Available online at www.sciencedirect.com
SCIENCE DIRECT
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
ELSEVIER
Journal of Computational and Applied Mathematics 183 (2005) 29–52
www.elsevier.com/locate/cam

General explicit difference formulas for numerical differentiation

Jianping Li*

National Key Laboratory of Numerical Modeling for Atmospheric Science and Geophysical Fluid Dynamics (LASG), Institute of Atmospheric Physics, Chinese Academy of Sciences, P.O. Box 9804, Beijing 100029, China

Received 6 November 2003; received in revised form 5 December 2004

Abstract

Through introducing the generalized Vandermonde determinant, the linear algebraic system of a kind of Vandermonde equations is solved analytically by use of the basic properties of this determinant, and then we present general explicit finite difference formulas with arbitrary order accuracy for approximating first and higher derivatives, which are applicable to unequally or equally spaced data. Comparing with other finite difference formulas, the new explicit difference formulas have some important advantages. Basic computer algorithms for the new formulas are given, and numerical results show that the new explicit difference formulas are quite effective for estimating first and higher derivatives of equally and unequally spaced data.

MSC: 65D25; 15A06; 65L12

Keywords: Numerical differentiation; Explicit finite difference formula; Generalized Vandermonde determinant; Taylor series; Higher derivatives

1. Introduction

Numerical differentiation is not only an elementary issue in numerical analysis [2,7,28], but also a very useful tool in applied sciences [3,26]. In practice, numerical approximations to derivatives are used mainly in two ways. First, we want to compute the derivatives of a function at specified points within its domain. The function is given to us either in the form of a discrete set of argument and function values, or in a continuous analytical form. Second, we use the numerical differentiation formulae in deriving

* Tel.: +86 10 6203 5479; fax: +86 10 6204 3526.
E-mail address: ljp@lasg.iap.ac.cn (J. Li).

0377-0427/$ - see front matter
doi:10.1016/j.cam.2004.12.026

numerical methods for solving ordinary differential equations (ODEs) and partial differential equations (PDEs).

A number of different techniques have been developed to construct useful difference formulas for numerical derivatives. Most approaches of them fall into five categories: finite difference type [2–4,7,17–22,25,26,28], polynomial interpolation type [2,3,7–9,11,27,28], operator type [7,33], lozenge diagrams [10], and undetermined coefficients [10,14]. Other numerical differentiation methods, which do not aim at developing difference formulas of derivatives but evaluating numerical derivatives by use of data or given analytical forms of functions, include: Richardson extrapolation [2,3,7,16,25,28], spline numerical differentiation [36,37], regularization method [1,6,23,35], and automatic differentiation (AD) [5,12,13,30].

The AD is an accurate differentiation technique based on the mechanical application of the chain rule to obtain the derivatives of a function expressed by a computer program, but it cannot be applied to the cases in which analytical expressions of functions are unknown. However, it is very familiar in practice that the derivatives of a function whose values are only obtained empirically at a discrete set of points need to be evaluated. The regularization method on numerical differentiation is effective and stable for estimating the first derivative of a function with non-exact data. This kind of method is divided mainly into three sorts: parameter regularization [31,32], mollification regularization [15,29,35] and variational regularization [24,34]. However, some of the regularization methods strongly depend on a regularization parameter whose optimal value choice is a nontrivial task. Some of the methods are limited to cases in which the spectrum of the data shows a clear division between the signal of the correct function and the noise, and some of them depend on solving a boundary value problem of second order differential equation whose numerical solutions themselves involve difference methods. It is difficult not only to improve the methods mentioned above so that they are approximations of higher order to derivatives, but also to use them directly to find higher derivatives. The dependence on evenly data is also another disadvantage. An alternative approach for evaluating derivatives is to use the Richardson extrapolation [2,3,7,16,25,28]. The technique, however, is actually equivalent to fitting a higher-order polynomial through the data and then computing the derivatives by centered differences [3]. Moreover, for the Richardson extrapolation, the data have to be evenly spaced and generated for successively halved intervals. Evidently, it is not applicable to the case of non-uniform data. A common characteristic of these methods stated above is that they cannot generally provide explicit difference formulas of derivatives for designing difference schemes of both ODEs and PDEs.

Numerical differentiation formulas based on interpolating polynomials (e.g., Lagrangian, Newton, Chebyshev, Hermite, Guass, Bessel, Sterling interpolating polynomials, and etc.) may be found in many literatures [2,3,7–9,11,27,28]. The advantages of the methods are that they do not require that the data be equispaced, and some specific difference formulas deduced from the methods can be used to estimate the derivative anywhere within the range prescribed by the known points. Unfortunately, the methods are generally implicit. Take Lagrange interpolating polynomial as an example, by using the polynomial may generate general derivate approximation formulas as follows:

$$f^{(m)}(x_i) = \sum_{k=0}^{n} L_k^{(m)}(x_i) f(x_k) + R(x_i), \quad i = 0, 1, \ldots, n, \tag{1.1}$$

where $L_k(x)$ denotes the kth Lagrange polynomial for the function f at $x_0, x_1, \ldots, x_n$ in some interval I, $f \in C^{n+1}(I)$, and $R(x)$ is the remainder term Eq. (1.1) is called an $(n+1)$-point formula to approximate $f^{(m)}(x_i)$. In general, it cannot be directly used to calculate the derivatives due to its dependence on $L_k^{(m)}(x)$

which is a very complex polynomial and depends on lower derivatives. Besides, it is also complicated to derive higher-order finite difference formulas by means of the method. By using operators [7,33] and by using lozenge diagrams [10] are other two useful and simple approaches to find numerical differentiation formulas. In fact, all the formulas constructed by interpolating polynomials can be generated by applying operators or lozenge diagrams. However, all of these formulas use difference tables constructed from sampling data and recursive procedures by expanding the higher differences step-by-step to lower differences.

Another alternative way for developing difference formulas is the method of undetermined coefficients, which solves n linear algebraic equations derived from certain polynomial or Taylor expansions, by imposing n necessary conditions on it [10,14]. The calculation complexity of the determination of the coefficients by solving n linear equations is drastically increased while the order of the approximation increases. Moreover, a new system of equations needs to be re-solved to obtain all coefficients if the order of the approximation is changed. Therefore, the method is limited to lower orders due to its complexity in calculation. However, direct use of the method will become wide if the general algebraic solutions of the linear equations on the coefficients can be theoretically found.

Based on Taylor series, recently, Khan et al. [17–21] have presented the explicit forward, backward and central difference formulas of finite difference approximations with arbitrary orders for first derivative, and the central difference approximations for higher derivates. In essential, it can be found from the mathematical proofs of their explicit formulas for the coefficients of finite difference approximations of first derivatives [22] that the explicit formulas were original from the undetermined coefficients by solving a special kind of linear equations. Most advantages of the explicit difference formulas are their convenience in calculations for numerical approximations of arbitrary order to derivatives and their direct use for solution of ODEs and PDEs. The applicability of these explicit formulas, however, appears to be limited to case in which the data had to be equispaced. In contrast, data from experiments or fields studies are often collected at unequal intervals, and such information cannot be analyzed with the explicit formulas mentioned above to this point. Thus, explicit techniques to handle nonequispaced data need to be developed. Besides, the coefficients of explicit central difference formulas for higher derivatives with evenly spaced data were only given in [17,21] based on numerical results, but any mathematical proof was not shown. Obviously, it is also worth studying for explicit forward and backward difference formulas for higher derivatives except that the explicit central difference formulas for higher derivatives in [17,21] should be proved. Moreover, the explicit forward, backward and central difference formulas do not give the whole circumstance. For example, if there are an eight given values of a function f at points $x_0 < x_1 < \cdots < x_7$, and one wants an 8-point difference formula to approximate first derivatives of f at x_1, then the explicit 8-point forward, backward and central difference formulas given in [17–21] are not available to this question since information about f outside the interval is not available. Therefore, the explicit approximation formulas to derivatives near the ends of an interval also need to be developed. In this paper, we study these questions, and present general explicit finite difference formulas for numerical differentiation at unequally or equally spaced gird-points that we believe avoids those limitations mentioned above.

In Section 2, the main explicit difference formulas of arbitrary order for first and higher numerical derivatives are given for unequally or equally spaced data. In Section 3, an nth generalized Vandermonde determinant is introduced, its basic properties are discussed and other some necessary lemmas are given. Moreover, the linear algebraic system of a kind of Vandermonde equations is solved analytically. Based on the lemmas and Taylor series the proofs of the main results described in Section 2 are then shown in Section 4. In Section 5, the new explicit formulas are compared with other finite difference formulas. Section 6 shows a discussion of numerical results with details of our implementation, combining with basic computer algorithms of the new formulas of equally and unequally spaced data for difference approximations of any order to first and higher derivatives of a function whose values are known only at a discrete set of points. Section 7 is devoted to a brief conclusion.

2. Main results

Suppose that $x_1, x_2, \ldots, x_n$ are n distinct real numbers, let

$$\begin{cases} a_n^{(0)} = a_0(x_1, x_2, \ldots, x_n) = x_1 x_2 \cdots x_n, \\ a_n^{(1)} = a_1(x_1, x_2, \ldots, x_n) = x_1 x_2 \cdots x_{n-1} + x_1 x_2 \cdots x_{n-2} x_n + \cdots + x_1 x_3 \cdots x_n + x_2 x_3 \cdots x_n, \\ a_n^{(2)} = a_2(x_1, x_2, \ldots, x_n) = x_1 x_2 \cdots x_{n-2} + \cdots + x_3 x_4 \cdots x_n, \\ \cdots \\ a_n^{(n-2)} = a_{n-2}(x_1, x_2, \ldots, x_n) = x_1 x_2 + x_1 x_3 + \cdots + x_{n-1} x_n, \\ a_n^{(n-1)} = a_{n-1}(x_1, x_2, \ldots, x_n) = x_1 + x_2 + \cdots + x_n, \\ a_n^{(n)} = a_n(x_1, x_2, \ldots, x_n) = 1. \end{cases} \tag{2.1}$$

And let $\Delta_{ij} = x_j - x_i$, writing the first divided difference of the function f with respect to x_i and x_j as

$$D(x_i, x_j) = \frac{f(x_j) - f(x_i)}{x_j - x_i} = \frac{f(x_j) - f(x_i)}{\Delta_{ij}}, \tag{2.2}$$

where $i \neq j$.

