未知目标函数之一阶数值微分公式验证与实践

张雨浓,郭东生,徐思洪,李海林
(中山大学 信息科学与技术学院,广东 广州 510275)

摘 要: 根据多项式插值理论,对于未知的目标函数,在离散采样点获取其对应的函数值后,即可构造 Lagrange 插值多项式以近似求得该未知函数的逼近表达式. 进而,对 Lagrange 插值多项式求一阶导数可得到该未知目标函数的多点一阶微分近似公式;即:等间距情况下的 2～16 个数据点的后向差分公式. 计算机数值实验进一步验证与表明:该用于未知目标函数一阶数值微分的多点公式可以取得较高的计算精度.

关键词: 未知目标函数;Lagrange 插值多项式;一阶导数;数值微分公式;计算精度
中图分类号: O241.4 **文献标志码**: A **文章编号**:1004-0366(2009)01-0013-06

Verification and Practice on First-Order Numerical Differentiation Formulas for Unknown Target Functions

ZHANG Yu-nong, GUO Dong-sheng, XU Si-hong, LI Hai-lin
(School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510275, China)

Abstract: Based on the polynomial-interpolation theory, the Lagrange interpolating polynomial could be constructed by using the corresponding discrete-time values of an unknown target-function. Then, the approximate first-order numerical-differentiation formulas could be derived in terms of those multiple sampling-nodes. Hence, the equally-spaced backward-difference formulas involving two to sixteen sampling-nodes. Experimental results verify that the relatively high computational-precision could be achieved by using these formulas, when estimating the numerical values of the first-order derivative of unknown target-functions are estimated.

Key words: unknown target function; Lagrange interpolating polynomial; first-order derivative; numerical differentiation formulas; computational accuracy

数值微分在数学分析及工程应用方面有着广泛的应用,常用来解决常微分方程组(ODE)和偏微分方程组(PDE)问题,而后者正是工程系统中常用的描述方法[1～4]. 其思想是用离散的采样数据来逼近未知函数在各点的微分值. 对于未知的映射关系(即,仅知道其离散采样数据点及其对应的函数值),计算其数值微分的方法通常有插值法(如 Lagrange 插值法、样条插值法等)、Taylor 级数展开法和有限差分法(如差分算子法、差分表法)[5～9]. 特别是对于有着广泛应用背景和实用性的有限差分类型的方法,人们对其研究兴趣与日俱增[6,9～11]. 简言之,Khan 和 Ohba 利用基于 Taylor 级数的待定系数法构造了 Vandermonde 线性方程组,从而可得显性的前向、后向和中间数值微分公式[7～9];而 Li 通过泛化的 Vandermonde 行列式推导得出任一插

收稿日期:2008-07-17
基金项目:国家自然科学基金(60775050);中山大学科研启动费、后备重点课题项目.

值点任意阶的显性差分公式[6]. 考虑到基于历史数据的一阶后向差分公式在数值微分中的重要性[1~4],重点研究并实验了 2～16 个数据点的一阶导数的等间距后向差分公式,并考虑到该研究工作的完备性,也给出了一种不同于上述文献[6～9]的简单明了的公式推导过程.

具体而言,推导了 Lagrange 插值多项式的一阶微分公式以近似原未知目标函数的一阶微分值,代入历史数据点值即可得到原未知目标函数的一阶微分估计值. 该生成的通用公式与文献[6～9]中所得到的公式具有一定的同质性,也因此说明了 Lagrange 插值法和 Taylor 级数法在求解数值微分时本质上皆为多项式逼近的原理. 另外,值得指出的是,有限差分公式包括(1)利用未来数据的前向差分、(2)利用历史数据的后向差分和(3)历史和未来数据均用的中间差分 3 种形式;而在实际应用中,通常是能且只能利用过去及当前的数据来估计与预测未来的变化趋势,因此基于历史数据的后向差分公式尤其具有较重要的实际工程意义(如在 PID 控制、图像跟踪等多方面均起着重要的作用)[12]. 目前有文献提供了 2、3 及 5 个数据点的后向差分公式[5],然而实际应用中往往要求达到更高的估计精度而需要用到更多数据点的后向差分公式. 鉴于其他多点差分公式尚未明确且完备地给出,这或许会给工程实践者带来诸多不便;因此在给出基于 Lagrange 插值法的多点等间距后向差分通项公式及其推导后,进一步明确和完备地给出了 2～16 个数据点的等间距后向差分公式及其实践与验证,以方便其他理论研究者和工程应用者的各种尝试与探讨.

