Signal Processing 104 (2014) 203–211

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

A unified framework for derivation and implementation of Savitzky–Golay filters

Çağatay Candan [a,*], Hakan Inan [b]

[a] Department of Electrical and Electronics Engineering, Middle East Technical University (METU), Ankara, Turkey
[b] Department of Electrical Engineering, Stanford University, CA, USA

ARTICLE INFO

Article history:
Received 2 September 2013
Received in revised form
8 April 2014
Accepted 14 April 2014
Available online 21 April 2014

Keywords:
Savitzky–Golay filters
Polynomial interpolation
Smoothing
Differentiation
Fractional delay

ABSTRACT

The Savitzky–Golay (SG) filter design problem is posed as the minimum norm solution of an underdetermined equation system. A unified SG filter design framework encompassing several important applications such as smoothing, differentiation, integration and fractional delay is developed. In addition to the generality and flexibility of the framework, an efficient SG filter implementation structure, naturally emerging from the framework, is proposed. The structure is shown to reduce the number of multipliers in the smoothing application. More specifically, the smoothing application, where an Lth degree polynomial to the frame of $2N+1$ samples is fitted, can be implemented with $N-L/2$ multiplications per output sample instead of $N+1$ multiplications with the suggested structure.

1. Introduction

Savitzky–Golay (SG) filters are suggested to reduce the effect of noise on the signals with the polynomial structure. The goal is to retain the salient features of the polynomials while suppressing the additive noise component accompanying the polynomial signal. A typical application can be the position estimation of a vehicle from the noisy measurements of an inertial measurement unit (IMU). If the acceleration of a vehicle is assumed to be constant in an interval, its position is described by a second degree polynomial of time in that interval. The goal of SG filters is to project the noisy observations to the signal space (the space of second degree polynomials for this application) and estimate the desired feature, which can be the velocity at a given time, from the projection result. This appealing analytical interpretation, based on polynomials, makes SG filters popular in several fields like chemistry, physics and other experimental sciences.

The filtering process with SG filters can be considered to take place in two stages. In the first stage, the least-squares polynomial fit to a given set of samples is found. The task of this stage is to reduce the noise by projecting the input to the signal space. In the second stage, the fitted polynomial is processed through a linear functional to extract the desired output. As an example, the functional for the smoothing application is the evaluation functional that evaluates the polynomial at a specific point [1]. It has been recognized by Savitzky and Golay that the cascade application of these two-stages can be realized through a simple linear time invariant (LTI) filtering scheme. The simplicity and efficacy of LTI filtering for the SG filter implementation have attracted significant attention of researchers, especially the ones involved in the experimental sciences.

The signal processing literature on the SG filter is rather scant. SG filters have been brought to the attention of signal processing community with the recent study of

* Corresponding author.
E-mail addresses: ccandan@metu.edu.tr (Ç. Candan), inanh@stanford.edu (H. Inan).

http://dx.doi.org/10.1016/j.sigpro.2014.04.016
0165-1684/

Schafer [2]. In this work, Schafer examines the problem of SG filter design and studies the frequency response of SG filters in detail. An important difference between SG filter design and conventional filter design is that SG filter specifications are given in terms of polynomials such as the highest polynomial degree to be retained; but not through the frequency domain specifications such as the ones for passband and stopband. The work of Schafer connects the conventional frequency domain specifications, say for low pass filter design, to the SG filter design parameters.

To the best of our knowledge, the most comprehensive work on SG filters is the work of Schüssler and Steffen given in [3, Chapter 8]. In this work, the authors present several equivalent definitions for SG filters and establish connections with other topics relevant to the signal processing. Another notable work is the detailed study of Ziegler on the properties of SG smoothing filters [4]. In addition to some other theoretical developments on SG filters [5–7]; there are significantly more number of papers reporting the utilization of SG filters in chemistry, physics, biomedical engineering and other experimental sciences [2,8,9]. The main application of SG filters in these fields is the smoothing/differentiation of the experimentally collected data using relatively few samples in the neighborhood of the samples as illustrated in [10, Fig. 2]. It can be said that the SG filters enable an easy-to-use method (convolution with SG filter impulse response) to realize the least-square polynomial fit to the experimental data.

The main contribution of this paper is to present a general framework for the derivation and implementation of SG filters. SG filters for smoothing, differentiation, integration, and fractional delay operations can be almost effortlessly derived through the described framework. It should be noted that SG filters in the literature are designed separately for each application [7,10]. The framework enables an easy method for the design of general purpose SG filters as illustrated with several examples in this work. In addition, an efficient structure for SG filter implementation is proposed. The structure is especially attractive for the smoothing applications when the fitted polynomial degree (L) is comparable with the number of samples in a frame ($2N+1$). Specifically, SG smoothing filter can be implemented with $N-L/2$ multiplications instead of $N+1$ multiplications per output sample with the proposed structure.

2. Preliminaries

We present a self-contained description for SG filters in this section. The first subsection gives the conventional two-stage definition for the SG filters and outlines the steps of derivation for the SG smoothing filter. The second subsection establishes some connections between the derivative and difference operators which are required for the development of the general framework.

2.1. Conventional two-stage derivation for Savitzky–Golay filters

We describe the operation of SG filters through a simple example illustrated in Fig. 1. The stem plot in Fig. 1 shows the input which is assumed to be corrupted by the additive noise. For the sake of illustration, we assume that the sample with the index of $m=5$ is to be estimated by processing nearest $N=2$ neighbors on each side of this sample. In other words, the samples with the indices from $m=3$ to $m=7$ are used to estimate the sample with the index of $m=5$. The smoothing method is the least-squares (LS) polynomial fit to the given set of samples, followed by the evaluation of the best fit polynomial at $m=5$.

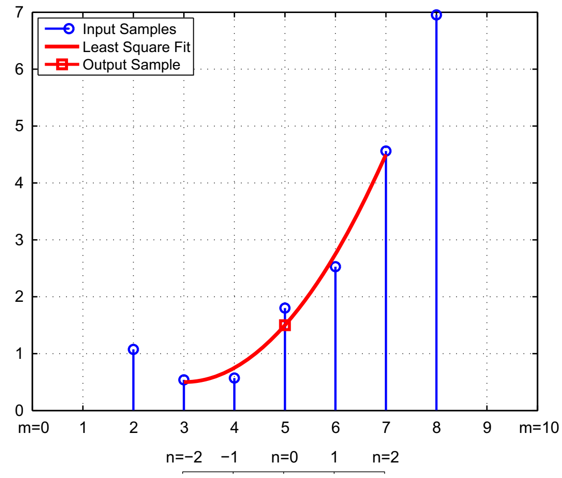

Fig. 1. Savitzky–Golay smoothing filter with $N=L=2$.

