Closed-form expressions for the finite difference approximations of first and higher derivatives based on Taylor series

Ishiaq Rasool Khan*, Ryoji Ohba1

Division of Applied Physics, Graduate School of Engineering, Hokkaido University, Sapporo 060-8628, Japan

Received 15 November 1998; received in revised form 16 February 1999

Abstract

Numerical differentiation formulas based on interpolating polynomials, operators and lozenge diagrams can be simplified to one of the finite difference approximations based on Taylor series. In this paper, we have presented closed-form expressions of these approximations of arbitrary order for first and higher derivatives. A comparison of the three types of approximations is given with an ideal digital differentiator by comparing their frequency responses. The comparison reveals that the central difference approximations can be used as digital differentiators, because they do not introduce any phase distortion and their amplitude response is closer to that of an ideal differentiator. It is also observed that central difference approximations are in fact the same as maximally flat digital differentiators. In the appendix, a computer program, written in MATHEMATICA is presented, which can give the approximation of any order to the derivative of a function at a certain mesh point. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Finite difference formula; Numerical differentiation; Taylor series; Digital filter

1. Introduction

The Taylor series relates the value of a differentiable function at any point to its first and higher order derivatives at a reference point, and consequently the first (or higher) order derivatives at the reference point can be obtained in terms of the sampled values of the function. In certain cases, it may be difficult to find analytical solutions of complicated differential and partial differential equations describing the physical systems. In such cases, numerical solutions can be obtained by replacing derivatives in the equation by approximations based on the Taylor series. The most commonly used approximations of derivatives are the forward difference, backward difference and

---

* Corresponding author. Tel.: +81-11-706-6628; fax: +81-11-706-7883.
E-mail addresses: khan@eng.hokudai.ac.jp (I.R. Khan), rohba@eng.hokudai.ac.jp (R. Ohba)
1 Tel.: +81-11-706-6626; fax: +81-11-706-7883.
the central difference approximations [1,3–5,8,11,12]. Forward difference approximations are useful when the values of the function are available only at the mesh point and some next (forward) equally spaced points of analysis, while the values of the function are not known at previous (backward) points. In contrast, the backward difference approximations assume that the values of the function at forward points are not known. If the values of the function are available at both the forward and the backward points, then the central difference approximations are the best choice. All these approximations are widely used to solve differential and partial differential equations. Some other numerical methods for the solution of differential and partial differential equations can be found in [6,7,9,15,16,18–26].

Another application of finite difference approximations is numerical differentiation of digital data whose generating function is not known. For example, in controlling the temperature of a centrally heated/air conditioned building, we may want to measure the heating/cooling rate, which is the derivative of the temperature data collected by a digital thermometer. Numerical differentiation is also needed for the functions, which can not be differentiated using analytic techniques. Using the digital circuit concepts, the coefficients of the finite difference approximations can be used as the tap coefficients of a digital differentiator [14]. The optical realization of the FIR qth-order mth derivative digital differentiators based on backward Taylor series expansion can be found in [13]. Certain other numerical differentiation formulas can be found in literature, for example, the formulas based on Lagrangian, Bessel, Newton–Gregory, Gauss, Sterling interpolating polynomials, etc. [5]. A very interesting and useful technique to obtain numerical differentiation formulas is by using operators [5,17]. In fact, all the formulas obtained by interpolating polynomials can be obtained in exactly same or some different forms by using operators. Another very simple way to generate any of these formulas is by using lozenge diagrams given in [8]. All of these formulas use difference tables and by expanding the higher differences step by step to lower differences, each of these formulas can be simplified to either a forward, backward or a central difference formula based on Taylor series. The symmetrical and unsymmetrical formulas given in the appendix of [4] are also equivalent forms of the finite difference formulas. Finite difference formulas expressed explicitly in terms of values of the samples are preferable over other forms using difference tables due to less computations and storage required.