1 基于历史数据的一阶数值微分公式

从多项式插值理论出发,通过构造 Lagrange 插值多项式 $L(x)$ 可以近似求得关于离散采样点及相应函数值的未知目标函数 $f(x)$ 的逼近表达式,即 $f(x) \approx L(x)$;同时,对 $L(x)$ 求一阶导数,可因此近似得到未知目标函数 $f(x)$ 的一阶数值微分公式,即 $f'(x) \approx L'(x)$. 在实际应用中,数字离散系统往往是等间距地采集数据[13],即 $x_{i+1} - x_i = h(i = 0,1,2,\cdots,n)$,其中 h 为采样间隔常数. 另外,考虑到一般系统都是利用过去以及当前的数据作为估计未来变化趋势的依据[12],后向差分公式因此更加显得具有较多实用价值,如下主要探讨等间距后向差分公式(通项公式及 2 ～ 16 点特定差分公式)[5].

定义 1 设未知目标函数 $f(x)$ 在离散采样数据点 x_i 处的函数值为 $f(x_i)$, $i = 0,1,2,\cdots,n$. 若存在 n 次多项式 $P_n(x) = a_0 + a_1 x + \cdots + a_n x^n$ 满足条件 $P_n(x_i) = f(x_i)$,则称 $P_n(x)$ 为未知目标函数 $f(x)$ 的 n 次插值多项式.

引理 1 如果 $x_0, x_1, x_2, \cdots, x_n$ 是采样区间 $[a,b]$ 上 $(n+1)$ 个互不相同的数据点,且未知目标函数 $f(x)$ 的 $(n+1)$ 阶导数连续(记 $f \in C^{n+1}[a,b]$),则对任一 $x \in [a,b]$,存在 $\xi \in (a,b)$,使得 $f(x) = P_n(x) + R_{n+1}(x)$,其中 $P_n(x)$ 为未知目标函数 $f(x)$ 的插值多项式, $R_{n+1}(x)$ 为余项且

$$R_{n+1}(x) := f^{(n+1)}(\xi)(x - x_0)(x - x_1)\cdots(x - x_n)/(n+1)!.$$

根据上述定义和引理,可以得到一个 n 次插值多项式 $P_n(x)$ 来逼近未知目标函数 $f(x)$,且通过对多项式 $P_n(x)$ 求一阶导数,即可近似求得未知目标函数 $f(x)$ 的一阶数值微分公式. 因此可推导得到如下引理[6~7].

引理 2 如果 $x_0, x_1, x_2, \cdots, x_n$ 是采样区间 $[a,b]$ 上 $(n+1)$ 个互不相同的数据点,且未知目标函数 $f(x)$ 的 $(n+1)$ 阶导数连续,则对任意 $x_i \in [a,b]$, $i = 0,1,\cdots,n$,有 $f'(x_i) = P'_n(x_i) + R'_{n+1}(x_i)$,其中 $R'_{n+1}(x_i)$ 为相应余项的一阶导数项. 当数据点为等间距分布时,可有如下 $f(x)$ 的一阶微分公式

$$f'(x_i) = \frac{1}{h}\sum_{j=0,j\neq i}^{n}\frac{(-1)^{i-j}\dfrac{i!(n-i)!}{j!(n-j)!}f(x_j) + f(x_i)}{(i-j)} + \frac{(-1)^{n-j}h^n f^{(n+1)}(\xi)}{(n+1)C_n^i}. \tag{1}$$

特别值得指出的是,对于 $f(x)$ 的一阶微分公式(1),当 $i = n$ 时, 忽略余项[即公式(1) 最右端项],可以近似得到未知目标函数 $f(x)$ 一阶导数的等间距后向差分估计公式:

$$f'(x_n) \approx \frac{1}{h}\sum_{j=0}^{n-1}\frac{(-1)^{n-j}C_n^j f(x_i) + f(x_n)}{(n-j)}. \tag{2}$$

证明 给出上述定理的推导过程及 2 ～ 16 个数据点的后项差分公式. 不同于文献[6 ～ 9] 给出的一阶数值微分公式推导,可以通过构造 Lagrange 插值多项式并求其一阶导数,从而推得多点一阶数值微分公式.