At the bottom part of Fig. 1, two axes, namely m-axis and n-axis, are given. The n-axis denotes the index offset from the output sample $n=m-5$. It should be clear that the described two-stage filtering operation through the least-squares polynomial fit can be implemented either over m-axis or n-axis with no change in the final result. This simple observation leads to the important fact that the mentioned two-stage filtering operation is time-invariant. In this work, we use n-axis to define the SG filters. Hence, the input $x[n]$ is considered to be in the interval of $-N \le n \le N$, where N is the maximum index offset from the output sample. It should also be noted that the SG filters designed within this framework are of length $2N+1$ samples. (In some applications, the length of the filter, $2N+1$ samples, is also denoted as the frame length.)

In order to describe the LS polynomial fit operation, we express the input samples and the samples of the polynomial functions in terms of vectors. (All variables with the boldface letters represent column vectors.) As an example, the input $x[n]$ is written in the vector form as follows:

$$\mathbf{x}^{\mathbf{T}} = [x[-N]x[-N+1]\ldots x[0]\ldots x[N-1]x[N]]^{\mathbf{T}}. \quad (1)$$

Similarly, the samples of the canonical polynomials, t^k, over the interval of $-N \le n \le N$, are written in the vector form as

$$\mathbf{t}_{\mathbf{k}}^{\mathbf{T}} = [(-N)^k(-N+1)^k\ldots 0\ldots(N-1)^k N^k]^{\mathbf{T}}. \quad (2)$$

The elements of the vector $\mathbf{t}_{\mathbf{k}}$ are simply the kth power of the integers in the interval $[-N,N]$. In addition to the

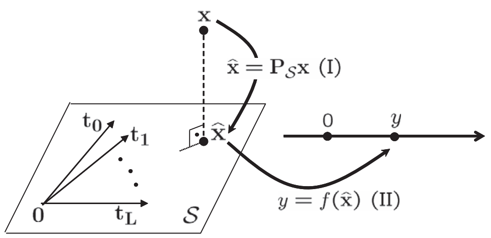

Fig. 2. Stages of Savitzky–Golay filtering.

polynomials, the constant function is denoted with all ones vector $\mathbf{1}$, $\mathbf{1}^\mathbf{T}=[1\ 1\ \ldots\ 1\ 1]^\mathbf{T}$.

The least-square mapping projects the input vector $\mathbf{x}\in\mathcal{R}^{2N+1}$ to $\mathcal{S}=\text{span}\{\mathbf{1},\mathbf{t_1},\mathbf{t_2},\ldots,\mathbf{t_L}\}$. $\mathcal{S}$ is the $L+1$ dimensional subspace of $\mathcal{R}^{2N+1}$, as shown in Fig. 2. The projection matrix to the space $\mathcal{S}$ can be defined by the introduction of $\mathbf{A}$ matrix:

$$\mathbf{A}=[\mathbf{1}\ \ \mathbf{t_1}\ \ \mathbf{t_2}\ \ \ldots\ \ \mathbf{t_{L-1}}\ \ \mathbf{t_L}]. \tag{3}$$

With this definition, the projection operator from $\mathcal{R}^{2N+1}$ to $\mathcal{S}$ becomes $\mathbf{P}_\mathcal{S}=\mathbf{A}(\mathbf{A}^\mathbf{T}\mathbf{A})^{-1}\mathbf{A}^\mathbf{T}$.

The projection to the space $\mathcal{S}$ is the first stage of SG filtering operation. In the second stage, SG filter output is produced by processing the projection result through a linear functional. For smoothing application, the functional evaluates the LS fit at a specific point. In general, the functional maps the projection result from $\mathcal{S}$ to the real line as shown in Fig. 2.

For the smoothing application shown in Fig. 1, the SG filter output is the evaluation result of the best fit polynomial at $n=0$ or $t=0$. Hence, once the input is projected to the space $\mathcal{S}$, $\widehat{\mathbf{x}}=\mathbf{P}_\mathcal{S}\mathbf{x}$; the central entry of the vector $\widehat{\mathbf{x}}$ becomes the output of SG smoothing filter. We can express the cascade of the projection and central entry selection operations as follows:

$$\mathbf{h}^\mathbf{T}=\mathbf{e}_\mathbf{0}^\mathbf{T}\mathbf{P}_\mathcal{S}=\mathbf{e}_\mathbf{0}^\mathbf{T}\mathbf{A}(\mathbf{A}^\mathbf{T}\mathbf{A})^{-1}\mathbf{A}^\mathbf{T}. \tag{4}$$

Here $\mathbf{e_0}$ is the canonical vector composed of all zeros except the entry of 1 at $n=0$ (central entry). Using these definitions, the SG smoothing filter output can be simply calculated by taking the inner product of $\mathbf{h}$ and $\mathbf{x}$, $y=\mathbf{h}^\mathbf{T}\mathbf{x}$.

For the case illustrated in Fig. 1, the projection operator maps $\mathcal{R}^5$ onto three-dimensional $\mathcal{S}=\text{span}\{\mathbf{1},\mathbf{t_1},\mathbf{t_2}\}$. Using (4), the vector $\mathbf{h}$ can be calculated as $\mathbf{h}^\mathbf{T}=1/35[-3\ 12\ 17\ 12\ -3]^\mathbf{T}$ and the SG smoothing output becomes $\mathbf{h}^\mathbf{T}\mathbf{x}$. Since the overall process is time-invariant, the inner product calculation can be implemented through the convolution with the time-reversed version of vector $\mathbf{h}$. It should be noted that for the examined smoothing application, $\mathbf{h}$ is an even sequence and time-reversal operation is not necessary.