Theorem 2.1. *If $x_0 < x_1 < \cdots < x_n$ are $(n+1)$ distinct numbers in the interval $[a, b]$ and f is a function whose values are given at these numbers and $f \in C^{n+1}[a, b]$, then for any one x_i $(i = 0, 1, \ldots, n)$ one can use the linear combination of $D(x_i, x_j)$ $(j = 0, 1, \ldots, n$ and $j \neq i)$ to construct an $(n+1)$-point formula to approximate $f'(x_i)$,* i.e.

$$f'(x_i) = \sum_{j=0, j \neq i}^{n} c_{n,i,j} D(x_i, x_j) + R_n(x_i), \tag{2.3}$$

where the coefficients

$$c_{1,0,1} = 1 \ \ and \ \ c_{1,1,0} = 1, \quad for \ n = 1, \tag{2.4}$$

$$c_{n,i,j} = \prod_{k=0, k \neq i, k \neq j}^{n} \frac{(x_k - x_i)}{(x_k - x_j)}, \quad j \neq i, \ n > 1, \ \ and \ \ i, j = 0, 1, \ldots, n, \tag{2.5}$$

and the remainder term

$$R_1(x_0) = \frac{f''(\xi_0)}{2}(x_0 - x_1) \text{ and } R_1(x_1) = \frac{f''(\xi_1)}{2}(x_1 - x_0), \quad \text{for } n = 1, \tag{2.6}$$

$$R_n(x_i) = \frac{1}{(n+1)!} \prod_{k=0,k\neq i}^{n} (x_i - x_k) \sum_{j=0,j\neq i}^{n} \frac{f^{(n+1)}(\xi_j)(x_i - x_j)^{n-1}}{\prod_{k=0,k\neq i,k\neq j}^{n}(x_k - x_j)}, \quad \text{for } n > 1, \tag{2.7}$$

where ξ_j depends on x_j and x_i.

Remark 2.1. The sum of the weighing coefficients $c_{n,i,j}$ for any reference point x_i is one, i.e.

$$\sum_{j=0,j\neq i}^{n} c_{n,i,j} = 1, \quad i = 0, 1, \ldots, n, \text{ and } n \geqslant 1. \tag{2.8}$$

This property of the differentiation approximation (2.3) guarantees that the first derivative of a linear function is a constant. In fact, for $n > 1$ and any real number x one has a more general formula as follows:

$$\sum_{j=1}^{n} \prod_{k=1,k\neq j}^{n} \frac{(x_k - x)}{(x_k - x_j)} = 1. \tag{2.9}$$

Remark 2.2. For $n > 1$ the coefficients of $f^{(n+1)}(\xi_j)$ $(j = 0, 1, \ldots, n)$ in (2.7) satisfy the following relation:

$$\sum_{j=0,j\neq i}^{n} \frac{(x_i - x_j)^{n-1}}{\prod_{k=0,k\neq i,k\neq j}^{n}(x_k - x_j)} = 1, \quad i = 0, 1, \ldots, n. \tag{2.10}$$

For $n > 1$ and any real number x, in fact, one can write

$$\sum_{j=1}^{n} \prod_{k=1,k\neq j}^{n} \frac{(x - x_j)}{(x_k - x_j)} = 1. \tag{2.11}$$

Corollary 2.1. *If $x_0, x_1, \ldots, x_n$ are $(n+1)$ distinct numbers in the interval $[a, b]$, they are equally spaced notes,* i.e.,

$$x_i = x_0 + ih \ (i = 0, 1, \ldots, n) \quad \text{for some } h \neq 0$$

and f is a function whose values are given at these notes and $f \in C^{n+1}[a, b]$, then for any one x_i $(i = 0, 1, \ldots, n)$ one can use the linear combination of $f(x_j)$ $(j = 0, 1, \ldots, n)$ to construct an $(n+1)$-point formula to approximate $f'(x_i)$, i.e.

$$f'(x_i) = \frac{1}{h} \sum_{j=0}^{n} d_{n+1,i,j} f(x_j) + O_{n,i}(h^n), \tag{2.12}$$

where the coefficients

$$d_{n+1,i,j} = \frac{(-1)^{i-j+1}}{j-i}\frac{i!(n-i)!}{j!(n-j)!}, \quad i, j = 0, 1, \ldots, n \ \ and \ \ j \neq i, \tag{2.13}$$

$$d_{n+1,i,i} = -\sum_{j=0, j\neq i}^{n} d_{n+1,i,j} \tag{2.14}$$

and the remainder term

$$O_{n,i}(h^n) = \frac{(-1)^{n-i} i!(n-i)!h^n}{(n+1)!} \sum_{j=0, j\neq i}^{n} \frac{f^{(n+1)}(\xi_j)(i-j)^n}{(-1)^j j!(n-j)!}, \tag{2.15}$$

where ξ_j *depends on* x_j *and* x_i*.*

Remark 2.3. It can been seen from (2.13) and (2.14) that the sum of the weighing coefficients $d_{n+1,i,j}$ for any given i $(i=0, 1, \ldots, n)$, $\sum_{j=0}^{n} d_{n+1,i,j} = 0$, to ensure that the slope of a constant function is zero. Moreover, $\sum_{j=0}^{n}(j-i)d_{n+1,i,j} = 1$ guarantees that the first derivative of a linear function is a constant.

Remark 2.4. The coefficients $d_{n+1,i,j}$ are anti-symmetric, i.e.

$$d_{n+1,i,j} = -d_{n+1,n-i,n-j}, \tag{2.16}$$

where $n \geqslant 1$, $j = 0, 1, \ldots, n$, $i = 0, 1, \ldots, [n/2]$, here the symbol $[x]$ denotes the greatest integer not greater than x. Especially, when $n=2l$, where l is a natural number, $d_{2l+1,l,l}=0$, $d_{2l+1,l,j} = -d_{2l+1,l,l-j}$, $j = 0, 1, \ldots, l-1$. In this case, one can use the function values at n points (n is an even number) to construct an n order numerical differentiation of the first derivative $f'(x_l)$ at the middle point x_l. As known, at this time the difference approximation is called the central differentiation.

Remark 2.5. In practice, to reduce computational burden, for large number n, one can use the following recursive procedure to calculate the coefficients $d_{n+1,i,j}$:

$$A_0 = 0!n!, \tag{2.17}$$

$$A_i = A_{i-1}\frac{i}{(n-i+1)}, \quad i = 1, \ldots, n, \tag{2.18}$$

$$d_{n+1,i,j} = \frac{(-1)^{i-j+1}}{j-i}\frac{A_i}{A_j}, \quad i = 0, 1, \ldots, [n/2], \ \ j = 0, 1, \ldots, n, \ \ \text{and} \ \ j \neq i \tag{2.19}$$

and $d_{n+1,i,i}$ is calculated by (2.14), and the other coefficients $d_{n+1,i,j}$ $(i=[n/2]+1, \ldots, n, j=0, 1, \ldots, n,$ $j \neq i)$ can be obtained easily by use of the formula (2.16).

Remark 2.6. For $n > 1$ and any given i $(i=0, 1, \ldots, n)$, the coefficients of $f^{(n+1)}(\xi_j)$ $(j=0, 1, \ldots, n)$ in the remainder term (2.15) satisfy

$$\sum_{j=0, j\neq i}^{n} \frac{(i-j)^n}{(-1)^j j!(n-j)!} = 1. \tag{2.20}$$

Theorem 2.2. *If $x_0, x_1, \ldots, x_n$ are $(n+1)$ distinct numbers in the interval $[a, b]$ and f is a function whose values are given at these numbers and $f \in C^{n+1}[a, b]$, then for any one x_i $(i = 0, 1, \ldots, n)$ one can use the linear combination of $D(x_i, x_j)$ $(j = 0, 1, \ldots, n$ and $j \neq i)$ to construct an $(n+1)$-point formula to approximate the mth derivative $f^{(m)}(x_i)$ where $m \leqslant n$,* i.e.

$$f^{(m)}(x_i) = \sum_{j=0, j\neq i}^{n} c_{n,i,j}^{(m)} D(x_i, x_j) + R_n^{(m)}(x_i), \tag{2.21}$$

where the coefficients

$$c_{1,0,1}^{(1)} = 1 \text{ and } c_{1,1,0}^{(1)} = 1, \quad \text{for } n = 1, \tag{2.22}$$

$$c_{n,i,j}^{(m)} = \frac{(-1)^{m-1} m! a_{n-1,i,j}^{(m-1)}}{\prod_{k=0,k\neq i,k\neq j}^{n} (x_k - x_j)}, \quad \text{for } n > 1, \ \ j \neq i \ \text{ and } \ i, j = 0, 1, \ldots, n, \tag{2.23}$$

where $a_{n-1,i,j}^{(m-1)} = a_{m-1}(\Delta_{i0}, \ldots, \Delta_{ik}, \ldots, \Delta_{in})$ $(k = 0, 1, \ldots, n$ and $k \neq i, j)$. The remainder term

$$R_1^{(1)}(x_0) = \frac{f''(\xi_0)}{2}(x_0 - x_1) \quad \text{and} \quad R_1^{(1)}(x_1) = \frac{f''(\xi_1)}{2}(x_1 - x_0), \quad \text{for } n = 1, \tag{2.24}$$

$$R_n^{(m)}(x_i) = \frac{(-1)^{n-m} m!}{(n+1)!} \sum_{j=0, j\neq i}^{n} \frac{f^{(n+1)}(\xi_j) a_{n-1,i,j}^{(m-1)} (x_i - x_j)^n}{\prod_{k=0,k\neq i,k\neq j}^{n} (x_k - x_j)}, \quad \text{for } n > 1, \tag{2.25}$$

where ξ_j depends on x_j and x_i.