The number of samples used to approximate the derivative defines the order of the approximation and generally, the greater is this number, the more is the accuracy of the approximations. An approximation of order n can be obtained either by solving n equations obtained by n Taylor series expansions each truncated after n + 1 terms, or from other approximations based on interpolating polynomials, operators or lozenge diagrams. Another way to obtain these approximations is the method of indeterminate coefficients, which solves n equations obtained from a polynomial of desired form, by imposing certain conditions on it [8,10]. It must be noted that if the order of the approximation is changed, coefficients of all the terms are to be calculated again by solving new system of equations, and big calculations are involved for approximations of higher orders. In this article we present the closed-form expressions for the first and higher order derivatives of a function for forward, backward and central difference approximations. With these closed-form expressions, the approximations of very high order, and consequently of high accuracy, can be achieved quite easily even by the use of a simple calculator, without the need to solve the system of equations. This will eventually lead to the accurate and efficient analysis of the systems described by differential or partial differential equations.
We have given the comparison of the three finite difference approximations by showing the errors in example differentiation of different functions. Frequency responses of the three are also compared with that of ideal non-recursive digital differentiator. It is found that only the central difference approximations can be used as stable digital differentiator. It is also shown that maximally flat digital differentiators [2], one of the important classes of digital differentiators, are the same as the central difference approximations. This result is important in the sense that it provides a direct relationship between digital differentiators and classical central difference approximations. Therefore a new class of Taylor-series-based higher-order differentiators has been established based on the closed-form expressions of the approximations of higher-order derivatives presented in this paper.

In Sections 2–4 of this article, the closed-form expressions for the forward, backward and central difference approximations of the first-order derivatives are explained. Section 5 gives a comparison of different types of numerical differentiation formulas. In Section 6, we have given the closed-form expressions for the forward and backward difference approximations of second derivative, while the central difference approximations are given for any higher derivative. A computer program is given in the appendix, which gives the derivative of a function using the presented closed-form expressions. In all the analysis, we have used a function of time, however this is not a limitation of the approximations, and the results are equally valid for a function of any variable.

2. Forward difference approximations

The Taylor Series defines the relation between the discrete time values of a time function \( f(t) \) sampled at \( t = kT \), where \( k = 0, \pm 1, \pm 2, \ldots \), and \( T \) is the sampling period, to the value of the function and its derivatives at origin \( t = 0 \). Mathematically, it can be written as

\[
f_k = f_0 + kT \frac{f^{(1)}}{0} + \frac{(kT)^2}{2!} \frac{f^{(2)}}{0} + \cdots + \frac{(kT)^n}{n!} \frac{f^{(n)}}{0} + O(T^{n+1}),
\]

where \( f_k \) denotes the value of \( f(t) \) at \( t = kT \), \( f^{(k)}_0 \) denotes the value of the \( k \)th derivative of \( f \) at \( t = 0 \) and \( O(T^{n+1}) \) is a term of the order of \( T^{n+1} \) coming from the truncation of the series after \( n + 1 \) terms. Using these notations, a set of Taylor series can be written as

\[
F_F = A_F \cdot D_F + O(T^{n+1})
\]

where \( F_F \) and \( D_F \) are the vectors of length \( n, A_F \) is a \( n \times n \) square matrix, and these are defined as

\[
F_F = \begin{bmatrix}
f_1 - f_0 \\
f_2 - f_0 \\
\vdots \\
f_n - f_0 
\end{bmatrix}, \quad D_F = \begin{bmatrix}
f^{(1)}_0 \\
f^{(2)}_0 \\
\vdots \\
f^{(n)}_0 
\end{bmatrix}, \quad A_F = \begin{bmatrix}
T & T^2/2! & \cdots & T^n/n! \\
2T & (2T)^2/2! & \cdots & (2T)^n/n! \\
\vdots & \vdots & \ddots & \vdots \\
nT & (nT)^2/2! & \cdots & (nT)^n/n!
\end{bmatrix}.
\]

The subscript \( F \) denotes the forward difference approximations. We will use the subscripts/superscripts \( C \) and \( B \) for central difference and backward difference approximations, respectively, in the subsequent sections. Also in the subsequent discussions, \( G(p) \) denotes the \( p \)th row of matrix \( G \) or the \( p \)th element of vector \( G \), while \( G(p,q) \) denotes the \( q \)th element in the \( p \)th row of the matrix \( G \). \( A_F(j,k) \) can be defined as \( (jT)^k/k! \).
Neglecting the remainder terms in the set of equations defined by (1), we can solve for the first derivative as

\[
f'_0^{(1)} \approx \begin{bmatrix}
    f_1 - f_0 & T^2/2! & \cdots & T^n/n! \\
    f_2 - f_0 & (2T)^2/2! & \cdots & (2T)^n/n! \\
    \vdots & \vdots & \ddots & \vdots \\
    f_n - f_0 & (nT)^2/2! & \cdots & (nT)^n/n! \\
\end{bmatrix} \cdot \begin{bmatrix}
    1/2! \\
    2^2/2! \\
    \vdots \\
    n^2/2! \\
\end{bmatrix}.
\]

(2)