2.2. Difference calculus and difference operators

The discrete analog of the differentiation operation, that is the finite difference, is required for the development of general SG filtering framework. We present the following definitions for the first and the second difference in analogy with first and second derivatives:

$$\frac{d}{dt}f(t)\leftrightarrow\nabla_1\{f[n]\}=\frac{1}{2}(f[n+1]-f[n-1]), \tag{5}$$

$$\frac{d^2}{dt^2}f(t)\leftrightarrow\nabla_2\{f[n]\}=f[n+1]-2f[n]+f[n-1]. \tag{6}$$

The higher order differences, corresponding to higher order derivatives, can also be defined as

$$\frac{d^k}{dt^k}f(t)\leftrightarrow\nabla_k\{f[n]\}=\begin{cases}\nabla_1(\nabla_2)^l\{f[n]\}, & k=2l+1\\ (\nabla_2)^l\{f[n]\}, & k=2l\end{cases} \tag{7}$$

Here $(\nabla_2)^l$ represents l-th power of the operator ∇_2, i.e. its l-fold repeated application. It should be noted that the definitions in (7) are different, but closely related with the classical definitions given in [11, Chapter 5].

When $f(t)=t^k$ is sampled with the period of T seconds, $f[n]=f(nT)$; the application of ∇_1 on $f[n]$ becomes an $O(T^2)$ approximation to $(d/dt)f(t)$, $\nabla_1\{f[n]\}=T(d/dt)f(t)\downarrow_{t=nT}+O(T^3)$. The repeated application of this fact shows that the p-th finite difference on the samples of $f(t)=t^k$ yields 0 function when $p>k$ in analogy with the differentiation. This simple fact about the annihilation of polynomials with sufficiently high ordered difference operators becomes a key fact in the derivation of SG filters given in the next section.

3. Minimum norm problem for Savitzky–Golay smoothing filter

The conventional design of SG filters starts with the projection of the input to the space $\mathcal{S}$ (the space of Lth degree polynomials) and then a linear functional is applied on the projection result. Temporarily, we assume that the input is already in the space $\mathcal{S}$, i.e. the input vector $\mathbf{x}$ is already on the plane shown with $\mathcal{S}$ in Fig. 2. Our initial focus is on the functional operating from $\mathcal{S}$ to the real line.

For the smoothing application, the functional is to map $\mathbf{t_k}\in\mathcal{S}$, which corresponds to the samples of t^k in $[-N,N]$, to the value of the function t^k at $t=0$. By writing the desired mapping operation for all $\mathbf{t_k}$ vectors $k=\{0,1,\ldots,L\}$, we get the following system of equations for the functional corresponding to the smoothing application:

$$\underbrace{\begin{bmatrix}1 & \ldots & 1 & \ldots & 1\\ -N & \ldots & 0 & \ldots & N\\ (-N)^2 & \ldots & 0 & \ldots & N^2\\ \vdots & & \vdots & & \vdots\\ (-N)^L & \ldots & 0 & \ldots & N^L\end{bmatrix}}_{\mathbf{A}^\mathbf{T}}\underbrace{\begin{bmatrix}h[-N]\\ \vdots\\ h[0]\\ \vdots\\ h[N]\end{bmatrix}}_{\mathbf{h}}=\underbrace{\begin{bmatrix}1\\0\\0\\ \vdots\\0\end{bmatrix}}_{\mathbf{e}_{(-N)}} \tag{8}$$

The first constraint (the first row) given in (8) shows that all one input (the function of $f(t)=1$) is mapped to 1; the second one enforces that the first power of t sampled in the interval $[-N,N]$ is mapped to the evaluation result at $t=0$, which is 0. The remaining rows similarly enforce the samples of t^k to be mapped to 0. By invoking the linearity, it can be said that the totality of the constraints guarantees that the samples of any Lth degree polynomial $f(t)=a_Lt^L+a_{L-1}t^{L-1}+\cdots+a_0$ in the interval $[-N,N]$ are mapped to $f(0)$.

The equation system given in (8) is underdetermined, hence its solution for $\mathbf{h}$ is not unique. A trivial solution to (8) can be given as $\mathbf{h}=\mathbf{e_0}$, where $\mathbf{e_0}$ is a $(2N+1)$-dimensional canonical vector with all zeros except a single 1 in the central entry. The central claim of the present work is that among the infinite number of solutions the one with the minimum Euclidean norm solution (min-norm solution) is identical to the SG smoothing filter.

The claim can be justified as follows: It is well known that the minimum norm solution (denoted as $\mathbf{x}_{mn}$) of the equation system $\mathbf{Mx}=\mathbf{c}$ is orthogonal to the null space of $\mathbf{M}$, $\mathbf{x}_{mn} \perp \text{null}\{\mathbf{M}\}$ [12]. When this result is applied to the equation system $\mathbf{A^T h}=\mathbf{e}_{(-\mathbf{N})}$ given in (8); we have $\mathbf{h}_{mn} \perp \text{null}\{\mathbf{A^T}\}$. Furthermore, since $\text{null}\{\mathbf{A^T}\} \perp \text{range}\{\mathbf{A}\}$, $\mathbf{h}_{mn}$ has to reside in the range space of $\mathbf{A}$, which is identical to the space of $\mathcal{S}$ from (3). Therefore, $\mathbf{h}_{mn}$ is the unique vector in the range space of $\mathbf{A}$ satisfying the constraints given in (8) and has to coincide with the SG smoothing filter found through the two-stage procedure described in the preliminaries section.

Next, we work towards the retrieval of the minimum norm solution of the equation system given in (8). Since $\mathbf{A^T}$ is a full row-rank matrix, $\text{rank}\{\mathbf{A^T}\}=\text{rank}\{\mathbf{A}\}=L+1$; the dimension of the null space becomes $\text{null}\{\mathbf{A^T}\}=(2N+1)-(L+1)=2N-L$. Hence, we need to find $2N-L$ linearly independent vectors to fully characterize the null space of $\mathbf{A^T}$ matrix. We use finite difference operators examined in the preliminaries section for the characterization of the null space.

As mentioned in the preliminaries section, $\{(L+1), (L+2), \ldots, 2N\}$ order finite differences annihilate Lth order polynomials. This fact is used to describe the null space of $\mathbf{A^T}$ matrix. To this aim, we represent the finite difference operators in terms of vectors.

In order to clarify the procedure of representing finite difference operators with vectors, we explicitly derive the SG smoothing filter for the case given in Fig. 1. The constraint equations for the SG filter shown in Fig. 1 with $N=L=2$ can be written from (8) as follows:

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 \\ (-2)^2 & (-1)^2 & 0 & 1^2 & 2^2 \end{bmatrix}}_{\mathbf{A^T}} \underbrace{\begin{bmatrix} h[-2] \\ h[-1] \\ h[0] \\ h[1] \\ h[2] \end{bmatrix}}_{\mathbf{h}} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{9}$$

By simple substitution, it can be easily verified that vectors $\mathbf{n_1}$ and $\mathbf{n_2}$

$$\text{null}\{\mathbf{A^T}\} = \left\{ \underbrace{\begin{bmatrix} -1/2 \\ 1 \\ 0 \\ -1 \\ 1/2 \end{bmatrix}}_{\mathbf{n_1}}, \underbrace{\begin{bmatrix} 1 \\ -4 \\ 6 \\ -4 \\ 1 \end{bmatrix}}_{\mathbf{n_2}} \right\}, \tag{10}$$

are in the null space of $\mathbf{A^T}$ given in (9).