Remark 2.7. It may be shown that, with $m \geqslant 2$, the sum of the weighing coefficients $c_{n,i,j}$ for any reference point x_i is zero, i.e.

$$\sum_{j=0, j\neq i}^{n} c_{n,i,j}^{(m)} = 0, \quad 2 \leqslant m \leqslant n, \quad i = 0, 1, \ldots, n. \tag{2.26}$$

This basic characteristic of the differentiation formula (2.12) guarantees that for any $m > 1$ the mth derivative of a linear function is always zero.

Remark 2.8. Generally, the following formula can be given for parts of the coefficients of $f^{(n+1)}(\xi_j)$ $(j = 0, 1, \ldots, n)$ in (2.25)

$$\sum_{j=0, j\neq i}^{n} \frac{(x_i - x_j)^K a_{n-1,i,j}^{(L)}}{\prod_{k=0,k\neq i,k\neq j}^{n} (x_k - x_j)} = \begin{cases} 1, & \text{for } K = L, \\ 0, & \text{for } K \neq L, \end{cases} \tag{2.27}$$

where $0 \leqslant K \leqslant n-1$, $0 \leqslant L \leqslant n-1$, $i = 0, 1, \ldots, n$.

Corollary 2.2. *If $x_0, x_1, \ldots, x_n$ are $(n+1)$ distinct numbers in the interval $[a, b]$, they are equally spaced notes,* i.e.,

$$x_i = x_0 + ih \ (i = 0, 1, \ldots, n) \quad \text{for some } h \neq 0$$

and f is a function whose values are given at these notes and $f \in C^{n+1}[a,b]$, then for any one x_i $(i=0,1,\ldots,n)$ one can use the linear combination of $f(x_j)$ $(j=0,1,\ldots,n)$ to construct an $(n+1)$-point formula to approximate the mth derivative $f^{(m)}(x_i)$, where $m \leqslant n$, i.e.

$$f^{(m)}(x_i) = \frac{1}{h^m} \sum_{j=0}^{n} d_{n+1,i,j}^{(m)} f(x_j) + O_{n,i}^{(m)}(h^{n-m+1}), \tag{2.28}$$

where the coefficients

$$d_{2,0,1}^{(1)} = 1 \text{ and } d_{2,1,0}^{(1)} = -1, \quad \text{for } n=1, \tag{2.29}$$

$$d_{n+1,i,j}^{(m)} = \frac{(-1)^{m-j} m! a_{n-1,i,j}^{(m-1)}}{j!(n-j)!}, \quad j \neq i \text{ and } n>1, \tag{2.30}$$

$$d_{n+1,i,i}^{(m)} = -\sum_{j=0, j\neq i}^{n} d_{n+1,i,j}^{(m)}, \quad \text{for } n \geqslant 1, \tag{2.31}$$

where $a_{n-1,i,j}^{(m-1)} = a_{m-1}(-i,\ldots,k-i,\ldots,n-i)$ $(k=0,1,\ldots,n$ and $k \neq i, j)$. The remainder term

$$O_{n,i}^{(m)}(h^{n-m+1}) = \frac{(-1)^{n-m} m! h^{n-m+1}}{(n+1)!} \sum_{j=0, j\neq i}^{n} \frac{f^{(n+1)}(\xi_j) a_{n-1,i,j}^{(m-1)} (i-j)^{n+1}}{(-1)^j j!(n-j)!}, \tag{2.32}$$

where ξ_i depends on x_j and x_i.

Remark 2.9. It follows from (2.30) and (2.31) that the sum $\sum_{j=0}^{n} d_{n+1,i,j}^{(m)} = 0$ $(i=0,1,\ldots,n)$ for $m \geqslant 1$, ensuring that the slope of a constant function is forever zero. Moreover, $\sum_{j=0}^{n} (j-i) d_{n+1,i,j}^{(m)} = 0$ for $m \geqslant 2$ and $m \leqslant n$, satisfying the basic property that the mth derivative of a linear function with $m>1$ must be zero.

Remark 2.10. The coefficients $d_{n+1,i,j}^{(m)}$ are symmetric for those even numbers m and anti-symmetric for those odd numbers m. That is to say,

$$d_{n+1,i,j}^{(m)} = \begin{cases} d_{n+1,n-i,n-j}^{(m)}, & \text{for } m=2k, \\ -d_{n+1,n-i,n-j}^{(m)}, & \text{for } m=2k+1, \end{cases} \tag{2.33}$$

where k is a natural number, $n \geqslant 1$, $j=0,1,\ldots,n$, $i=0,1,\ldots,[n/2]$. Especially, by letting k and l be positive integers, and $k \leqslant l$, when $m=2k$ and $n=2l$, since $d_{2l+1,l,j}^{(2k)} = d_{2l+1,l,2l-j}^{(2k)}$ $(j=0,1,\ldots,l)$, then $\sum_{j=0,j\neq l}^{2l} (j-l)^{2l+1} d_{2l+1,l,j}^{(2k)} = 0$, and at the same time the remainder

$$O(h^{2l-2k+2}) = \frac{-(2k)! h^{2l-2k+2}}{(2l+2)!} \sum_{j=0, l\neq i}^{n} \frac{f^{(2l+2)}(\xi_j) a_{2l+1,l,j}^{(2k-1)} (l-j)^{2l+2}}{(-1)^j j!(2l-j)!}, \tag{2.34}$$

where ξ_i depends on x_j and x_i. This suggests that in this case the order of central numerical differentiation of the even order derivative $f^{(m)}(x)$ is one more than that of non-central numerical differentiation. If $m=2k$ and $n=2l+1$, one has $a^{(2k-1)}_{2l+1,l,2l+1}=a^{(2k-1)}_{2l+1,l+1,0}=0$, $d^{(2k)}_{2l+2,l,2l+2}=d^{(2k)}_{2l+2,l+1,0}=0$. As a result,

$$d^{(2k)}_{2l+2,l,j}=d^{(2k)}_{2l+2,l+1,j+1}=d^{(2k)}_{2l+1,l,j}, \quad j=0,1,\ldots,2l+1,$$

$$d^{(2k)}_{2l+2,l,j}=d^{(2k)}_{2l+2,l,2l-j}, \quad j=0,1,\ldots,l-1,$$

$$d^{(2k)}_{2l+2,l+1,j}=d^{(2k)}_{2l+2,l+1,2l+2-j}, \quad j=1,\ldots,l.$$

For $m=2k+1$ and $n=2l$, it follows from (2.34) that

$$d^{(2k+1)}_{2l+1,l,l}=0.$$

Hence, we may use the function values at n points (n is an even number) to construct a numerical differentiation of n order for the odd order derivative $f^{(m)}(x)$ at the centered point x_l.

3. Some lemmas

As known, an nth Vandermonde determinant V_n is defined as

$$V_n=V(x_1,x_2,\ldots,x_n)=\begin{vmatrix} 1 & 1 & \cdots & 1\\ x_1 & x_2 & \cdots & x_n\\ x_1^2 & x_2^2 & \cdots & x_n^2\\ \cdots & \cdots & \cdots & \cdots\\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{vmatrix}. \tag{3.1}$$

Introducing the nth generalized Vandermonde determinant $V_n^{(i)}$ as follows, for $i=0$,

$$V_n^{(0)}=V^{(0)}(x_1,x_2,\ldots,x_n)=\begin{vmatrix} x_1 & x_2 & \cdots & x_n\\ x_1^2 & x_2^2 & \cdots & x_n^2\\ x_1^3 & x_2^3 & \cdots & x_n^3\\ \cdots & \cdots & \cdots & \cdots\\ x_1^{n} & x_2^{n} & \cdots & x_n^{n} \end{vmatrix}, \tag{3.2}$$

for $i=1,\ldots,n-1$,

$$V_n^{(i)}=V^{(i)}(x_1,x_2,\ldots,x_n)=\begin{vmatrix} 1 & 1 & \cdots & 1\\ x_1 & x_2 & \cdots & x_n\\ \cdots & \cdots & \cdots & \cdots\\ x_1^{i-1} & x_2^{i-1} & \cdots & x_n^{i-1}\\ x_1^{i+1} & x_2^{i+1} & \cdots & x_n^{i+1}\\ \cdots & \cdots & \cdots & \cdots\\ x_1^{n} & x_2^{n} & \cdots & x_n^{n} \end{vmatrix} \tag{3.3}$$

and for $i=n$,

$$V_n^{(n)}=V^{(n)}(x_1,x_2,\ldots,x_n)=V(x_1,x_2,\ldots,x_n)=V_n. \tag{3.4}$$

Lemma 3.1. *For any* i $(i = 0, 1, \ldots, n)$,

$$V_n^{(i)} = a_n^{(i)} V_n. \tag{3.5}$$

Proof. For the cases of $i = 0$ and $i = n$, the lemma is obviously true. For the cases of $i = 1, \ldots, n-1$, constructing an $(n+1)$th Vandermonde determinant

$$g(y) = V(x_1, x_2, \ldots, x_n, y) = \begin{vmatrix} 1 & 1 & \cdots & 1 & 1 \\ x_1 & x_2 & \cdots & x_n & y \\ x_1^2 & x_2^2 & \cdots & x_n^2 & y^2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^n & x_2^n & \cdots & x_n^n & y^n \end{vmatrix}$$

$$= V_n \prod_{i=1}^{n} (y - x_i).$$

It is easy to know that x_i $(i = 1, \ldots, n)$ are roots of the polynomial $g(y)$. In the light of relations between roots and coefficients we have the coefficients of y^i $(i = 1, \ldots, n-1)$ of the polynomial $g(y)$ are

$$(-1)^{n-i} a_n^{(i)} V_n, \quad (i = 1, \ldots, n-1).$$

On the other hand, the coefficients of y^i $(i = 1, \ldots, n-1)$ of the polynomial $g(y)$ also equal to

$$(-1)^{n+2+i} V_n^{(i)}, \quad (i = 1, \ldots, n-1).$$

Thus, $V_n^{(i)} = a_n^{(i)} V_n$. □

Lemma 3.2. *For any* j $(j = 1, \ldots, n)$,

$$V_n = (-1)^{1+j} V_{n-1,j} \prod_{i=1, i \neq j}^{n} (x_i - x_j), \tag{3.6}$$

where

$$V_{n-1,j} = V(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n). \tag{3.7}$$

