)$$