A simple examination of vectors $\mathbf{n_1}$ and $\mathbf{n_2}$ reveals a connection with the third and fourth difference operators defined in (7). More explicitly, the first vector ($\mathbf{n_1}$) in (10) corresponds to the impulse response of ∇_3, that is $\nabla_3\{\delta[n]\}$, and the second vector ($\mathbf{n_2}$) corresponds to the response of $\nabla_4\{\delta[n]\}$. This connection can be anticipated, since the order of the difference operators (third and fourth) is higher than the degree of polynomials listed as the rows of $\mathbf{A^T}$ in (9); therefore, the row vectors of $\mathbf{A^T}$ matrix are annihilated by the difference operators of higher degrees.

A particular solution $\mathbf{p}$ for the equation system shown in (9) is $\mathbf{p}=[0\ 0\ 1\ 0\ 0]^T$. By calculating the inner product of $\mathbf{p}$ with the vectors given on the right hand side of (10), we can see that $\mathbf{p}$ is not orthogonal to $\mathbf{n_2}$ in (10). Hence, $\mathbf{p}$ is not the minimum norm solution and therefore, it is not the SG filter that we are looking for.

The approach we follow to retrieve the minimum norm solution is to "strip away" the components of a particular solution residing in the null space. For this purpose, we form the matrix $\mathbf{N}$ whose columns are composed of the vectors spanning $\text{null}\{\mathbf{A^T}\}$. The components of $\mathbf{p}$ in the null space of $\mathbf{A^T}$ can be found by $(\mathbf{N^T N})^{-1}\mathbf{N^T p}$. For the examined case, the coefficients of $\mathbf{n_1}$ and $\mathbf{n_2}$ are found as 0 and $3/35$, respectively. Hence, the minimum norm solution becomes $\mathbf{h}_{mn}=\mathbf{p}-3/35\mathbf{n_2}=1/35[-3\ 12\ 17\ 12\ -3]^\mathbf{T}$ coinciding with the result found earlier. This concludes the derivation of SG smoothing filter through an associated minimum norm problem.

An alternative approach for the design of SG filters is the utilization of the Gram polynomials which are the discrete polynomials orthogonal in the interval $[-N, N]$ [3,11]. The Gram polynomials are the discrete analogs of the Legendre polynomials defined in $t \in [0,1]$. The utilization of orthogonal polynomials simplifies the calculation of $\mathbf{P}_\mathcal{S}$ operator and enables an easy way to alter the dimension of the subspace $\mathcal{S}$ (order update). Unfortunately, the complexity of algebraic expressions for the Gram polynomials limits their utilization in the analysis and design of SG filters. Interested readers can examine [10] where a Pascal code is provided for the generation of the Gram polynomials. To assist readers, a Matlab implementation of the suggested framework, which is significantly simpler and more general, is also made available at [13].

4. General framework for Savitzky–Golay filter design

The presented definition through the minimum-norm framework enables an almost effortless generalization of SG smoothing filters to other applications. It should be remembered that the constraint equations are established through the desired functional mapping from the polynomial space $\mathcal{S}$ to the real line. If we denote the desired functional as $f(\cdot)$ as in Fig. 2, then the constraint equation becomes

$$\begin{bmatrix} 1 & \ldots & 1 & \ldots & 1 \\ -N & \ldots & 0 & \ldots & N \\ (-N)^2 & \ldots & 0 & \ldots & N^2 \\ \vdots & & \vdots & & \vdots \\ (-N)^L & \ldots & 0 & \ldots & N^L \end{bmatrix} \begin{bmatrix} h[-N] \\ \vdots \\ h[0] \\ \vdots \\ h[N] \end{bmatrix} = \begin{bmatrix} f(1) \\ f(t) \\ f(t^2) \\ \vdots \\ f(t^L) \end{bmatrix} \tag{11}$$

Here $f(\cdot)$ denotes the linear functional mapping from the subspace $\mathcal{S}$ (polynomial space) to the real numbers [12, p. 104]. To assist the design and implementation of general purpose SG filters, we provide a MATLAB function generating SG filters for various applications in [13].

4.1. Savitzky–Golay differentiation filters

The desired mapping for the differentiation application can be established through the functional evaluating the first derivative at $t=0$. Hence, $f(t^k) = (d/dt)t^k\downarrow_{t=0}$ is equal to 0 for $k \neq 1$ and $f(t^k)=1$ for $k=1$. By substituting $f(t^k)$ values to the right hand side of (11), we can finalize the procedure for constraint equation writing. Specifically for the case of $N=L=2$ the constraint equations for the SG differentiation filters reduce to the following set of equations:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 \\ (-2)^2 & (-1)^2 & 0 & 1^2 & 2^2 \end{bmatrix} \begin{bmatrix} h[-2] \\ h[-1] \\ h[0] \\ h[1] \\ h[2] \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (12)$$

The SG filter design procedure requires an initialization by a particular solution of the constraint equations. For the present application, it can be confirmed that $h[1] = -h[-1] = 1/2$ and $h[k]=0$ for $|k| \neq 1$ is a particular solution. The procedure for finding the minimum norm solution from this particular solution leads to the following SG differentiation filter for $N=L=2$:

$$h[n] = \nabla_1\{\delta[n]\} - \tfrac{2}{5}\nabla_3\{\delta[n]\}. \quad (13)$$

It should be noted that the filter impulse given by (13) corresponds to the inner product (correlation) operation of the input vector $\mathbf{x}$ with $\mathbf{h}^{\mathbf{T}} = [-1/5, -1/10, 0, 1/10, 1/5]^{\mathbf{T}}$.

Table 1 lists the set of possible smoothing and differentiation SG filters for $N=3$. Readers may refer to [4,14] for the properties of SG differentiation filters.