Let

$$W_{n-1,j}^{(i)} = W_{n-1,j}^{(i)}(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n) = \begin{vmatrix} 1 & \cdots & 1 & 0 & 1 & \cdots & 1 \\ x_1 & \cdots & x_{j-1} & 0 & x_{j+1} & \cdots & x_n \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^{i-1} & \cdots & x_{j-1}^{i-1} & 0 & x_{j+1}^{i-1} & \cdots & x_n^{i-1} \\ x_1^i & \cdots & x_{j-1}^i & 1 & x_{j+1}^i & \cdots & x_n^i \\ x_1^{i+1} & \cdots & x_{j-1}^{i+1} & 0 & x_{j+1}^{i+1} & \cdots & x_n^{i+1} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_1^{n-1} & \cdots & x_{j-1}^{n-1} & 0 & x_{j+1}^{n-1} & \cdots & x_n^{n-1} \end{vmatrix}, \tag{3.8}$$

where $i = 0, 1, \ldots, n-1$, $j = 1, \ldots, n$.

It follows from Lemma 3.1 that

Lemma 3.3. *For any* $i = 0, 1, \ldots, n-1$ *and* $j = 1, \ldots, n$,

$$W_{n-1,j}^{(i)} = (-1)^{i+1+j} V_{n-1,j}^{(i)} = (-1)^{i+1+j} a_{n-1,j}^{(i)} V_{n-1,j}, \tag{3.9}$$

where

$$V_{n-1,j}^{(i)} = V^{(i)}(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n), \tag{3.10}$$

$$a_{n-1,j}^{(i)} = a_i(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n) \tag{3.11}$$

and $V_{n-1,j}$ *is given by* (3.7).

Lemma 3.4. *For any* i $(i = 0, 1, \ldots, n-1)$,

$$\sum_{j=1}^{n} x_j^i W_{n-1,j}^{(k)} = \begin{cases} V_n, & \text{for } i = k, \\ 0, & \text{for } i \neq k \end{cases} \tag{3.12}$$

and for any j $(j = 1, \ldots, n)$,

$$\sum_{i=0}^{n-1} x_j^i W_{n-1,k}^{(i)} = \begin{cases} V_n, & \text{for } j = k, \\ 0, & \text{for } j \neq k. \end{cases} \tag{3.13}$$

Remark 3.1. Remarks 2.1, 2.2, 2.7 and 2.8 can be derived from Lemma 3.4.

Theorem 3.1. *The linear algebraic system of equations*

$$VC = G_i, \tag{3.14}$$

where

$$V = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{bmatrix}, \tag{3.15}$$

$$C = (c_1, c_2, \ldots, c_n)^{\mathrm{T}}, \tag{3.16}$$

$$G_i = (\underbrace{0, \ldots, 0,}_{i-1}\, g, 0, \ldots, 0)^{\mathrm{T}}, \tag{3.17}$$

$(i = 1, \ldots, n)$, *then the solution of the system is*

$$c_j = \frac{(-1)^{i-1} g a_{n-1,j}^{(i-1)}}{\prod_{k=1,k\neq j}^{n} (x_k - x_j)}, \quad \text{for } j = 1, \ldots, n, \tag{3.18}$$

where $a_{n-1,j}^{(i-1)}$ *is given by* (3.11).

Proof. Using the Cramer's rule to this linear algebraic system, we have its solution

$$c_j = \frac{gW_{n-1,j}^{(i-1)}}{V_n}, \quad \text{for } j = 1, \ldots, n.$$

From Lemmas 3.2 and 3.3 one has

$$c_j = \frac{g(-1)^{i+j} a_{n-1,j}^{(i-1)} V_{n-1,j}}{(-1)^{1+j} V_{n-1,j} \prod_{k=1,k\neq j}^{n} (x_k - x_j)} = \frac{(-1)^{i-1} g a_{n-1,j}^{(i-1)}}{\prod_{k=1,k\neq j}^{n} (x_k - x_j)}, \tag{3.19}$$

where $i = 1, \ldots, n,\ j = 1, \ldots, n$. The theorem is true. □

4. Proofs of the main theorems

Now we give the proofs of the main Theorems 2.1 and 2.2 in this paper. Suppose the function $f \in C^n[a, b]$, that $f^{(n+1)}$ exits on the interval $[a, b]$, and $x_0 < x_1 < \cdots < x_n$ are $(n + 1)$ numbers in $[a, b]$, now only consider the case of $n > 1$. Applying the Taylor series gives

$$D(x_i, x_j) = f'(x_i) + \frac{\Delta_{ij}}{2!} f''(x_i) + \frac{\Delta_{ij}^2}{3!} f^{(3)}(x_i) + \cdots + \frac{\Delta_{ij}^{n-1}}{n!} f^{(n)}(x_i) + r_n(x_j), \tag{4.1}$$

where $i, j = 0, 1, \ldots, n$ and $j \neq i$, the first divided difference $D(x_i, x_j) = (f(x_j) - f(x_i))/\Delta_{ij}$, $\Delta_{ij} = x_j - x_i$, and the remainder term $r_n(x_j) = [\Delta_{ij}^n/(n + 1)!] f^{(n+1)}(\xi_j)$.

Proof of Theorem 2.1. For a given x_i $(i = 0, 1, \ldots, n)$, if one use the linear combination of $D(x_i, x_j)$ $(j = 0, 1, \ldots, n$ and $j \neq i)$ to construct an $(n + 1)$-point formula to approximate $f'(x_i)$, i.e.

$$f'(x_i) = \sum_{j=0, j\neq i}^{n} c_{n,i,j} D(x_i, x_j) + R_n(x_i), \tag{4.2}$$

then in the light of (4.1) the coefficients $c_{n,i,j}$ can be determined by solving the following linear system:

$$VC = G_1, \tag{4.3}$$

where

$$V = \begin{bmatrix} 1 & \cdots & 1 & 1 & \cdots & 1 \\ \Delta_{i0} & \cdots & \Delta_{i(i-1)} & \Delta_{i(i+1)} & \cdots & \Delta_{in} \\ \Delta_{i0}^2 & \cdots & \Delta_{i(i-1)}^2 & \Delta_{i(i+1)}^2 & \cdots & \Delta_{in}^2 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \Delta_{i0}^{n-1} & \cdots & \Delta_{i(i-1)}^{n-1} & \Delta_{i(i+1)}^{n-1} & \cdots & \Delta_{in}^{n-1} \end{bmatrix}, \tag{4.4}$$

$$C = (c_{n,i,0}, \ldots, c_{n,i,i-1}, c_{n,i,i+1}, \ldots, c_{n,i,n})^{\mathrm{T}}, \tag{4.5}$$

$$G_1 = (1, 0, \ldots, 0)^{\mathrm{T}}. \tag{4.6}$$

Using Theorem 3.1 one has

$$c_{n,i,j} = \frac{W^{(0)}_{n-1,i,j}}{V_{n,i}} = \frac{a^{(0)}_{n-1,i,j}}{\prod_{k=0,k\neq i,k\neq j}^{n} \Delta_{ik}} = \prod_{k=0,k\neq i,k\neq j}^{n} \frac{(x_k - x_i)}{(x_k - x_j)}, \tag{4.7}$$

where $V_{n,i} = V_n(\Delta_{i0}, \Delta_{i1}, \ldots, \Delta_{i(i-1)}, \Delta_{i(i+1)}, \ldots, \Delta_{in})$, $W^{(0)}_{n-1,i,j} = W^{(0)}_{n-1,i,j}(\Delta_{i0}, \ldots, \Delta_{ik}, \ldots, \Delta_{in})$ and $a^{(0)}_{n-1,i,j} = a_0(\Delta_{i0}, \ldots, \Delta_{ik}, \ldots, \Delta_{in})$ ($k = 0, 1, \ldots, n$ and $k \neq i, j$). The remainder term

$$\begin{aligned} R_n(x_i) &= -\frac{1}{(n+1)!} \sum_{j=0, j\neq i}^{n} f^{(n+1)}(\xi_j)\Delta_{ij}^{n} c_{n,i,j} \\ &= -\frac{1}{(n+1)!} \prod_{k=0,k\neq i}^{n} (x_k - x_i) \sum_{j=0, j\neq i}^{n} \frac{f^{(n+1)}(\xi_j)\Delta_{ij}^{n-1}}{\prod_{k=0,k\neq i,k\neq j}^{n} (x_k - x_j)} \\ &= \frac{1}{(n+1)!} \prod_{k=0,k\neq i}^{n} (x_i - x_k) \sum_{j=0, j\neq i}^{n} \frac{f^{(n+1)}(\xi_j)(x_i - x_j)^{n-1}}{\prod_{k=0,k\neq i,k\neq j}^{n} (x_k - x_j)}, \end{aligned} \tag{4.8}$$

for $n > 1$. Therefore, the proof of Theorem 2.1 is complete. □

Proof of Theorem 2.2. If, for any one x_i $(i = 0, 1, \ldots, n)$, we use the linear combination of $D(x_i, x_j)$ ($j = 0, 1, \ldots, n$ and $j \neq i$) to construct an $(n+1)$-point formula to approximate mth derivative $f^{(m)}(x_i)$, where $m \leqslant n$, i.e.