The matrix \( A_F \) has the special structure that its determinant i.e., the denominator in Eq. (2), is unity for \( T = 1 \), whatever may be the order. Hence, Eq. (2) can be simplified as

\[
f'_0^{(1)} \approx \frac{1}{T} \begin{bmatrix}
    f_1 - f_0 \\
    f_2 - f_0 \\
    \vdots \\
    f_n - f_0 \\
\end{bmatrix} \cdot \begin{bmatrix}
    1/2! \\
    2^2/2! \\
    \vdots \\
    n^2/2! \\
\end{bmatrix}.
\]

(3)

From Eq. (3), it can be seen that the coefficient of a term \( f_k, k = 1, 2, \ldots, n \), is \( 1/T \) times the minor of matrix \( A_F \) corresponding to \( k \)th element of first column, when the value of \( T \) is set to one. The coefficient of \( f_0 \) is additive inverse of the sum of these minors, or \(-1/T\) times the determinant of \( A_F \), when it is made free of \( T \) and every element in first column is set to one. Let us define \( g_{k,n}^{F,1} \) as the coefficient of a term \( f_k \), in a forward difference approximation of first derivative of order \( n \), then Eq. (3) can be written as

\[
f'_0^{(1)} = \frac{1}{T} \sum_{k=0}^{n} g_{k,n}^{F,1} f_k + O(T^n)
\]

or

\[
f'_i^{(1)} = \frac{1}{T} \sum_{k=0}^{n} g_{k,n}^{F,1} f_{k+i} + O(T^n).
\]

(4)

We calculated the coefficients \( g \) for different orders of the approximation, and observed that they can be expressed by the following explicit formulas:

\[
g_{0,n}^{F,1} = -\sum_{j=1}^{n} \frac{1}{j}
\]

(5)

and

\[
g_{k,n}^{F,1} = \frac{(-1)^{k+1}}{k} C_n^k , \quad k = 1, 2, \ldots, n,
\]

(6)

where \( C_n^a \) is defined as \( a!/(a-b)! \).

From Eq. (3), it may be noted that \( g_{0,n}^{F,1} \) is additive inverse of \( \sum_{k=1}^{n} g_{k,n}^{F,1} \). So sum of \( g_{0,n}^{F,1}, k = 0, 1, 2, \ldots, n \) is zero. This satisfies the basic characteristic of the differentiation operation that the derivative of a constant is zero.
Eqs. (4)–(6) define the closed-form expressions for the forward difference approximations of the first derivative of a function. Using these expressions, the approximations for a very high order can be obtained very easily without any need to solve a large set of equations. For higher values of $n$, the calculation of $C^n_k$ may be a computational burden. In such cases, the following iterative procedure can be used to calculate the coefficients in Eq. (6):

$$
g_{k,n}^{F,1} = n,
$$

$$
g_{1,n}^{F,1} = -g_{k-1,n}^{F,1}(k-1)(n-k+1)/k^2, \quad k = 2, 3, 4, \ldots, n. \tag{7}
$$

### 3. Backward difference approximations

In forward difference approximations, the current and the forward values of the function are used to approximate the current value of the derivative of the function, whereas in backward difference approximations the derivative is estimated from the current and the backward values of the function. In this section all the equations and terminology are similar to that in the previous section with the subscript and superscript $F$ replaced by $B$ to denote the backward difference approximations.

A set of Taylor series obtained by $n+1$ terms backward Taylor series expansion of a function $f(t)$ can be written as

$$
F_B = A_B \cdot D_B + O(T^{n+1}), \tag{8}
$$

where $F_B$ is a vector of length $n$ and $A_B$ is a $n \times n$ square matrix, and these are defined as

$$
F_B = \begin{bmatrix}
 f_{-1} - f_0 \\
 f_{-2} - f_0 \\
 \vdots \\
 f_{-n} - f_0
\end{bmatrix}, \quad A_B = \begin{bmatrix}
 -T & T^2/2! & \cdots & (-T)^n/n! \\
 -2T & (2T)^2/2! & \cdots & (-2T)^n/n! \\
 \vdots & \vdots & \ddots & \vdots \\
 -nT & (nT)^2/2! & \cdots & (-nT)^n/n!
\end{bmatrix}.
$$

$A_B(j,k)$ can be defined as $(-jT)^k/k!$ and $D_B$ is the same as $D_F$.