4.2. Savitzky–Golay integration filters

The functional for the integration application can be written as

$$f(t^k) = \int_{-1/2}^{1/2} t^k \, dt = \begin{cases} \dfrac{2^{-k}}{k+1}, & k\text{: even} \\ 0, & k\text{: odd} \end{cases} \quad (14)$$

By substituting the right hand side of the constraint equations (12) with the desired functional values, we finalize the construction of the minimum norm problem for the integration application. For the exemplary case of $N=L=2$, the set of constraint equations becomes as follows:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 \\ (-2)^2 & (-1)^2 & 0 & 1^2 & 2^2 \end{bmatrix} \begin{bmatrix} h[-2] \\ h[-1] \\ h[0] \\ h[1] \\ h[2] \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1/12 \end{bmatrix}. \quad (15)$$

A particular solution for (15) can be found as $\mathbf{h} = \frac{1}{72}[-11, 47, 0, 47, -11]^{\mathbf{T}}$. It should be noted that any other particular solution can be utilized at this stage, since the final result, that is the minimum norm solution, is unique. By running the procedure of finding the minimum norm solution, we can get the integrating SG filter as

$$h[n] = \tfrac{1}{5} S\{\delta[n]\} - \tfrac{23}{24}\nabla_2\{\delta[n]\} - \tfrac{23}{84}\nabla_4\{\delta[n]\}. \quad (16)$$

Here $S\{\delta[n]\}$ denotes the system corresponding to the $\mathbf{s}$ vector, that is the system summing $(2N+1)$ input samples. More details on $S\{\delta[n]\}$ are given in Section 5. Further details of the integration application can be found in [15].

4.3. Savitzky–Golay fractional delay filters

The functional for this application is the evaluation functional with the following definition:

$$f(t^k) = t^k\downarrow_{t=d} = d^k \quad \text{where } |d| < 1/2. \quad (17)$$

Here d corresponds to the value of the fractional delay from the central sample ($n=0$). For the running case of $N=L=2$, the constraint equations become as follows:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 \\ (-2)^2 & (-1)^2 & 0 & 1^2 & 2^2 \end{bmatrix} \begin{bmatrix} h[-2] \\ h[-1] \\ h[0] \\ h[1] \\ h[2] \end{bmatrix} = \begin{bmatrix} 1 \\ d \\ d^2 \end{bmatrix} \quad (18)$$

By following the procedure described earlier, we can get the SG fractional delay filter as follows:

$$h[n] = \frac{S}{5} + d\nabla_1 + \frac{d^2-2}{2}\nabla_2 - \frac{2d}{5}\nabla_3 + \frac{d^2-2}{7}\nabla_4. \quad (19)$$

∇_k appearing in (19) should be interpreted as $\nabla_k\{\delta[n]\}$ as in (16).

The fractional delay SG filters can be considered as the generalized form of the Lagrange interpolation based fractional delay filters [16]. In the Lagrange interpolation, an Nth degree polynomial is fitted to $N+1$ consecutive data samples; hence the Lagrange interpolation, within this context, becomes $L=2N$ special case of the SG filters. Furthermore, the delay value d appearing in (19) can also be altered during run-time as in the Farrow structures [17,16].

Table 1
SG smoothing and differentiation filters for $N=3$.

L	Smoothing	Differentiation
0	$1+2\nabla_2+\nabla_4+1/7\nabla_6$	Not available
1	Same as $L=0$ case	$\nabla-\nabla_3-3/14\nabla_5$
2	$1-3/7\nabla_4-2/21\nabla_6$	Same as $L=1$ case
3	Same as $L=2$ case	$\nabla+1/6\nabla_3+11/63\nabla_5$
4	$1+5/231\nabla_6$	Same as $L=3$ case
5	Same as $L=4$ case	$\nabla+1/6\nabla_3-1/30\nabla_5$
6	1	Same as $L=5$ case

5. On the implementation of SG filters

A side benefit of the minimum norm based derivation is the resulting efficient implementation through the finite difference operators. The root cause of efficiency is the characterization of the null space through the difference

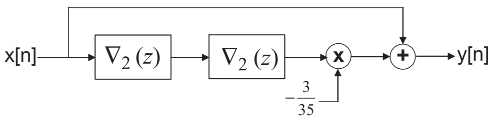

Fig. 3. Implementation of Savitzky–Golay smoothing filter with $N=L=2$.

operators. It can be noted from (7) that ∇_k is the repeated application of ∇_2 and ∇_1 neither of which contains any multiplications other than multiplication by 2 and 1/2. For the exemplary smoothing application shown in Fig. 1, the system with the impulse response of $h[n]=\delta[n]-\frac{3}{35}\nabla_4\{\delta[n]\}$ is the resulting SG filter. The output of this filter can be calculated with a single multiplication, as shown in Fig. 3. It should be noted that the top branch between the filter input $x[n]$ and the output is due to the particular solution selected, which is $\delta[n]$, and the bottom branch is due to the subtraction operation of the components which are in the null space of $\mathbf{A}^\mathbf{T}$ matrix. In the next section, we present a general filtering structure based on finite difference operators for the implementation arbitrary SG filters.

5.1. Expansion of arbitrary FIR filters with sum/difference operators

In this section, we show that an arbitrary FIR filter of length $2N+1$ can be expressed in terms of finite difference and sum operators. The SG smoothing filter shown in Fig. 3 becomes a special case of the general framework.

The output of the difference operator ∇_k at $n=0$, to $x[n]$ can be written as an inner product operation, $y[0]=\mathbf{v_k^T x}$, $\mathbf{x}=[x[-N]\ldots x[0]\ldots x[N]]^\mathbf{T}$. For the running example shown in Fig. 1, the vectors $\mathbf{v_k}$ corresponding to the operation of ∇_k for $k=\{1,2,3,4\}$ are as follows:

$$\mathcal{V}=\left\{\underbrace{\begin{bmatrix}0\\1/2\\0\\-1/2\\0\end{bmatrix}}_{\mathbf{v_1}},\underbrace{\begin{bmatrix}0\\1\\-2\\1\\0\end{bmatrix}}_{\mathbf{v_2}},\underbrace{\begin{bmatrix}-1/2\\1\\0\\-1\\1/2\end{bmatrix}}_{\mathbf{v_3}},\underbrace{\begin{bmatrix}1\\-4\\6\\-4\\1\end{bmatrix}}_{\mathbf{v_4}}\right\}$$

The independence of $\mathbf{v_k}$ vectors can be easily verified using the differences of support (non-zero elements of the vectors) and also the evenness/oddness of the vectors having the same support. The vector of all ones, that is $\mathbf{s}=[1\ 1\ \ldots\ 1\ 1]^\mathbf{T}$, is orthogonal to $\mathbf{v_k}$ vectors, $k=\{1,2,\ldots,2N\}$. Hence, the set of $\mathbf{v_k}$ vectors and the vector $\mathbf{s}$ (denoting the summation of $2N+1$ samples) form a linearly independent set of vectors. This result leads to the fact that any FIR filter of length $2N+1$ points can be expressed in terms of difference and sum operators.