$$f^{(m)}(x_i) = \sum_{j=0, j\neq i}^{n} c^{(m)}_{n,i,j} D(x_i, x_j) + R^{(m)}_n(x_i), \tag{4.9}$$

then it follows from (4.1) that the coefficients $c^{(m)}_{n,i,j}$ can be determined by solving the following linear system:

$$VC = G_m, \tag{4.10}$$

$$V = \begin{bmatrix} 1 & \cdots & 1 & 1 & \cdots & 1 \\ \Delta_{i0} & \cdots & \Delta_{i(i-1)} & \Delta_{i(i+1)} & \cdots & \Delta_{in} \\ \Delta_{i0}^2 & \cdots & \Delta_{i(i-1)}^2 & \Delta_{i(i+1)}^2 & \cdots & \Delta_{in}^2 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \Delta_{i0}^{n-1} & \cdots & \Delta_{i(i-1)}^{n-1} & \Delta_{i(i+1)}^{n-1} & \cdots & \Delta_{in}^{n-1} \end{bmatrix}, \tag{4.11}$$

$$C = (c^{(m)}_{n,i,0}, \ldots, c^{(m)}_{n,i,i-1}, c^{(m)}_{n,i,i+1}, \ldots, c^{(m)}_{n,i,n})^{\mathrm{T}}, \tag{4.12}$$

$$G_m = (\underbrace{0, \ldots, 0,}_{m-1}\ m!, 0, \ldots, 0)^{\mathrm{T}}. \tag{4.13}$$

From Theorem 3.1 we have

$$c^{(m)}_{n,i,j} = \frac{W^{(m-1)}_{n-1,i,j}}{V_{n,i}} = \frac{(-1)^{m-1} m!\, a^{(m-1)}_{n-1,i,j}}{\prod_{k=0,k\neq i,k\neq j}^{n} \Delta_{ik}} = \frac{(-1)^{m-1} m!\, a^{(m-1)}_{n-1,i,j}}{\prod_{k=0,k\neq i,k\neq j}^{n} (x_k - x_i)}, \tag{4.14}$$

where $a^{(m-1)}_{n-1,i,j} = a_{m-1}(\Delta_{i0}, \ldots, \Delta_{ik}, \ldots, \Delta_{in})$ $(k = 0, 1, \ldots, n$ and $k \neq i,\ j)$. The remainder term

$$
\begin{aligned}
R_n^{(m)}(x_i) &= -\frac{1}{(n+1)!} \sum_{j=0,\, j\neq i}^{n} f^{(n+1)}(\xi_j)\Delta_{ij}^n c^{(m)}_{n,i,j} \\
&= \frac{(-1)^m m!}{(n+1)!} \sum_{j=0,\, j\neq i}^{n} \frac{f^{(n+1)}(\xi_j) a^{(m-1)}_{n-1,i,j}\Delta_{ij}^n}{\prod_{k=0,k\neq i,k\neq j}^{n}(x_k - x_j)} \\
&= \frac{(-1)^{n-m} m!}{(n+1)!} \sum_{j=0,\, j\neq i}^{n} \frac{f^{(n+1)}(\xi_j) a^{(m-1)}_{n-1,i,j}(x_i - x_j)^n}{\prod_{k=0,k\neq i,k\neq j}^{n}(x_k - x_j)},
\end{aligned}
\tag{4.15}
$$

where ξ_j depends on x_j and x_i. The proof of Theorem 2.2 is then complete. □

Besides, Corollaries 2.1 and 2.2 are easily deduced from Theorems 2.1 and 2.2, respectively.

5. Comparison with other finite difference approximations

To compare with other finite difference approximations for numerical derivatives, we first show some special numerical differentiation formulas from the new method in this paper. As a matter of convenience, we write $f_k = f(x_k) = f(x_0 + kh)$ for equally spaced notes, where h is the stepsize or sampling period. From Corollaries 2.1 and 2.2, we have the following 5-point numerical differentiation formulas of equally spaced notes as examples for the first, second, third and fourth derivatives.

5-points:

$$
\begin{cases}
f'(x_0) = \dfrac{1}{12h}(-25f_0 + 48f_1 - 36f_2 + 16f_3 - 3f_4) + O(h^4), \\
f'(x_0) = \dfrac{1}{12h}(-3f_{-1} - 10f_0 + 18f_1 - 6f_2 + f_3) + O(h^4), \\
f'(x_0) = \dfrac{1}{12h}(f_{-2} - 8f_{-1} + 8f_1 - f_2) + O(h^4), \\
f'(x_0) = \dfrac{1}{12h}(-f_{-3} + 6f_{-2} - 18f_{-1} + 10f_0 + 3f_1) + O(h^4), \\
f'(x_0) = \dfrac{1}{12h}(3f_{-4} - 16f_{-3} + 36f_{-2} - 48f_{-1} + 25f_0) + O(h^4).
\end{cases}
\tag{5.1}
$$

$$
\begin{cases}
f''(x_0) = \dfrac{1}{12h^2}(35f_0 - 104f_1 + 114f_2 - 56f_3 + 11f_4) + O(h^3), \\
f''(x_0) = \dfrac{1}{12h^2}(11f_{-1} - 20f_0 + 6f_1 + 4f_2 - f_3) + O(h^3), \\
f''(x_0) = \dfrac{1}{12h^2}(-f_{-2} + 16f_{-1} - 30f_0 + 16f_1 - f_2) + O(h^4), \\
f''(x_0) = \dfrac{1}{12h^2}(-f_{-3} + 4f_{-2} + 6f_{-1} - 20f_0 + 11f_1) + O(h^3), \\
f''(x_0) = \dfrac{1}{12h^2}(11f_{-4} - 56f_{-3} + 114f_{-2} - 104f_{-1} + 35f_0) + O(h^3).
\end{cases}
\tag{5.2}
$$

$$\begin{cases} f'''(x_0) = \dfrac{1}{2h^3}(-5f_0 + 18f_1 - 24f_2 + 14f_3 - 3f_4) + O(h^2), \\ f'''(x_0) = \dfrac{1}{2h^3}(-3f_{-1} + 10f_0 - 12f_1 + 6f_2 - f_3) + O(h^2), \\ f'''(x_0) = \dfrac{1}{2h^3}(-f_{-2} + 2f_{-1} - 2f_1 + f_2) + O(h^2), \\ f'''(x_0) = \dfrac{1}{2h^3}(f_{-3} - 6f_{-2} + 12f_{-1} - 10f_0 + 3f_1) + O(h^2), \\ f'''(x_0) = \dfrac{1}{2h^3}(3f_{-4} - 14f_{-3} + 24f_{-2} - 18f_{-1} + 5f_0) + O(h^2). \end{cases} \tag{5.3}$$

$$\begin{cases} f^{(4)}(x_0) = \dfrac{1}{h^4}(f_0 - 4f_1 + 6f_2 - 4f_3 + f_4) + O(h), \\ f^{(4)}(x_0) = \dfrac{1}{h^4}(f_{-1} - 4f_0 + 6f_1 - 4f_2 + f_3) + O(h), \\ f^{(4)}(x_0) = \dfrac{1}{h^4}(f_{-2} - 4f_{-1} + 6f_0 - 4f_1 + f_2) + O(h^2), \\ f^{(4)}(x_0) = \dfrac{1}{h^4}(f_{-3} - 4f_{-2} + 6f_{-1} - 4f_0 + f_1) + O(h), \\ f^{(4)}(x_0) = \dfrac{1}{h^4}(f_{-4} - 4f_{-3} + 6f_{-2} - 4f_{-1} + f_0) + O(h). \end{cases} \tag{5.4}$$

From Corollaries 2.1 and 2.2 one can easily obtain more numerical differentiation formulas with more accurate approximation than those mentioned above. However, these example formulas showed here are simply to compare the new method directly with other finite difference methods of numerical differentiation. It can be easily found that these example formulas are the same as to those known corresponding numerical differentiation formulas based on interpolating polynomials (such as the Lagrange, Newton, Hermite interpolating polynomials, and etc.) [2,3,7–9,11,27,28], operators [7,33] and lozenge diagrams [10]. The new method in this paper is essentially based on the Taylor series expansion. In fact, different numerical differentiation formulas from interpolating polynomials, operators and lozenge diagrams are equivalent form of one of the finite difference formulas from Taylor series expansion [17,19]. As mentioned before, however, the forms based on interpolating polynomials, operators and lozenge diagrams are implicit and complicated, whereas the new method here has some important advantages. First, it gives explicit formulas that use given function values at sampling notes directly and easily to calculate numerical approximations of arbitrary order at any sampling data for the first and higher derivatives. Evidently, the explicit formulas are also handy for estimating the derivate of unequally spaced data. Second, the explicit difference formulas can be directly used for designing difference schemes of ODEs and PDEs and solving them. Third, the explicit formulas need less calculational burden, computing time and storage to estimate the derivatives than the other methods stated above.

Forward, backward and centered difference formulas are widely used to approximate derivatives in practice. The forward and backward difference formulas are useful for end-point approximations, particularly with regard to the clamped cubic spline interpolation. For evenly spaced data their general forms can be yielded as follows by use of Corollaries 2.1 and 2.2.