Using the same procedure as in case of forward difference approximations, the first derivative can be written as

$$
f_0^{(1)} \approx -\frac{1}{T} \begin{vmatrix}
 f_{-1} - f_0 & 1/2! & \cdots & 1/n! \\
 f_{-2} - f_0 & 2^2/2! & \cdots & 2^n/n! \\
 \vdots & \vdots & \ddots & \vdots \\
 f_{-n} - f_0 & n^2/2! & \cdots & n^n/n!
\end{vmatrix}. \tag{9}
$$

It can be seen that the coefficient of a term $f_{-k}$ in backward difference approximation is simply the additive inverse of the coefficient of the term of the $f_k$ in forward difference approximation. Let us denote the coefficient of a term $f_k$ as $g_{k,n}^{B,1}$, then a backward difference approximation of order $n$ can be written as

$$
f_1^{(1)} = \frac{1}{T} \sum_{k=-n}^{0} g_{k,n}^{B,1} f_{k+1} + O(T^n), \tag{10}
$$
where

\[ g_{0,n}^{F1} = -g_{0,n}^{B1} = \sum_{j=1}^{n} \frac{1}{j}, \quad (11) \]

and

\[ g_{k,n}^{B1} = g_{k,n}^{F1} = \frac{(-1)^k}{k} c_k, \quad k = 1, 2, \ldots, n. \quad (12) \]

The iterative procedure to calculate the coefficients of forward difference approximations in Eq. (7) can be modified for backward difference approximations as

\[ g_{-1,n}^{B1} = -n, \]

\[ g_{-k,n}^{B1} = g_{-k+1,n}^{B1} \frac{(k-1)(n-k+1)}{k^2}, \quad k = 2, 3, 4, \ldots, n. \quad (13) \]

### 4. Central difference approximations

In central difference approximations, both the backward and the forward values of the function are used to approximate the current value of the derivative. Therefore, the derivative is obtained by solving a set of \(2n\) equations obtained by \(2n+1\) terms Taylor expansion of a function \(f(t)\) at \(t = kT, k = \pm 1, \pm 2, \ldots, \pm n\). These equations can be written as

\[ F_C = A_C D_C + O(T^{2n+1}), \quad (14) \]

where \(F_C\) and \(D_C\) are the vectors of length \(2n\), \(A_C\) is a \(2n \times 2n\) square matrix, and these are defined as

\[ F_C = \begin{bmatrix} f_1 - f_0 \\ f_1 - f_0 \\ f_2 - f_0 \\ \vdots \\ f_n - f_0 \\ f_n - f_0 \end{bmatrix}, \quad D_C = \begin{bmatrix} f_0^{(1)} \\ f_0^{(2)} \\ f_0^{(3)} \\ \vdots \\ f_0^{(2n-1)} \\ f_0^{(2n)} \end{bmatrix}, \]

\[ A_C = \begin{bmatrix} T & T^2/2! & T^3/3! & \cdots & T^{2n}/(2n)! \\ -T & (-T)^2/2! & (-T)^3/3! & \cdots & (-T)^{2n}/(2n)! \\ 2T & (2T)^2/2! & (2T)^3/3! & \cdots & (2T)^{2n}/(2n)! \\ -2T & (-2T)^2/2! & (-2T)^3/3! & \cdots & (-2T)^{2n}/(2n)! \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ nT & (nT)^2/2! & (nT)^3/3! & \cdots & (nT)^{2n}/(2n)! \\ -nT & (-nT)^2/2! & (-nT)^3/3! & \cdots & (-nT)^{2n}/(2n)! \end{bmatrix}. \]

\(A_C(j,k)\) can be defined as \([\text{int}[(j+1)/2], (1)^{k+1} T^j/k!], \) where \(\text{int}[i]\) rounds off \((i - 1/2)\) to the nearest integer value if it is already not an integer. A simplest way to construct the matrix \(A_C\) is to
generate its transpose knowing that

\[ A^T_C[1] = T \cdot \{1, -1, 2, -2, \ldots, n, -n\} \]

and

\[ A^T_C[j] = A^T_C[1]^j/j! , \]

where the operations, multiplying by \(T\) and dividing by \(j!\), defined on a row of matrix mean to be applied to each element in the row. Like in the case of forward and backward difference approximations, determinant of the matrix \(A_C\) is unity, so the derivative can be written as

\[
\begin{align*}
  f_0^{(1)} &\approx \frac{1}{T} \sum_{k=-n}^{n} g_{k,2n} f_{k+i} + O(T^{2n}), \\
  f_0^{(1)} &= \frac{1}{T} \begin{bmatrix} f_1 - f_0 & T^2/2! & T^3/3! & \cdots & T^{2n}/(2n)! \\
                       f_{-1} - f_0 & (-T)^2/2! & (-T)^3/3! & \cdots & (-T)^{2n}/(2n)! \\
                       f_2 - f_0 & (2T)^2/2! & (2T)^3/3! & \cdots & (2T)^{2n}/(2n)! \\
                       f_{-2} - f_0 & (-2T)^2/2! & (-2T)^3/3! & \cdots & (-2T)^{2n}/(2n)! \\
                       \vdots & & & & \\
                       f_n - f_0 & (nT)^2/2! & (nT)^3/3! & \cdots & (nT)^{2n}/(2n)! \\
                       f_{-n} - f_0 & (-nT)^2/2! & (-nT)^3/3! & \cdots & (-nT)^{2n}/(2n)! \\
\end{bmatrix}.
\end{align*}
\]