首先,根据文献[5]n次Lagrange插值多项式 $L_n(x)$ 可写为

$$L_n(x) = f(x_0)\,l_0(x) + f(x_1)\,l_1(x) + \cdots + f(x_n)\,l_n(x) = \sum_{i=0}^{n} f(x_i)\,l_i(x), \tag{3}$$

其中 $l_0(x), l_1(x), \cdots, l_n(x)$ 为 n 次插值基函数,其表达式如下:

$$l_i(x) = \frac{(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)} = \prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j}.$$

引入记号

$$w(x) = (x-x_0)(x-x_1)\cdots(x-x_n) = \prod_{i=0}^{n}(x-x_i),$$

则易得到 $w(x)$ 的一阶导数为

$$w'(x) = \sum_{i=0}^{n}(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n).$$

基函数 $l_i(x)$ 因此简记为

$$l_i(x) = \prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j} = \frac{w(x)}{(x-x_i)\,w'(x_i)},$$

公式(3)可以改写为

$$L_n(x) = \sum_{i=0}^{n} f(x_i)\,l_i(x) = \sum_{i=0}^{n} f(x_i)\,\frac{w(x)}{(x-x_i)\,w'(x_i)}. \tag{4}$$

再令

$$w_i(x) = w(x)/(x-x_i),\ w_{i,j}(x) = w(x)/((x-x_i)(x-x_j))$$

且 $i \neq j$,则 $w_{i,j}(x)/w_i(x) = 1/(x-x_j)$ 且 $i \neq j$;这样,可以得到 $w_i(x_i) = w'(x_i)$. 由定理1可知

$$f'(x_i) = P_n'(x_i) + R_{n+1}'(x_i).$$

为求 $L_n(x)$ 在 x_i 处的一阶微分值,可先将 $L_n(x)$ 改写成

$$L_n(x) = \left(\sum_{\substack{j=0\\ j\neq i}}^{n} f(x_j)\,\frac{w(x)}{(x-x_j)\,w'(x_j)}\right) + f(x_i)\,\frac{w_i(x)}{w'(x_i)}.$$

对上述 $L_n(x)$ 求关于 x 的一阶导数,可得

$$L_n'(x) = \sum_{\substack{j=0\\ j\neq i}}^{n} f(x_j)\,\frac{w'(x)(x-x_j) - w(x)}{(x-x_j)^2 w'(x_j)} + f(x_i)\,\frac{w_i'(x)}{w'(x_i)},$$

考虑到 $w(x_i) = 0 (i = 0,1,2,\cdots,n)$ 且 $w_i'(x) = \sum_{\substack{j=0\\ j\neq i}}^{n} w_{i,j}(x)$,可以进一步推得

$$L_n'(x) = \sum_{\substack{j=0\\ j\neq i}}^{n} f(x_j)\left(\frac{w'(x)}{(x-x_j)\,w'(x_j)} - \frac{w(x)}{(x-x_j)^2 w'(x_j)}\right) + f(x_i)\,\frac{\sum_{\substack{j=0\\ j\neq i}}^{n} w_{i,j}(x)}{w'(x_i)}. \tag{5}$$

将 x_i 代入式(5)可得

$$L_n'(x_i) = \sum_{\substack{j=0\\ j\neq i}}^{n} f(x_j)\,\frac{w'(x_i)}{(x_i-x_j)\,w'(x_j)} + f(x_i)\sum_{\substack{j=0\\ j\neq i}}^{n}\frac{1}{x_i-x_j}.$$