An $(n+1)$-point forward difference formula of order n to approximate first derivative of a function $f(x)$ at the left end-point x_0 can be expressed as

$$f'(x_0) = \frac{1}{h} \sum_{j=1}^{n} d_{n+1,0,j} f(x_j) + O_{n,0}(h^n), \tag{5.5}$$

where the coefficients

$$d_{n+1,0,j} = \frac{(-1)^{j-1}}{j} \binom{n}{j}, \quad j = 1, \ldots, n, \tag{5.6}$$

and

$$d_{n+1,0,0} = -\sum_{j=1}^{n} d_{n+1,0,j} = -\sum_{j=1}^{n} \frac{(-1)^{j-1}}{j} \binom{n}{j} = -\sum_{j=1}^{n} \frac{1}{j}. \tag{5.7}$$

An $(n+1)$-point backward difference formula of order n to estimate first derivative of a function $f(x)$ at the right end-point x_0 can be written as

$$f'(x_0) = \frac{1}{h} \sum_{j=-n}^{0} d_{n+1,0,j} f(x_j) + O_{n,0}(h^n), \tag{5.8}$$

where the coefficients

$$d_{n+1,0,-j} = \frac{(-1)^{j}}{j} \binom{n}{j} = -d_{n+1,0,j}, \quad j = 1, \ldots, n \tag{5.9}$$

and

$$d_{n+1,0,0} = -\sum_{j=-n}^{-1} d_{n+1,0,j} = \sum_{j=1}^{n} \frac{1}{j}. \tag{5.10}$$

An $(2n+1)$-point centered difference formula of order $2n$ to approximate first derivative of a function $f(x)$ at the middle point x_0 can be determined as

$$f'(x_0) = \frac{1}{h} \sum_{j=-n}^{n} d_{2n+1,0,j} f(x_j) + O_{2n,0}(h^{2n}), \tag{5.11}$$

where the coefficients

$$d_{2n+1,0,j} = \frac{(-1)^{j+1}(n!)^2}{j(n-j)!(n+j)!}, \quad j = \pm 1, \pm 2, \ldots, \pm n \tag{5.12}$$

and

$$d_{2n+1,0,0} = 0. \tag{5.13}$$

For $(2n+1)$ distinct points $x_0, x_1, \ldots, x_{2n}$, an $(2n+1)$-point centered difference formula to approximate mth derivative of a function $f(x)$ at the middle point x_n, where $m \leqslant 2n$, $m \geqslant 2$, can be written as

$$f^{(m)}(x_n) = \frac{1}{h^m} \sum_{j=0}^{2n} d^{(m)}_{2n+1,n,j}\, f(x_j) + O^{(m)}_{n,n}(h^{2n-m+1}), \tag{5.14}$$

where the coefficients

$$d^{(m)}_{2n+1,n,j} = \frac{(-1)^{m-j} m! a^{(m-1)}_{2n-1,n,j}}{j!(2n-j)!}, \quad j \neq n \tag{5.15}$$

$$d^{(m)}_{2n+1,n,n} = -\sum_{j=0,\, j\neq n}^{2n} d^{(m)}_{2n+1,n,j}, \tag{5.16}$$

where $a^{(m-1)}_{2n-1,n,j} = a_{m-1}(-n, \ldots, k-n, \ldots, n)$ ($k = 0, 1, \ldots, 2n$ and $k \neq n, j$).

Refs. [17–22] also provided the forward, backward and central difference formulas of the first derivative of a function, and the central difference approximations of higher derivatives for equally spaced data. Comparing them with the corresponding forward, backward and central difference formulas mentioned above, it could be found that they are equivalent. However, the new formulas in this paper do not limit themselves to these particular aspects, and possess more general and superior than the former for some reasons. First, evidently, the former formulas are only the three special cases of the new formulas of uniform grid-points. Moreover, the new method gives a unified form of expressions for the three particular formulas. Second, for the formers, the data had to be evenly spaced, whereas the new approach applies to both equally spaced data and unequally spaced data. Third, even for the case of equally spaced notes the formers do not generate the whole of question. For the $(N+1)$ given values of a function f at distinct points $x_0 < x_1 < \cdots < x_N$, for instance, the $(n+1)$-point forward difference formula of first derivative has to been applied to approximate of the first derivatives of the function at the left end-point x_0, but for the approximation of the first derivative at the point x_1 the forward difference formula does not adequately utilizes the known information on f since it does not use the information about f at the left adjacent point x_0. Notice that at this time we cannot use the $(n+1)$-point central difference formula for $n > 2$ since information on f outside the interval is unknown. As a consequence, its numerical accuracy is less than that of the $(n+1)$-point difference formula that uses the known information $f(x_0)$. For example, in the 5-point difference formula (5.1), the first formula is forward difference, and the second is neither forward nor central difference. Their errors are $(h^4/5)f^{(5)}(\xi)$ and $-(h^4/20)f^{(5)}(\xi)$, respectively. Although the errors in both formulae are of order 4, the error in the latter is approximately $\frac{1}{4}$ the error in the former. This point can be also seen from the following numerical results. The new formulas can avoid the case stated above and therefore may adequately employ known information of a function. Fourth, Refs. [17,21] gave the central difference approximations of higher derivatives for equally spaced data, based on some numerical results, but they did not show any mathematical proof. In contrast, the new approach is based on the strict algebraic proof. Moreover, the expressions of central difference approximations of higher derivatives (5.14) and (5.15) from the new technique are more simple and convenient than those of Refs. [17,21].

6. Numerical results

In this section we will present some numerical results to illustrate the performance of the new method. Two programs to test the performance of the method, respectively, for equally or unequally spaced data, were written in FORTRAN 77 and run on a SGI ORIGIN 2000 work-station with double precision of 16 significant digits. The basic algorithms are as follows:

Algorithm 6.1. For the evenly spaced points x_i $(i=0,1,\ldots,N)$, with the stepsize $h=x_{i+1}-x_i$, and given the function values $f(x_i)$ at x_i, if we use an $(n+1)$-point formula to approximate the mth derivative of $f(x)$ at x_i, let $K=[n/2]$, then there are four main steps:

Step 1: For $i,\ j=0:n$, compute $d^{(m)}_{n+1,i,j}$ by use of (2.31) and (2.32).

Step 2: For $i=0:K-1$, $f^{(m)}(x_i)=\frac{1}{h^m}\sum_{j=0}^{n} d^{(m)}_{n+1,i,j}f(x_j)$.

Step 3: For $i=K:N-K-1$, $f^{(m)}(x_i)=\frac{1}{h^m}\sum_{j=0}^{n} d^{(m)}_{n+1,K,j}f(x_{i-K+j})$.

Step 4: For $i=N-K:N$, $f^{(m)}(x_i)=\frac{1}{h^m}\sum_{j=0}^{n} d^{(m)}_{n+1,n+i-N,j}f(x_{N-n+j})$.

Note that the array d mentioned above is two-dimensional in the program.

Algorithm 6.2. For the unevenly spaced notes $x_0<x_1<\cdots<x_N$, and known the function values $f(x_i)$ at x_i $(i=0,1,\ldots,N)$, if one applies an $(n+1)$-point formula to estimate the mth derivative of $f(x)$ at x_i, let $K=[n/2]$, then three critical steps contain:

Step 1: For $i=0:K-1$, $j=0:n$ and $j\neq i$, calculate $a^{(m-1)}_{n-1,i,j}=a_{m-1}(\Delta_{i0},\ldots,\Delta_{ik},\ldots,\Delta_{in})$ $(k=0:n$ and $k\neq i,\ j)$ and

$$c^{(m)}_{n,i,j}=\frac{(-1)^{m-1}m!a^{(m-1)}_{n-1,i,j}}{\prod_{k=0,k\neq i,k\neq j}^{n}(x_k-x_j)},$$

then $f^{(m)}(x_i)=\sum_{j=0,j\neq i}^{n} c^{(m)}_{n,i,j}D(x_i,x_j)$.

Step 2: For $i=K:N-K-1$, $j=0:n$ and $j\neq K$, compute $a^{(m-1)}_{n-1,K,j}=a_{m-1}(\Delta_{i(i-K)},\ldots,\Delta_{i(i-K+k)},\ldots,\Delta_{i(i-K+n)})$ $(k=0:n$ and $k\neq K,\ j)$ and

$$c^{(m)}_{n,K,j}=\frac{(-1)^{m-1}m!a^{(m-1)}_{n-1,K,j}}{\prod_{k=0,k\neq K,k\neq j}^{n}(x_{i-K+k}-x_{i-K+j})},$$

then $f^{(m)}(x_i)=\sum_{j=0,j\neq K}^{n} c^{(m)}_{n,K,j}D(x_i,x_{i-K+j})$.

Step 3: For $i=N-K:N$, $j=0:n$ and $j\neq i+n-N$, calculate $a^{(m-1)}_{n-1,i+n-N,j}=a_{m-1}(\Delta_{i(N-n)},\ldots,\Delta_{i(N-n+k)},\ldots,\Delta_{iN})$ $(k=0:n$ and $k\neq i+n-N,\ j)$ and

$$c^{(m)}_{n,i+n-N,j}=\frac{(-1)^{m-1}m!a^{(m-1)}_{n-1,i+n-N,j}}{\prod_{k=0,k\neq i+n-N,k\neq j}^{n}(x_{i-K+k}-x_{i-K+j})},$$

then $f^{(m)}(x_i)=\sum_{j=0,j\neq i+n-N}^{n} c^{(m)}_{n,i+n-N,j}D(x_i,x_{N-n+j})$.