By denoting the z-transform of an FIR filter $h[n]$ ($h[n]=0$ for $|n|>N$) with $H(z)$, we can write the following relation:

$$H(z)=c_0S(z)+\sum_{k=1}^{2N}c_k\nabla_k(z) \quad (20)$$

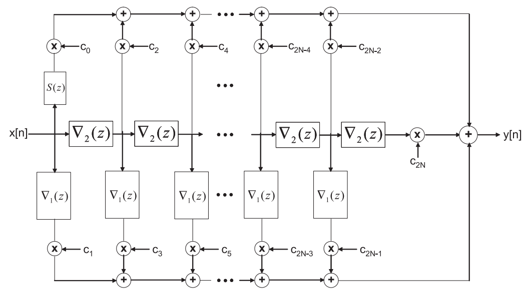

Fig. 4. Suggested filtering structure for Savitzky–Golay filters.

Here $S(z)=z^N+z^{N-1}+\cdots+1+\cdots z^{-N+1}+z^{-N}$ is the z-transform of the summation operator denoted by the vector $\mathbf{s}$ and $\nabla_k(z)$ is the z-transform of the difference operators. Fig. 4 shows the filtering diagram for the suggested expansion.

It should be noted that the expansion given in (20) contains $2N+1$ coefficients shown with c_k, $k=\{0,1,\ldots,2N\}$. Since $S(z)$ and $\nabla_k(z)$ contain only multiplication by 2 and 1/2; the main implementation cost of the suggested structure is due to the multiplications by c_k coefficients.

5.2. Implementation of Savitzky–Golay filters

The structure shown in Fig. 4 can be utilized for the implementation of SG filters. This structure is especially attractive for SG filter applications in which several c_k coefficients are identical to 0, as in Fig. 3.

Specifically for the smoothing application, the conventional FIR filtering implementation for the SG filter requires $N+1$ multiplications per output sample. The suggested structure requires $N-\lfloor L/2\rfloor$ multiplications for the same computation.[1] The cost savings get more significant as L increases.

The cost reduction for a generic SG filter implementation depends on the application and also on the SG filter parameters N and L. As an example, in the fractional delay application the fractional delay value and the choices of N and L jointly determine c_k coefficients. It should be noted that the computational cost of the suggested structure is at the worst case (when $c_k\neq0$ for all k values) identical to the cost of conventional FIR filtering. Hence, there is no loss of computational efficiency in comparison to the conventional method with the adoption of suggested structure. Our aim in this study is to underline the potential of using the structural properties of the SG filters, the relation between the null space and finite difference operators, to our advantage in the reduction of implementation costs.

Another possible usage of the mentioned structure is the concurrent implementation of multiple SG filters on the same input stream. In an application where both the smoothed version of the input and its derivative are of interest, the suggested structure can be utilized to produce two outputs by multiplexing the multipliers among two sets of c_k coefficients.

[1] $\lfloor x\rfloor$ denotes the floor operation, i.e. rounding down the argument to the closest integer.

6. Numerical experiments

Two sets of numerical experiments are provided. The first set uses synthetic data and examines the trade-offs in the choice of N and L in SG filter design. The second set uses experimentally collected EEG data and illustrates the efficacy of the suggested implementation structure.

6.1. Trade offs in SG filter design

The signal of interest is assumed to be a mixture of Gaussian waveforms

$$s(t) = \sum_{k=1}^{4} \exp(-4k^2(t-2k)^2). \tag{21}$$

In this experiment, $s(t)$ is assumed to be observed under additive white Gaussian noise, $r(t) = s(t) + w(t)$ where $w(t)$ is white noise. The noisy observations are sampled at the rate of 50 samples per second, $r[n] = r(n/50)$. Our goal is to reduce the effect of noise by smoothing $r[n]$ samples.

Fig. 5 shows the noisy observation and the desired signal at the signal-to-noise-ratio (SNR) of 20 dB. (SNR is the sample SNR of the input, that is the ratio of average signal power to the noise variance.) Before embarking on the details of the filtering process, we note that the desired signal $s(t)$ is not a polynomial; but can be closely approximated with a polynomial in an interval of interest. In addition, it should be evident from an examination of the consecutive peak points of the mixture components given in Fig. 5 that the radius of curvature for each component is different and the curvature gets smaller as t increases. We may interpret the change of curvature as a type of "non-stationarity" for the desired signal.

The main design issue for the application of SG filters is the choice of N and L. Higher N values increase the length of the window (frame) from which the output is produced. Therefore, higher N values reduce the output noise variance. An increase in L for a fixed N increases the subspace dimension and this leads to an increase in the output noise variance. Hence, N should be set as high as possible, while L should be set as low as possible for the suppression of noise. On the other hand, choosing high N and low L values to reduce the effect of noise at the output can cause significant distortion on the desired signal. This experiment is designed to show the trade-off between noise suppression and signal distortion in the parameter choice for SG filters.

Fig. 6 shows the application of three SG smoothing filters for a fixed window size of $N=8$ and $L=\{0,2,4\}$. The case of $L=0$ corresponds to the simple averaging of eight consecutive samples. This filter simply assumes that the signal of interest is the DC signal. As can be seen from Fig. 6, this filter presents good results in terms of noise suppression; however, it cannot capture rapidly varying signal components and this leads to significant distortion on the desired signal. In the title of each plot in Fig. 6, the noise suppression value of each filter is provided. The noise suppression value is defined as the ratio of the noise variances at the input and output of the filter. The noise suppression value can be expressed in a decibel scale as $10\log_{10}(1/||\mathbf{h}||^2)$ dB where $\mathbf{h}$ is the impulse response of the associated SG filter.