Denoting \(g_{k,2n}^{C1}\) as the coefficient of a term \(f_k\), the central difference approximation of order \(2n\) can be written as

\[
\begin{align*}
  f_i^{(1)} &= \frac{1}{T} \sum_{k=-n}^{n} g_{k,2n}^{C1} f_{k+i} + O(T^{2n}), \\
  g_{0,2n}^{C1} &= 0 \\
  \text{and} \\
  g_{k,2n}^{C1} &= (-1)^{k+1} \frac{(n!)^2}{k(n-k)(n+k)!}, \quad k = \pm 1, \pm 2, \ldots, \pm n.
\end{align*}
\]

It can be seen here also that \(\sum_{k=-n}^{n} g_{k,2n}^{C1} = 0\), to ensure that the slope of a constant function is zero. Similarly, \(\sum_{k=-n}^{n} k \cdot g_{k,2n}^{C1}\) is always a constant ensuring that the slope of a ramp signal is a constant.

Computational burden can be reduced substantially by using the following procedure to implement Eq. (18):

\[
\begin{align*}
  g_{1,2n}^{C1} &= \frac{n}{n+1}, \\
  g_{k,2n}^{C1} &= -g_{k-1,2n}^{C1} \frac{(k-1)(n-k+1)}{k(n+k)}, \quad k = 2, 3, \ldots, n, \\
  g_{-k,2n}^{C1} &= -g_{k,2n}^{C1}, \quad k = 1, 2, 3, \ldots, n.
\end{align*}
\]
Table 1
Errors in differentiation of example functions with finite difference approximations of different types and orders using different sampling periods

<table>
<thead>
<tr>
<th>Function</th>
<th>Order = 6</th>
<th></th>
<th>Order = 10</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$T = 10^{-3}$</td>
<td>$T = 10^{-6}$</td>
<td>$T = 10^{-3}$</td>
<td>$T = 10^{-6}$</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>C</td>
<td>F</td>
<td>C</td>
</tr>
<tr>
<td>$t - t^6$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$t^7 - t$</td>
<td>7.2E–16</td>
<td>3.6E–17</td>
<td>7.2E–34</td>
<td>3.6E–35</td>
</tr>
<tr>
<td>$e^{-t}$</td>
<td>1.4E–17</td>
<td>7.1E–19</td>
<td>1.4E–35</td>
<td>7.1E–37</td>
</tr>
<tr>
<td>$e^{-100\pi t}$</td>
<td>0.006</td>
<td>0.0007</td>
<td>1.4E–20</td>
<td>6.9E–22</td>
</tr>
<tr>
<td>$\sin(\pi t)$</td>
<td>1.4E–14</td>
<td>6.9E–16</td>
<td>1.4E–32</td>
<td>6.9E–34</td>
</tr>
<tr>
<td>$\sin(100\pi t)$</td>
<td>0.009</td>
<td>0.0007</td>
<td>1.4E–20</td>
<td>6.9E–22</td>
</tr>
<tr>
<td>$\sin(400\pi t)$</td>
<td>37.3</td>
<td>2.1</td>
<td>5.6E–17</td>
<td>2.8E–18</td>
</tr>
<tr>
<td>$\sin(500\pi t)$</td>
<td>66</td>
<td>6.6</td>
<td>2.1E–16</td>
<td>1.1E–17</td>
</tr>
</tbody>
</table>

5. Comparison of different approximations

5.1. Comparison with other formulas

As mentioned before, different numerical differentiation formulas based on interpolating polynomials, operators and lozenge diagrams are, in fact, equivalent form of one of the finite difference formulas based on Taylor series. The forms based on interpolating polynomials, operators and lozenge diagrams use difference tables generated from the samples of data. A difference table generated from $n$ samples has $n(n+1)/2$ entries, each (except the actual samples) generated by difference of other two entries in the table, whereas the explicit forms based on Taylor series use the sample values directly to find the derivative at a mesh point. The latter require, therefore, less computational time and storage to approximate the derivatives, as compared to the former.