Table 1
Error in difference approximation to $f'(x)$ for equally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	1.82E − 07	2.67E − 09	1.49E − 09	4.94E − 11	8.15E − 12
1	0.03	3.03E − 08	4.09E − 10	1.86E − 10	5.57E − 12	8.21E − 13
2	0.06	1.21E − 08	1.45E − 10	5.28E − 11	1.41E − 12	1.82E − 13
3	0.09	9.06E − 09	9.26E − 11	2.63E − 11	6.16E − 13	6.93E − 14
4	0.12	8.87E − 09	1.40E − 10	2.10E − 11	4.20E − 13	3.64E − 14
5	0.15	8.58E − 09	1.85E − 10	2.01E − 11	4.88E − 13	2.75E − 14
6	0.18	8.21E − 09	2.29E − 10	1.92E − 11	4.97E − 13	3.82E − 14
7	0.21	7.75E − 09	2.33E − 10	2.38E − 11	7.55E − 13	6.62E − 14
8	0.24	1.02E − 08	3.96E − 10	4.73E − 11	1.78E − 12	1.71E − 13
9	0.27	2.54E − 08	1.21E − 09	1.65E − 10	7.19E − 12	8.10E − 13
10	0.30	1.51E − 07	8.64E − 09	1.31E − 09	6.57E − 11	7.85E − 12

Notice please that all of the symbols a, c, and D stated above are only one-dimensional arrays about the index j in the program.

The example function is

$$f(x) = x\mathrm{e}^{ax} + \sin bx, \tag{6.1}$$

where $a=-2$ and $b=3$. Then its first, second, third and fourth derivatives are $f'(x)=(1+a)x\mathrm{e}^{ax}+b\cos bx$, $f''(x)=a(2+a)x\mathrm{e}^{ax}-b^2\sin bx$, $f'''(x)=a^2(3+a)x\mathrm{e}^{ax}-b^3\cos bx$, and $f^{(4)}(x)=a^3(4+a)x\mathrm{e}^{ax}+b^4\sin bx$, respectively. Based on the algorithms mentioned above, we have carried out the $(n+1)$-point formulas, where n is from 1 to 10, to approximate the first, second, third and fourth derivatives of $f(x)$ for two cases, the equally and unequally spaced data. To save space, we only show the numerical results of the $(n+1)$-point formulas, where n is from 6 to 10. Tables 1–4, respectively, represent errors in difference approximation to the first, second, third and fourth derivatives of $f(x)$ for equally spaced data with the stepsize $h=0.03$. Tables 5–8 illustrate, respectively, errors in difference approximation to the first, second, third and fourth derivatives of $f(x)$ for unequally spaced data. From the tables, evidently, some conclusions can be given as follows:

(1) The new method performs well whether the data are equally spaced or unequally spaced, and whether the derivative is first or higher orders.
(2) In general, using more evaluation points produces greater accuracy. That is to say, increasing the order of numerical differentiation, reduces the error.
(3) Given n, the accuracies of the $(n+1)$-point forward and backward difference approximations are obviously less than those of the other $(n+1)$-point formulas. This is because the $(n+1)$-point forward and backward formulas use data on only one side of reference point and the other $(n+1)$-point formulas use data on both sides of reference point.

Table 2
Error in difference approximation to $f''(x)$ for equally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	2.97E − 05	4.49E − 07	2.70E − 07	9.25E − 09	1.58E − 09
1	0.03	2.59E − 06	3.77E − 08	1.97E − 08	6.33E − 10	9.96E − 11
2	0.06	4.72E − 07	6.99E − 09	3.36E − 09	1.02E − 10	1.50E − 11
3	0.09	1.19E − 09	1.19E − 09	7.96E − 10	2.48E − 11	3.30E − 12
4	0.12	1.99E − 09	1.99E − 09	5.40E − 12	5.40E − 12	9.15E − 13
5	0.15	2.76E − 09	2.76E − 09	6.52E − 12	6.52E − 12	3.48E − 13
6	0.18	3.50E − 09	3.50E − 09	7.39E − 12	7.39E − 12	1.31E − 12
7	0.21	4.19E − 09	4.19E − 09	7.05E − 10	3.20E − 11	3.53E − 12
8	0.24	3.92E − 07	2.12E − 08	2.98E − 09	1.31E − 10	1.45E − 11
9	0.27	2.16E − 06	1.19E − 07	1.74E − 08	8.31E − 10	9.71E − 11
10	0.30	2.46E − 05	1.50E − 06	2.37E − 07	1.24E − 08	1.51E − 09

Table 3
Error in difference approximation to $f'''(x)$ for equally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	2.74E − 03	4.31E − 05	2.92E − 05	1.05E − 06	1.95E − 07
1	0.03	8.21E − 05	7.93E − 07	2.20E − 08	6.83E − 09	2.22E − 09
2	0.06	9.40E − 05	1.06E − 06	3.23E − 07	7.61E − 09	7.84E − 10
3	0.09	8.22E − 05	8.49E − 07	2.30E − 07	5.14E − 09	5.47E − 10
4	0.12	8.05E − 05	1.28E − 06	1.99E − 07	3.99E − 09	3.59E − 10
5	0.15	7.79E − 05	1.69E − 06	1.91E − 07	4.64E − 09	3.00E − 10
6	0.18	7.45E − 05	2.08E − 06	1.82E − 07	4.69E − 09	3.35E − 10
7	0.21	7.03E − 05	2.11E − 06	2.08E − 07	6.21E − 09	5.10E − 10
8	0.24	8.00E − 05	2.73E − 06	2.92E − 07	9.17E − 09	7.75E − 10
9	0.27	7.10E − 05	1.35E − 06	3.45E − 09	1.12E − 08	2.10E − 09
10	0.30	2.25E − 03	1.53E − 04	2.54E − 05	1.43E − 06	1.84E − 07

7. Conclusion

General explicit finite difference formulas for numerical derivatives of unequally and equally spaced data are studied in this paper. Through introducing the generalized Vandermonde determinant, the linear algebraic system of a kind of Vandermonde equations is solved analytically by use of the basic properties of the determinant, and then combining with Taylor series general explicit finite difference formulas with arbitrary order accuracy for approximating first and higher derivates of a function are presented for unequally or equally spaced grid-points. A comparison with other implicit finite difference formulas

Table 4
Error in difference approximation to $f^{(4)}(x)$ for equally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	1.65E − 01	2.75E − 03	2.22E − 03	8.51E − 05	1.72E − 05
1	0.03	3.14E − 02	4.44E − 04	2.08E − 04	6.28E − 06	9.31E − 07
2	0.06	7.86E − 03	1.16E − 04	5.36E − 05	1.58E − 06	2.24E − 07
3	0.09	2.15E − 05	2.15E − 05	1.44E − 05	4.46E − 07	6.07E − 08
4	0.12	3.61E − 05	3.61E − 05	9.80E − 08	9.80E − 08	1.70E − 08
5	0.15	5.01E − 05	5.01E − 05	1.16E − 07	1.16E − 07	1.25E − 09
6	0.18	6.35E − 05	6.35E − 05	1.32E − 07	1.32E − 07	1.81E − 08
7	0.21	7.61E − 05	7.61E − 05	1.28E − 05	5.67E − 07	6.16E − 08
8	0.24	6.55E − 03	3.46E − 04	4.76E − 05	2.02E − 06	2.17E − 07
9	0.27	2.62E − 02	1.33E − 03	1.85E − 04	8.07E − 06	8.73E − 07
10	0.30	1.34E − 01	1.07E − 02	1.92E − 03	1.18E − 04	1.61E − 05

Table 5
Error in difference approximation to $f'(x)$ for unequally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	6.93E − 07	2.20E − 08	8.90E − 09	4.52E − 10	7.26E − 11
1	0.03	1.83E − 07	5.44E − 09	1.88E − 09	8.84E − 11	1.26E − 11
2	0.07	1.10E − 07	2.89E − 09	7.98E − 10	3.33E − 11	4.03E − 12
3	0.13	6.32E − 08	1.29E − 09	2.38E − 10	7.91E − 12	7.50E − 13
4	0.17	2.39E − 08	5.62E − 10	7.95E − 11	2.16E − 12	1.66E − 13
5	0.19	1.58E − 08	5.75E − 10	5.76E − 11	2.03E − 12	1.31E − 13
6	0.23	2.02E − 08	8.84E − 10	6.85E − 11	3.12E − 12	2.41E − 13
7	0.28	5.03E − 09	2.87E − 10	2.82E − 11	1.65E − 12	1.50E − 13
8	0.29	5.10E − 09	3.13E − 10	3.20E − 11	1.97E − 12	1.87E − 13
9	0.33	3.34E − 08	2.65E − 09	3.12E − 10	2.26E − 11	2.41E − 12
10	0.36	1.73E − 07	1.62E − 08	2.08E − 09	1.69E − 10	1.93E − 11

based on interpolating polynomials, operators and lozenge diagrams shows that the new explicit formulas are very easy to implement for numerical approximations of arbitrary order to first and higher derivatives of equally and unequally spaced data, and they need less computation time and storage and can be directly used for designing difference schemes of ODEs and PDEs. Comparing with the forward, backward and central difference formulas of the first derivative of a function provided in [17,19,22], they are only three special cases of the new formulas for equally spaced data, whereas the new formulas are also applicable to nonequispaced notes and even for some cases of uniform points they may employ known information of a function more sufficient than the former. Moreover, the central difference approximations