当 x_i 以等间距分布,即 $x_{i+1} - x_i = h (i = 0,1,2,\cdots,n-1)$,则可由此得到等间距差分公式如下:

$$L_n'(x_i) = \frac{1}{h}\sum_{\substack{j=0\\ j\neq i}}^{n}\frac{(-1)^{i-j}\,\frac{i!(n-i)!}{j!(n-j)!}f(x_j) + f(x_i)}{i-j}.$$

此时根据引理1可知,未知目标函数 $f(x)$ 的一阶导数估计的余项为

$$R_{n+1}'(x_i) = \frac{(-1)^{n-i}h^n f^{(n+1)}(\xi)}{(n+1)\,C_n^i},$$

因此,可得

$$f'(x_i) = P'_n(x_i) + R'_{n+1}(x_i) = L_n'(x_i) + R'_{n+1}(x_i),$$

即公式(1)所示. 特别地,当 $i = n$ 时,忽略余项 $R'_{n+1}(x_i)$,即可推得未知目标函数 $f(x)$ 一阶导数的等间距后向差分近似公式如公式(2)所示. 引理 2 证明完毕.

对于公式(2),分别令 $n = 1, 2, \cdots, 15$,经过计算,即可推得 2～16 个数据点的等间距后向差分公式,以下列出 2～16 个数据点的等间距后向差分公式,以方便其他研究者和应用者的各种尝试与探讨.

$$f'(x_1) \approx (f(x_1) - f(x_0))/h,$$

$$f'(x_2) \approx (3f(x_2) - 4f(x_1) + f(x_0))/2h,$$

$$f'(x_3) \approx (11f(x_3) - 18f(x_2) + 9f(x_1) - 2f(x_0))/6h,$$

$$f'(x_4) \approx (25f(x_4) - 48f(x_3) + 36f(x_2) - 16f(x_1) + 3f(x_0))/12h,$$

$$f'(x_5) \approx (137f(x_5) - 300f(x_4) + 300f(x_3) - 200f(x_2) + 75f(x_1) - 12f(x_0))/60h,$$

$$f'(x_6) \approx (147f(x_6) - 360f(x_5) + 450f(x_4) - 400f(x_3) + 225f(x_2) - 72f(x_1) + 10f(x_0))/60h,$$

$$f'(x_7) \approx (1\,089f(x_7) - 2\,940f(x_6) + 4\,410f(x_5) - 4\,900f(x_4) + 3\,675f(x_3) - 1\,764f(x_2) + 490f(x_1) - 60f(x_0))/420h,$$

$$f'(x_8) \approx (2\,283f(x_8) - 6\,720f(x_7) + 11\,760f(x_6) - 15\,680f(x_5) + 14\,700f(x_4) - 9\,408f(x_3) + 3\,920f(x_2) - 960f(x_1) + 105f(x_0))/840h,$$

$$f'(x_9) \approx (7\,129f(x_9) - 22\,680f(x_8) + 45\,360f(x_7) - 70\,560f(x_6) + 79\,380f(x_5) - 63\,504f(x_4) + 35\,280f(x_3) - 12\,960f(x_2) + 2\,835f(x_1) - 280f(x_0))/2\,520h,$$

$$f'(x_{10}) \approx (7\,381f(x_{10}) - 25\,200f(x_9) + 56\,700f(x_8) - 100\,800f(x_7) + 132\,300f(x_6) - 127\,008f(x_5) + 88\,200f(x_4) - 43\,200f(x_3) + 14\,175f(x_2) - 2\,800f(x_1) + 252f(x_0))/2\,520h,$$

$$f'(x_{11}) \approx (83\,711f(x_{11}) - 304\,920f(x_{10}) + 762\,300f(x_9) - 1\,524\,600f(x_8) + 2\,286\,900f(x_7) - 2\,561\,328f(x_6) + 2\,134\,440f(x_5) - 1\,306\,800f(x_4) + 271\,725f(x_3) - 169\,400f(x_2) + 30\,492f(x_1) - 2\,520f(x_0))/27\,720h,$$