5.2. Comparison of the three finite difference approximations

Forward and backward difference approximations of same order have same accuracy, and central difference approximations of same order are generally more accurate. This can be seen in Table 1, where the errors in the differentiation of different functions are tabulated for different sampling periods and two different orders of approximations. The columns headed by F and C show the errors by using forward (or backward) and central difference approximations, respectively. To minimize the round off error, rounding is done after 100 digits. Except in the first two rows for polynomials, all the entries in the table give percentage errors. From the table, certain important points can be noted as given below.

1. An approximation of order $n$ is exact for polynomials of order less than or equal to $n$.
2. Reducing the sampling period reduces the error.
3. As the frequency of oscillating functions is increased, the error is increased.
First two points are well understood, however in order to understand the last point, we must investigate the frequency response of the approximations as given in subsequent subsection.

5.3. Comparison with ideal differentiator

All the finite difference approximations are in the form of a digital filter, magnitude and phase responses of which can be calculated from discrete Fourier transforms of vectors $g^F, g^B$ and $g^C$, given by Eqs. (4), (10) and (16). Transfer function of an ideal differentiator, i.e., the ratio of the output to the input in frequency domain with no initial conditions, is $H(e^{j\omega}) = j\omega$. This can be observed by taking the ratio of the derivatives of different functions to the actual functions in frequency domain. For example, derivative of $\sin(t)$ is $\cos(t)$, and their Fourier transforms are $1/(1 - \omega^2)$ and $j\omega/(1 - \omega^2)$, respectively, giving a ratio of $j\omega$. Therefore, the amplitude response of an ideal fullband differentiator is a ramp function ranging from 0 to 1, as the frequency changes from 0 to $f_s = 1/T$ is the sampling frequency. Phase response should be constant at $\pi/2$, however in practical applications a linear phase response is also acceptable.

The coefficient vector of a backward difference approximation is reverse of that of forward difference approximation. Therefore, the amplitude response of the both is the same and is given in Fig. 1(a) for different orders. It can be seen that the amplitude responses are close to ideal only for very small orders. When the order of the approximation is increased, the error decreases for low frequencies but increases drastically for high frequencies. Therefore, beyond a certain frequency (which gets smaller for higher orders), the low-order approximations become more accurate. This is clear from Table 1 also, where the error increases drastically when the frequency of the sinusoidal function sampled at 1 kHz ($T = 10^{-3}$) is increased from 0.5 to 250 Hz. Error is even higher for higher orders, but reduces with increase in sampling frequency. Magnitude responses of central difference approximations are plotted for different orders in Fig. 1(b) and they show higher accuracy at every frequency with an increase in the order. They have an error less than forward (or backward) difference approximations at every frequency and even very close to the Nyquist frequency ($f_s/2$), they have an error less than 100%. The increase in the error of exponential function in Table 1 can also be explained from the amplitude responses. Since $e^{-100\omega T}$ covers a wider band of frequencies compared to $e^{-t}$, the error is much more in the derivative of the former. However central difference
approximation, having a better response at higher frequencies, is much accurate than forward (or backward) difference approximation.

The phase responses of the three approximations of different orders are given in Fig. 2. It can be seen that both forward and backward difference approximations have non-linear responses, and therefore, are not good as a digital differentiators. Central difference approximations have linear phase responses, so no phase distortion will be introduced in the input signals and therefore they can be used as non-recursive linear phase finite impulse response (FIR) digital differentiators. The tap coefficients of these filters are odd in number and observe odd symmetry, so they belong to type III filters.

It is interesting to note that the coefficients of central difference approximations are the same as those of maximally flat differentiators [2]. The importance of this observation is summarized below.

1. It is proved that maximally flat digital differentiators of order \(N\) are exact for polynomial signals of order less than or equal to \(N\). This result, which comes from the fact that central difference approximation of order \(N\) gives exact differentiation for the polynomials of order less than or equal to \(N\), may be useful in many practical applications, for example, controlling the acceleration of elevators etc.

2. A link has been developed between the classical finite difference formulas and design of a digital differentiator. As many finite difference formulas are available based on different treatments of Taylor series, new procedures can be developed to design the digital differentiators which may prove to be better than the currently available.
3. The method to find to closed-form expressions for central difference approximations of first derivative given in Section 4 can be extended to the approximations of higher-order derivatives (Section 6). These approximations can be implemented as higher-order digital differentiators. Therefore, a new class of higher-order digital differentiators based on Taylor series has been established.