Table 6
Error in difference approximation to $f''(x)$ for unequally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	9.82E − 05	3.21E − 06	1.39E − 06	7.27E − 08	1.20E − 08
1	0.03	7.38E − 06	2.47E − 07	1.05E − 07	5.45E − 09	8.56E − 10
2	0.07	4.51E − 07	3.14E − 08	1.82E − 08	9.76E − 10	1.52E − 10
3	0.13	2.20E − 06	5.88E − 08	1.45E − 08	5.51E − 10	5.96E − 11
4	0.17	1.61E − 06	4.63E − 08	5.76E − 09	1.81E − 10	1.56E − 11
5	0.19	9.05E − 07	2.57E − 08	3.20E − 09	9.15E − 11	7.58E − 12
6	0.23	1.89E − 07	3.94E − 09	4.54E − 10	5.57E − 11	6.30E − 12
7	0.28	9.32E − 07	4.89E − 08	4.57E − 09	2.53E − 10	2.22E − 11
8	0.29	9.72E − 07	6.41E − 08	6.80E − 09	4.34E − 10	4.15E − 11
9	0.33	2.31E − 06	2.15E − 07	2.72E − 08	2.15E − 09	2.38E − 10
10	0.36	2.72E − 05	2.72E − 06	3.60E − 07	3.04E − 08	3.57E − 09

Table 7
Error in difference approximation to $f'''(x)$ for unequally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	7.42E − 03	2.54E − 04	1.23E − 04	6.74E − 06	1.16E − 06
1	0.03	8.50E − 04	2.37E − 05	6.75E − 06	2.77E − 07	3.13E − 08
2	0.07	4.31E − 04	1.14E − 05	2.94E − 06	1.16E − 07	1.27E − 08
3	0.13	2.16E − 04	3.74E − 06	4.30E − 07	7.96E − 09	3.87E − 12
4	0.17	1.79E − 04	3.40E − 06	5.79E − 07	1.31E − 08	7.87E − 10
5	0.19	1.36E − 04	5.61E − 06	5.20E − 07	2.03E − 08	1.24E − 09
6	0.23	1.07E − 04	4.87E − 06	3.84E − 07	1.71E − 08	1.28E − 09
7	0.28	3.87E − 05	3.40E − 06	3.93E − 07	2.67E − 08	2.62E − 09
8	0.29	3.35E − 05	7.79E − 07	4.41E − 09	5.27E − 09	8.62E − 10
9	0.33	1.13E − 04	5.68E − 06	4.31E − 07	8.01E − 09	7.96E − 10
10	0.36	2.37E − 03	2.63E − 04	3.66E − 05	3.28E − 06	3.98E − 07

of higher derivatives for equally spaced data were only presented in [17,21], without mathematical proof. In contrast, the new formulas are strictly proved and can give an equivalent but handier expression of central difference approximations of higher derivatives. Therefore, the new formulas are more general and superior. Basic computer algorithms of the new difference formulas of evenly and unevenly data are given for numerical approximations of any order to first and higher derivatives. Numerical results suggest that the new explicit difference formulae are very efficient for evaluating first and higher derivatives of equally and unequally spaced data.

Table 8
Error in difference approximation to $f^{(4)}(x)$ for unequally spaced data

i	$x(i)$	$(n+1)$-point formula				
		7-points	8-points	9-points	10-points	11-points
0	0.00	3.55E − 01	1.30E − 02	7.45E − 03	4.38E − 04	8.31E − 05
1	0.03	1.09E − 01	3.51E − 03	1.38E − 03	6.87E − 05	1.05E − 05
2	0.07	1.42E − 02	5.28E − 04	2.20E − 04	1.08E − 05	1.54E − 06
3	0.13	1.00E − 02	3.04E − 04	7.57E − 05	2.78E − 06	2.71E − 07
4	0.17	9.86E − 03	3.60E − 04	4.18E − 05	1.50E − 06	1.37E − 07
5	0.19	4.41E − 03	3.78E − 05	1.61E − 05	1.80E − 07	3.75E − 08
6	0.23	7.82E − 04	9.42E − 05	1.05E − 06	4.37E − 07	5.32E − 08
7	0.28	6.20E − 03	2.83E − 04	2.26E − 05	8.96E − 07	5.26E − 08
8	0.29	7.81E − 03	5.34E − 04	5.60E − 05	3.43E − 06	3.15E − 07
9	0.33	2.99E − 02	2.58E − 03	3.12E − 04	2.30E − 05	2.45E − 06
10	0.36	1.34E − 01	1.73E − 02	2.59E − 03	2.53E − 04	3.21E − 05

Acknowledgements

This work was supported jointly by the NSFC projects (Grant Nos. 40325015, 40221503, and 49905007).

References

[1] R.S. Anderssen, P. Bloomfield, Numerical differentiation procedures for non-exact data, Numer. Math. 22 (1973/74) 157–182.
[2] R.L. Burden, J.D. Faires, Numerical Analysis, seventh ed., Brooks/Cole, Pacific Grove, CA, 2001.
[3] S.C. Chapra, R.P. Canale, Numerical Methods for Engineers, third ed., McGraw-Hill, New York, 1998.
[4] L. Collatz, The Numerical Treatment of Differential Equations, third ed., Springer, Berlin, 1966.
[5] G. Corliss, C. Faure, A. Griewank, L. Hascoet, U. Naumann, Automatic Differentiation of Algorithms—From Simulation to Optimization, Springer, Berlin, 2002.
[6] J. Cullum, Numerical differentiation and regularization, SIAM J. Numer. Anal. 8 (2) (1971) 254–265.
[7] G. Dahlquist, A. Bjorck, Numerical Methods, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1974.
[8] T. Dokken, T. Lyche, A divided difference formula for the error in Hermite interpolation, BIT 19 (1979) 539–540.
[9] W. Forst, Interpolation und numerische differentiation, J. Approx. Theory 39 (2) (1983) 118–131.
[10] C.F. Gerald, P.O. Wheatley, Applied Numerical Analysis, fourth ed., Addison-Wesley Pub. Co., Reading, MA, 1989.
[11] L.P. Grabar, Numerical differentiation by means of Chebyshev polynomials orthonormalized on a system of equidistant points, Zh. Vychisl, Mat. i Mat. Fiz. 7 (6) (1967) 1375–1379.
[12] A. Griewank, Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation, Number 19 in Frontiers in Applied Mathematics, SIAM, Philadelphia, 2000.
[13] A. Griewank, F. Corliss, Automatic Differentiation of Algorithms—Theory, Implementation and Application, SIAM, Philadelphia, 1991.
[14] R.W. Hamming, Numerical Methods for Scientists and Engineers, McGraw-Hill, New York, 1962.
[15] M. Hanke, O. Scherzer, Inverse problems light—numerical differentiation, Amer. Math. Monthly 6 (2001) 512–522.
[16] M.T. Heath, Science Computing—An Introductory Survey, second ed., McGraw-Hill, New York, 1997.
[17] I.R. Khan, R. Ohba, Closed-form expressions for the finite difference approximations of first and higher derivatives based on Taylor series, J. Comput. Appl. Math. 107 (1999) 179–193.

[18] I.R. Khan, R. Ohba, Digital differentiators based on Taylor series, IEICE Trans. Fund. E82-A (12) (1999) 2822–2824.
[19] I.R. Khan, R. Ohba, New finite difference formulas for numerical differentiation, J. Comput. Appl. Math. 126 (2001) 269–276.
[20] I.R. Khan, R. Ohba, Mathematical proof of explicit formulas for tap-coefficients of Taylor series based FIR digital differentiators, IEICE Trans. Fund. E84-A (6) (2001) 1581–1584.
[21] I.R. Khan, R. Ohba, Taylor series based finite difference approximations of higher-degree derivatives, J. Comput. Appl. Math. 154 (2003) 115–124.
[22] I.R. Khan, R. Ohba, N. Hozumi, Mathematical proof of closed form expressions for finite difference approximations based on Taylor series, J. Comput. Appl. Math. 150 (2003) 303–309.
[23] T. King, D. Murio, Numerical differentiation by finite dimensional regularization, IMA J. Numer. Anal. 6 (1986) 65–85.
[24] I. Knowles, R. Wallace, A variational method for numerical differentiation, Numer. Math. 70 (1995) 91–110.
[25] E. Kreyzig, Advanced Engineering Mathematics, seventh ed., Wiley, New York, 1994.
[26] T.N. Krishnamurti, L. Bounoua, An Introduction to Numerical Weather Prediction Technique Techniques, CRC Press, Boca Raton, FL, 1995.
[27] B.I. Kvasov, Numerical differentiation and integration on the basis of interpolation parabolic splines, Chisl. Metody Mekh. Sploshn. Sredy 14 (2) (1983) 68–80.
[28] J.H. Mathews, K.D. Fink, Numerical Methods Using MATLAB, third ed., Prentice-Hall, Inc., Englewood Cliffs, NJ, 1999.
[29] D.A. Murio, The Mollification Method and the Numerical Solution of Ill-posed Problems, Wiley, New York, 1993.
[30] L.B. Rail, Automatic Differentiation—Techniques and Applications, Springer, Berlin, 1981.
[31] A.G. Ramm, On numerical differentiation, Mathem. Izvestija vuzov 11 (1968) 131–135.
[32] A.G. Ramm, Stable solutions of some ill-posed problems, Math. Meth. Appl. Sci. 3 (1981) 336–363.
[33] P. Silvester, Numerical formation of finite-difference operators, IEEE Trans. Microwave Theory Technol. MTT-18 (10) (1970) 740–743.
[34] A.N. Tikhonov, V.Y. Arsenin, Solutions of Ill-posed Problems, Wiley, New York, 1977.
[35] V.V. Vasin, Regularization of the problem of numerical differentiation, Matem. zap. Ural'skii univ. 7 (2) (1969) 29–33.
[36] V.V. Vershinin, N.N. Pavlov, Approximation of derivatives by smoothing splines, Vychisl. Sistemy 98 (1983) 83–91.
[37] G. Wahba, Spline Models for Observational Data, SIAM, Philadelphia, 1990.