$$f'(x_{12}) \approx (86\,021f(x_{12}) - 332\,640f(x_{11}) + 914\,760f(x_{10}) - 2\,032\,800f(x_9) + 3\,430\,350f(x_8) - 4\,390\,848f(x_7) + 4\,268\,880f(x_6) - 3\,136\,320f(x_5) + 1\,715\,175f(x_4) - 677\,600f(x_3) + 183\,952f(x_2) - 30\,240f(x_1) + 2\,310f(x_0))/27\,720h,$$

$$f'(x_{13}) \approx (1\,145\,993f(x_{13}) - 4\,684\,680f(x_{12}) + 14\,054\,040f(x_{11}) - 34\,354\,320f(x_{10}) + 64\,414\,350f(x_9) - 92\,756\,664f(x_8) + 103\,062\,960f(x_7) - 88\,339\,680f(x_6) + 57\,972\,915f(x_5) - 28\,628\,600f(x_4) + 10\,306\,296f(x_3) - 2\,555\,280f(x_2) + 393\,390f(x_1) - 27\,720f(x_0))/360\,360h,$$

$$f'(x_{14}) \approx (1\,171\,733f(x_{14}) - 5\,045\,040f(x_{13}) + 16\,396\,380f(x_{12}) - 43\,723\,680f(x_{11}) + 90\,180\,090f(x_{10}) - 144\,288\,144f(x_9) + 180\,360\,180f(x_8) - 176\,679\,360f(x_7) + 135\,270\,135f(x_6) - 80\,160\,080f(x_5) + 36\,072\,036f(x_4) - 11\,924\,640f(x_3) + 2\,732\,730f(x_2) - 388\,080f(x_1) + 25\,740f(x_0))/360\,360h,$$

$$f'(x_{15}) \approx (1\,195\,757f(x_{15}) - 5\,405\,400f(x_{14}) + 18\,918\,900f(x_{13}) - 54\,654\,600f(x_{12}) + 122\,972\,850f(x_{11}) - 216\,432\,216f(x_{10}) + 300\,600\,300f(x_9) - 331\,273\,800f(x_8) + 289\,864\,575f(x_7) - 200\,400\,200f(x_6) + 108\,216\,108f(x_5) - 44\,717\,400f(x_4) + 13\,663\,650f(x_3) - 2\,910\,600f(x_2) + 386\,100f(x_1) - 24\,024f(x_0))/360\,360h.$$

2 计算机验证与实践

为验证上述计算公式的精度与有效性,选取目标函数 $f(x) = x^3(e^x\cos(x) + x)$ 来进行计算机数值实验,欲求取目标函数 $f(x)$ 在测试节点 $x = 0.5$ 处的一阶导数值. 考虑各种间距 h 值,使用上述多点后向差分公式的数值计算误差如表 1 所示.

从表 1 各种不同情况下的计算误差的比较,可以发现:在间距相对较大(如 ≥ 0.2)时,差分公式应选取更多历史数据点为宜(截断误差就会减小),这样计算出的一阶导数精度较高;而间距相对较小(如 $< 5 \times$

10^{-6}) 时,因为舍入误差的缘故,选取较多历史数据点反而会降低差分公式的计算精度(此时也有差分公式系数较大的原因). 总而言之,通过优化间距值和使用合适点数的后向差分公式,可以得到对目标函数一阶导数的高精度估计(如计算误差介于 10^{-8} ～ 10^{-14}).

为了更全面地考察等间距后向差分公式的计算精度,同时也对另外一个目标函数 $f(x) = \sin(x)$ 进行了计算机数值实验(测试节点仍然是 $x = 0.5$). 计算结果如表 2 所示,结论也与上述分析相一致;即,可以达到较高的估计精度(介于 10^{-8} ～ 10^{-14}).