6. Second and higher order derivatives

In this section, we provide the forward, backward and central difference approximations of the second derivative of a function. For higher derivatives, only the central difference approximations are given due to their superiority over others. The basic idea of the determination of an approximation of higher derivative is the same as that for the first derivative. For example a $p$th derivative is simply $1/T^p$ times the determinant of matrix $A$, when its $p$th column is replaced by vector $F$, and $T$ is set to be equal to 1. In this section, we have given only the end results without going into the details of the calculations.

Based on forward values of the function,

$$f^{(2)}_i = \frac{1}{T^2} \sum_{k=0}^{n} g_{k,n}^{F,2} f_{k+i} + O(T^n),$$

(20)

where

$$g_{k,n}^{F,2} = \frac{(-1)^k \binom{n}{k}}{k} \sum_{j=1}^{n} \frac{1}{j}, \quad k = 1, 2, 3, \ldots, n,$$

(21)

$$g_{0,n}^{F,2} = - \sum_{k=1}^{n} g_{k,n}^{F,2}.$$  

(22)

Similarly based on the backward values of the function, we may write

$$f^{(2)}_i = \frac{1}{T^2} \sum_{k=-n}^{0} g_{k,n}^{B,2} f_{k+i} + O(T^n),$$

(23)

where $g_{k,n}^{B,2}$, the coefficient of $f_k$, is the same as the corresponding $k$th coefficient in forward difference approximation given by Eqs. (21) and (22).

A central difference approximation of order $2n$ for $p$th derivative can be written as

$$f^{(p)}_i = \frac{1}{T} \sum_{k=-n}^{n} g_{k,2n}^{C,p} f_{k+i} + O(T^{2n}),$$

(24)

where

$$g_{0,2n}^{C,p} = -2 \sum_{k=1}^{n} g_{k,2n}^{C,p}, \quad p = \text{even},$$

(25)

$$g_{0,2n}^{C,p} = 0, \quad p = \text{odd}$$

(26)
and for $k = \pm 1, \pm 2, \ldots \pm n$,

$$g_{k,2n}^C = (-1)^{k+d} \frac{2!}{k^2} \frac{(n!)^2}{(n-k)(n+k)!},$$  (27)

$$g_{k,2n}^C = (-1)^{k+d} \frac{3!}{k^2} \frac{(n!)^2}{(n-k)(n+k)!} \sum_{i=1 \atop i \neq |k|}^n \frac{1}{i^2},$$  (28)

$$g_{k,2n}^C = (-1)^{k+d} \frac{4!}{k^2} \frac{(n!)^2}{(n-k)(n+k)!} \sum_{i=1 \atop i \neq |k|}^n \frac{1}{i^2},$$  (29)

$$g_{k,2n}^C = (-1)^{k+d} \frac{5!}{k} \frac{(n!)^2}{(n-k)(n+k)!} \sum_{i=1 \atop j=i+1 \atop j \neq |k|}^{n-1} \frac{1}{(ij)^2},$$  (30)

$$g_{k,2n}^C = (-1)^{k+d} \frac{6!}{k^2} \frac{(n!)^2}{(n-k)(n+k)!} \sum_{i=1 \atop j=i+1 \atop j \neq |k|}^{n-1} \frac{1}{(ij)^2},$$  (31)

$$g_{k,2n}^C = (-1)^{k+d} \frac{7!}{k} \frac{(n!)^2}{(n-k)(n+k)!} \sum_{i=1 \atop j=i+1 \atop j \neq |k|}^{n-1} \frac{1}{(ij)^2},$$  (32)

$$g_{k,2n}^C = (-1)^{k+d} \frac{8!}{k^2} \frac{(n!)^2}{(n-k)(n+k)!} \sum_{i=1 \atop j=i+1 \atop j \neq |k|}^{n-1} \frac{1}{(ij)^2},$$  (33)

From Eqs. (27)–(33), we may find a pattern in order to define $k$th coefficient, $k = 1, 2, \ldots, n$, in an approximation of $p$th derivative of order $2n$ as given below

$$g_{k,2n}^C = (-1)^{k+d} \frac{p!}{k^{1+c}} \frac{(n!)^2}{(n-k)(n+k)!} \sum_{i=1}^c \frac{1}{X(i)^2}, \quad k = \pm 1, \pm 2, \ldots \pm n,$$  (34)

where

$$c = \text{int}[(p - 1)/2],$$

$$d = \begin{cases} 1, & c \text{ even}, \\ 0, & c \text{ odd}, \end{cases}$$

$$e = \begin{cases} 1, & p \text{ even}, \\ 0, & p \text{ odd} \end{cases}$$

and the vector $X$ is generated in the following way:
1. Take a vector \( Y \) containing all integers from 1 to \( n \) except \( k \).
2. The vector \( X \) contains the product of all the possible combinations of length \( c \) in \( Y \). The length of \( X \) will be \( \binom{n-1}{c} \).