Table 2 presents a detailed distortion and noise variance results for the SG filters shown in Fig. 6. The performance of each SG filter for each mixture component is examined separately. (Four mixture components are $\exp(-4k^2(t-2k)^2)$ for $k=\{1,2,3,4\}$.) The signal distortion is the square error between the mixture component and

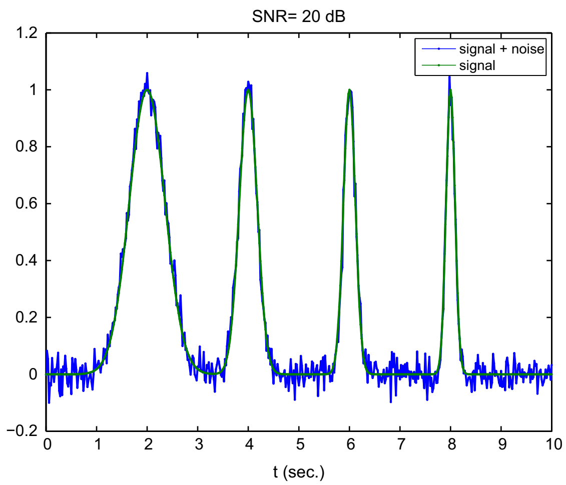

Fig. 5. Desired signal and its noisy version.

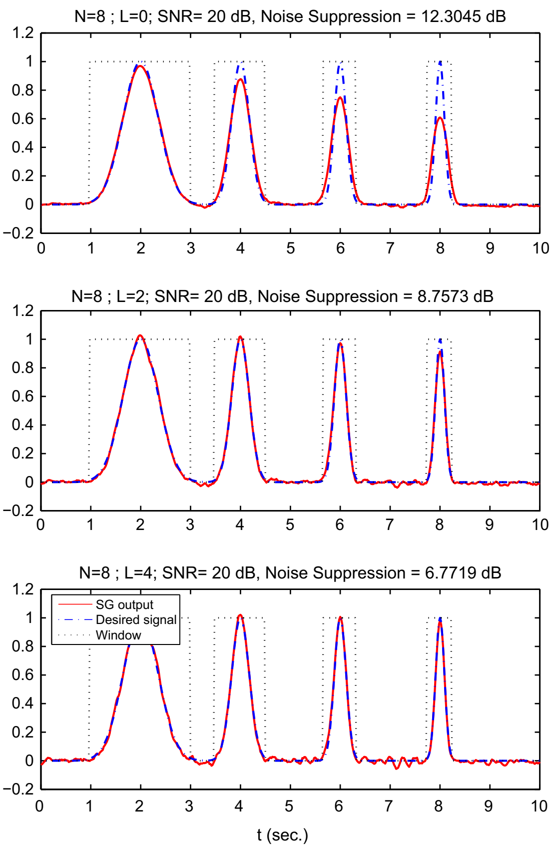

Fig. 6. SG smoothing filter output for $N=8$ and $L=\{0,2,4\}$.

Table 2
Error of SG smoothing filter for the cases shown in Fig. 6.

	Mixture component index (k)			
	$k=1$	$k=2$	$k=3$	$k=4$
$L=0$				
Signal distortion	0.0326	0.2207	0.5719	0.9752
Total noise var.	0.2145	0.1083	0.0701	0.0531
Total MSE	**0.2470**	0.3290	0.6420	1.0283
$L=2$				
Signal distortion	0.0000	0.0007	0.0079	0.0379
Total noise var.	0.4853	0.2451	0.1586	0.1201
Total MSE	0.4853	**0.2457**	**0.1665**	**0.1581**
$L=4$				
Signal distortion	0.0000	0.0000	0.0000	0.0007
Total noise var.	0.7666	0.3871	0.2505	0.1898
Total MSE	0.7666	0.3871	0.2505	0.1904

its filtered version in the absence of noise. The error for each mixture component is calculated subtracting the filter output from the desired signal (the mixture component) in the associated window shown in Fig. 6. The total noise variance in Table 2 refers to the summation of noise variances in the window associated with that mixture. The total MSE is summation of signal distortion and total noise variance and it refers to the total mean square error for the smoothing of the mixture component.

It can be noted from Table 2 that the first mixture component has the smallest total MSE with the choice of $L=0$. The total MSE of other mixture components is minimized with the choice of $L=2$. For the fourth mixture component, the signal distortion is reduced significantly (from 0.0379 to 0.0007) when a SG filter with $L=4$ is utilized instead of $L=2$. Due to the mentioned trade-off, the reduction in signal distortion comes at the cost of an increase in the total noise variance. (The noise variance increases from 0.1201 to 0.1898 in this case.) Since the reduction in signal distortion cannot compensate the increase in the total noise variance; the choice of $L=2$ can be preferred for this scenario. On the other hand, at higher SNR values the choice of $L=4$ can be preferred.

This experiment illustrates the classical trade-offs between noise suppression and signal distortion. For example, the SG smoothing filters are expected to yield a better noise suppression when L is increased; but this is achieved at a potential peril of increased distortion on the desired signal. One immediate suggestion can be the rapid adaptation of the smoothing filter parameters to the input, an adaptive or data dependent operation, as in [18,5].

6.2. SG smoothing filter implementation for EEG applications

A common application that utilizes SG filters is the smoothing of raw EEG data [8,9]. In this application, the raw data is low pass filtered in order to reject the signals beyond the highest frequency activity of interest [8,9]. Fig. 8 shows a real EEG data from [19] which is a recording of P8-O2 channel for 5 s, with the rate of 256 samples per second. Our task is to design a smoothing filter that has a corner frequency around 40 Hz. We utilize SG

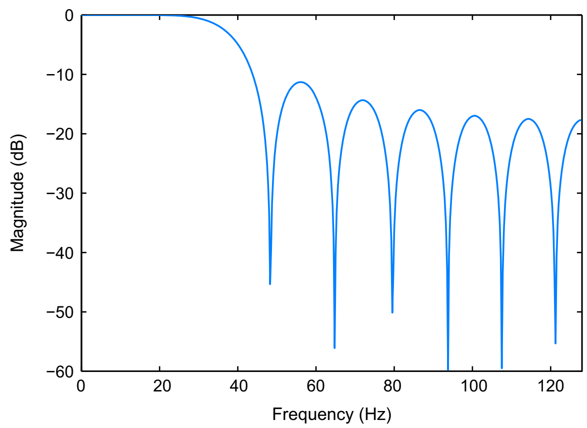

Fig. 7. Magnitude response of SG smoothing filter used in the EEG data smoothing example ($N=10$, $L=8$).