表 1　目标函数 $f(x) = x^3(e^x\cos(x) + x)$ 在利用 2～8 个历史数据点的一阶微分计算误差

h	2	3	4	5	6	7	8
0.500	1.18	9.41×10^{-1}	1.32	3.09×10^{-1}	5.07×10^{-3}	2.33×10^{-1}	3.12×10^{-1}
0.200	6.65×10^{-1}	2.68×10^{-1}	6.99×10^{-2}	1.90×10^{-2}	1.68×10^{-2}	7.38×10^{-3}	2.01×10^{-3}
0.100	3.69×10^{-1}	7.30×10^{-2}	5.28×10^{-3}	2.73×10^{-3}	8.92×10^{-4}	1.63×10^{-4}	1.49×10^{-5}
0.050	1.94×10^{-1}	1.86×10^{-2}	3.31×10^{-4}	2.36×10^{-4}	3.47×10^{-5}	2.84×10^{-6}	8.53×10^{-8}
0.020	7.98×10^{-2}	3.00×10^{-3}	5.83×10^{-6}	7.20×10^{-6}	4.01×10^{-7}	1.21×10^{-8}	8.60×10^{-11}
0.010	4.03×10^{-2}	7.50×10^{-4}	2.46×10^{-8}	4.76×10^{-7}	1.30×10^{-8}	1.91×10^{-10}	4.74×10^{-13}
0.005	2.02×10^{-2}	1.87×10^{-4}	4.25×10^{-8}	3.06×10^{-8}	4.14×10^{-10}	2.91×10^{-12}	1.14×10^{-13}
0.002	8.11×10^{-3}	3.00×10^{-5}	4.50×10^{-9}	7.95×10^{-10}	4.37×10^{-12}	1.52×10^{-13}	1.40×10^{-14}
0.001	4.06×10^{-3}	7.49×10^{-6}	6.37×10^{-10}	5.01×10^{-11}	1.52×10^{-13}	4.10×10^{-14}	1.52×10^{-13}
5×10^{-4}	2.03×10^{-3}	1.87×10^{-6}	8.42×10^{-11}	2.93×10^{-12}	4.10×10^{-14}	4.10×10^{-14}	1.51×10^{-12}
2×10^{-4}	8.13×10^{-4}	3.00×10^{-7}	4.70×10^{-12}	8.47×10^{-13}	2.92×10^{-13}	8.47×10^{-13}	2.51×10^{-12}
10^{-4}	4.06×10^{-4}	7.49×10^{-8}	8.18×10^{-13}	2.48×10^{-12}	4.70×10^{-12}	8.47×10^{-13}	1.80×10^{-11}
5×10^{-5}	2.03×10^{-4}	1.87×10^{-8}	8.47×10^{-13}	3.07×10^{-12}	3.59×10^{-12}	8.47×10^{-13}	3.07×10^{-12}
2×10^{-5}	8.13×10^{-5}	2.99×10^{-9}	5.26×10^{-12}	2.48×10^{-12}	2.47×10^{-11}	8.03×10^{-11}	3.58×10^{-11}
10^{-5}	4.06×10^{-5}	7.41×10^{-10}	1.36×10^{-11}	1.42×10^{-12}	3.07×10^{-12}	3.07×10^{-12}	1.36×10^{-10}
5×10^{-6}	2.03×10^{-5}	1.58×10^{-10}	4.75×10^{-11}	3.64×10^{-11}	5.86×10^{-11}	3.64×10^{-11}	4.36×10^{-10}
2×10^{-6}	8.13×10^{-6}	5.30×10^{-11}	5.30×10^{-11}	1.36×10^{-10}	8.08×10^{-11}	2.53×10^{-11}	2.53×10^{-11}
10^{-6}	4.06×10^{-6}	1.97×10^{-10}	8.08×10^{-11}	8.08×10^{-11}	8.08×10^{-11}	5.25×10^{-10}	1.19×10^{-9}