For example, for fourth coefficient \((k=4)\) in fifth derivative \((p=5)\) of order \( 2n=10 \), \( Y=\{1,2,3,5\} \).
In this case \( c = 2 \), and \( X = \{\{1 \times 2\}, \{1 \times 3\}, \{1 \times 5\}, \{2 \times 3\}, \{2 \times 5\}, \{3 \times 5\}\} = \{2,3,5,6,10,15\} \).

Eqs. (23)–(25), (33) give the explicit formulas to generate a central difference approximation of first or any higher derivative of any arbitrary order.

7. Conclusions

The finite difference approximations based on Taylor series can be used for numerical differentiation of the functions, which are difficult to differentiate analytically. Moreover, these are used extensively for the numerical solutions of differential and partial differential equations. In this paper, we have presented the closed-form expressions for the coefficients of the forward, backward and central difference approximations of first and second derivative for arbitrary orders. Moreover, higher-order central difference approximations are given for any higher-order derivative. With these expressions, it is possible to generate the approximations of very high order with a minimum effort and obtain a maximum accuracy in the numerical solutions of differential and partial differential equations. Other numerical differential formulas based on interpolating polynomials, operators and lozenge diagrams, which are just the equivalent forms and are less efficient than finite difference approximations are no more needed after these closed-form expressions. A computer program written in MATHEMATICA is provided for the numerical differentiation of a function at a certain point. It has also been shown by comparison with an ideal differentiator that the central difference approximations can be used as non-recursive digital differentiators and are in fact the same as maximally flat digital differentiators. Hence, a new class of higher-order digital differentiators is established based on closed-form expressions of higher-order central difference approximations.

Appendix

The computer program given here is written in MATHEMATICA. The commands in MATHEMATICA are very easy to understand because they are written just like the formulas and equations commonly used in Mathematics. Therefore, anyone can transform it very easily in his/her preferred language.

(*Numerical differentiation based on Taylor series*)

\[
\begin{align*}
\text{n}=6; & \quad (* \text{order of approximation}*) \\
\text{f}=&\text{t}^7; \quad (* \text{the function to be differentiated}*) \\
\text{dt}=10^{-3}; & \quad (* \text{sampling period}*) \\
\text{mp}=0; & \quad (* \text{mesh point}*) \\
\end{align*}
\]

(*Forward difference approximations*)
\begin{verbatim}
a = Table[f, {t, mp, mp + n*dt, dt}];  (* samples of signal *)
gf = Table[0, {n}];
gf[1] = n;
For[k = 2, k < = n, k++, 
gf[[k]] = -gf[[k - 1]](k - 1)(n - k + 1)/k^2;
gf0 = Sum[-1/j, {j, 1, n}];
gf = Join[{gf0}, gf];
df = gf.a/dt;
Print[df];

(* Backward difference approximations *)
a = Table[f, {t, mp - n*dt, mp, dt}];
gb = -Reverse[gf];
db = gb.a/dt;
Print[db];

(* Central difference approximations *)
m = n/2;
a = Table[f, {t, mp - m*dt, mp + m*dt, dt}];
gc = Table[0, {m}];
gc[1] = m/(m + 1);
For[k = 1, k < m, k++, 
gc[[k + 1]] = -k(m - k)/(k + 1)/(m + k + 1)gc[[k]];
gc = Join[-Reverse[gc], {0}, gc];
dc = gc.a/dt;
Print[dc];
\end{verbatim}

In MATHEMATICA, kth element of a vector \( A \) is written as \( A[[k]] \) and a dot “.” between two vectors gives their dot product. All the other commands are self-explanatory. The program gives the derivative of the function \( f \), at the mesh point \( mp \), using forward, backward and central difference approximations. Other situations like taking the derivative of a sampled data at all the sampling points can be handled by small changes in the program. Data from a file can be read with the following command

\begin{verbatim}
a = ReadList["path and name of file"];  
\end{verbatim}

To write a vector \( a \) to a file, following command can be used

\begin{verbatim}
Write["path and name of file", a];  
\end{verbatim}

If the sampling period is not known for a sampled data, it can be taken as 1, and the output will be proportional to the derivative.

References