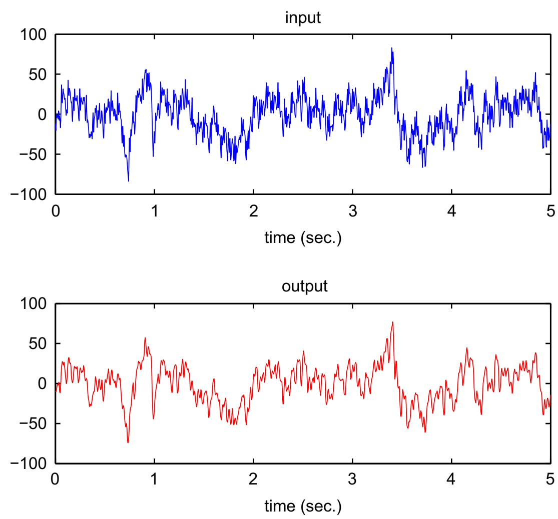

Fig. 8. Input and output waveforms in the EEG data smoothing example.

smoothing filters in this application in order to demonstrate the versatility of the suggested framework.

The SG smoothing filter with $N=10$ and $L=8$ has the magnitude response as in Fig. 7 meeting the mentioned frequency domain specifications. (Further details on SG filter design from frequency domain specifications can be found in [2].) Fig. 8 also displays the smoothed output data below the raw EEG waveform, where it can be observed that the smoothed data maintains the important features of the input such as max–min positions and overall shape.

The advantage of the proposed implementation over the conventional filtering becomes evident when the computation complexities for both implementations are compared. The SG smoothing filter with $N=10$ and $L=8$ has 21 coefficients whose numerical values can be easily generated using the SG filter design code given in [13] by running "SGgeneral(10, 8, 0.^(0: 2*N))" from workspace. The 21 coefficient smoothing filter can be expanded in terms of difference operators as follows:

$$h[n] = \delta[n] + \frac{42}{3335}\nabla_{20}\{\delta[n]\} + \frac{14}{69}\nabla_{18}\{\delta[n]\} + \frac{567}{437}\nabla_{16}\{\delta[n]\} + \frac{1800}{437}\nabla_{14}\{\delta[n]\} + \frac{2100}{323}\nabla_{12}\{\delta[n]\} + \frac{1323}{323}\nabla_{10}\{\delta[n]\} \quad (22)$$

It can be noted from (22) that 21 coefficient filters can be implemented using only six coefficients through the suggested structure.

7. Conclusions

A general framework for the design of the SG filters is described. The framework enables an almost effortless derivation of the SG filters for different applications. In addition, an implementation structure, emerging from the minimum norm problem described within the framework, is given. It has been shown that SG filter implementation with the proposed structure results in a reduced number of multipliers for the smoothing application. The worst case computational complexity of the suggested structure is identical to the conventional FIR filtering; hence, the structure carries the potential to be useful in other SG filtering applications.

To assist the design of SG filters, we provide a MATLAB function generating SG filters and expressing the filter in terms of difference operators in [13]. The MATLAB function can be used to design other SG filters that are not discussed in this work. For example, the filters given by Gorry in [10, Table III] to smooth the end points of the frame (not the central point) can be simply generated by running "SGgeneral(N, L, N.^(0: 2*N))".

References

[1] A. Savitzky, M.J.E. Golay, Smoothing and differentiation of data by simplified least squares procedures, Anal. Chem. 36 (1964) 1627–1639.
[2] R. Schafer, What is a Savitzky–Golay filter? [Lecture Notes], IEEE Signal Process. Mag. 28 (2011) 111–117.
[3] J.S. Lim, A.V. Oppenheim, Advanced Topics in Signal Processing, Signal Processing Series, Prentice Hall, New Jersey, USA, 1988.
[4] H. Ziegler, Properties of digital smoothing polynomial (DISPO) filters, Appl. Spectrosc. 35 (1981) 88–92.
[5] S. Krishnan, C. Seelamantula, On the selection of optimum Savitzky–Golay filters, IEEE Trans. Signal Process. 61 (2013) 380–391.
[6] Q. Quan, K.-Y. Cai, Time-domain analysis of the Savitzky–Golay filters, Digital Signal Process. 22 (2012) 238–245.
[7] J. Luo, K. Ying, J. Bai, Savitzky–Golay smoothing and differentiation filter for even number data, Signal Process. 85 (2005) 1429–1434.
[8] B. Molaee-Ardekani, M. Shamsollahi, O. Tirel, B. Vosoughi-Vahdat, E. Wodey, L. Senhadji, Investigation of the modulation between EEG alpha waves and slow/fast delta waves in children in different depths of desflurane anesthesia, IRBM 31 (2010) 55–66.
[9] J. Hofmanis, O. Caspary, V. Louis-Dorr, R. Ranta, L. Maillard, Denoising depth EEG signals during DBS using filtering and subspace decomposition, IEEE Trans. Biomed. Eng. 60 (2013) 2686–2695.
[10] P.A. Gorry, General least-squares smoothing and differentiation by the convolution (Savitzky–Golay) method, Anal. Chem. 62 (1990) 570–573.
[11] F.B. Hildebrand, Introduction to Numerical Analysis, Dover Publications, New York, USA, 1987.
[12] D.G. Luenberger, Optimization by Vector Space Methods, Wiley, New York, 1990.
[13] C. Candan, Savitzky–Golay Filter Design, MATLAB Code, 2013. URL ⟨http://www.eee.metu.edu.tr/~ccandan/pub.htm⟩.
[14] J. Luo, K. Ying, P. He, J. Bai, Properties of Savitzky–Golay digital differentiators, Digital Signal Process. 15 (2005) 122–136.
[15] C. Candan, Digital wideband integrators with matching phase and arbitrarily accurate magnitude response, IEEE Trans. Circuits Syst. II 58 (2011) 610–614.
[16] C. Candan, An efficient filtering structure for Lagrange interpolation, IEEE Signal Process. Lett. 14 (2007) 17–19.
[17] C.W. Farrow, A continuously variable digital delay element, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 1988, pp. 2641–2645.
[18] G. Vivó-Truyols, P.J. Schoenmakers, Automatic selection of optimal Savitzky–Golay smoothing, Anal. Chem. 78 (2006) 4598–4608. 16808471.
[19] A. Shoeb, Application of machine learning to epileptic seizure onset detection and treatment (Ph.D. thesis), Massachusetts Institute of Technology, MA, USA, 2009. URL ⟨http://physionet.org/pn6/chbmit/⟩.