表 2　目标函数 $f(x) = \sin(x)$ 在利用 2～8 个历史数据点的一阶微分计算误差

h	2	3	4	5	6	7	8
0.500	8.13×10^{-2}	8.13×10^{-2}	3.02×10^{-3}	1.14×10^{-2}	2.67×10^{-3}	1.44×10^{-3}	7.89×10^{-4}
0.200	4.19×10^{-2}	1.25×10^{-2}	5.12×10^{-4}	3.14×10^{-4}	3.78×10^{-6}	9.05×10^{-6}	1.93×10^{-7}
0.100	2.25×10^{-2}	3.03×10^{-3}	9.26×10^{-5}	1.89×10^{-5}	4.69×10^{-7}	1.39×10^{-7}	2.34×10^{-9}
0.050	1.16×10^{-2}	7.46×10^{-4}	1.33×10^{-5}	1.14×10^{-6}	1.99×10^{-8}	2.08×10^{-9}	3.30×10^{-11}
0.020	4.74×10^{-3}	1.18×10^{-4}	9.16×10^{-7}	2.86×10^{-8}	2.35×10^{-10}	8.28×10^{-12}	6.20×10^{-14}
0.010	2.38×10^{-3}	2.94×10^{-5}	1.17×10^{-7}	1.77×10^{-9}	7.65×10^{-12}	1.71×10^{-13}	1.39×10^{-13}
0.005	1.19×10^{-3}	7.33×10^{-6}	1.48×10^{-8}	1.10×10^{-10}	1.49×10^{-13}	2.80×10^{-14}	2.95×10^{-13}
0.002	4.79×10^{-4}	1.17×10^{-6}	9.55×10^{-10}	2.85×10^{-12}	7.30×10^{-14}	3.80×10^{-14}	1.84×10^{-13}
0.001	2.40×10^{-4}	2.93×10^{-7}	1.20×10^{-10}	5.17×10^{-13}	2.95×10^{-13}	2.95×10^{-13}	2.95×10^{-13}
5×10^{-4}	1.20×10^{-4}	7.31×10^{-8}	1.50×10^{-11}	7.30×10^{-14}	5.17×10^{-13}	5.17×10^{-13}	3.18×10^{-12}
2×10^{-4}	4.79×10^{-5}	1.17×10^{-8}	8.50×10^{-13}	2.95×10^{-13}	1.41×10^{-12}	1.93×10^{-12}	2.95×10^{-13}
10^{-4}	2.40×10^{-5}	2.93×10^{-9}	2.95×10^{-13}	1.41×10^{-12}	3.04×10^{-12}	3.04×10^{-12}	3.04×10^{-12}
5×10^{-5}	1.20×10^{-5}	7.31×10^{-10}	1.41×10^{-12}	8.07×10^{-12}	3.63×10^{-12}	8.07×10^{-12}	2.14×10^{-11}
2×10^{-5}	4.79×10^{-6}	1.18×10^{-10}	1.93×10^{-12}	1.47×10^{-11}	7.48×10^{-12}	3.63×10^{-12}	3.69×10^{-11}
10^{-5}	2.40×10^{-6}	2.96×10^{-11}	1.47×10^{-11}	4.80×10^{-11}	1.86×10^{-11}	3.63×10^{-12}	4.80×10^{-11}
5×10^{-6}	1.20×10^{-6}	1.86×10^{-11}	3.63×10^{-12}	3.63×10^{-12}	2.58×10^{-11}	1.86×10^{-11}	6.30×10^{-11}
2×10^{-6}	4.80×10^{-7}	7.48×10^{-12}	4.80×10^{-11}	1.04×10^{-10}	4.80×10^{-11}	6.30×10^{-11}	4.80×10^{-11}
10^{-6}	2.40×10^{-7}	6.30×10^{-11}	4.80×10^{-11}	4.80×10^{-11}	1.59×10^{-10}	5.07×10^{-10}	1.27×10^{-9}

3 结论

一阶数值微分在各个科研领域都有着广泛的应用,如对常微分方程组(ODE)问题的近似求解. 插值法是数值微分方法中较为方便和有效的一种,利用插值法可较容易求得多点有限差分公式,其中包括前向差分、后向差分以及中间差分 3 种形式. 在实际应用中,基于历史数据的后向差分公式往往具有更为重要的使用价值和意义. 当前 2、3 及 5 个数据点的后向差分公式较为常见,但其他数目的多点差分公式及其更细致深入的实践验证尚未在文献中明确列出,给工程应用者带来不便. 基于多项式插值理论,构造了 Lagrange 插值多项式并对其求一阶导数,以求得未知目标函数一阶导数的等间距后向差分公式. 计算机数值实验进一步验证与表明,该多点等间距后向差分公式可以取得较好的一阶导数估计精度,这对于 PID 控制、图像跟踪等方面的应用或有着一定重要的意义. 值得指出的是,有限的实践也表明了这些数值微分公式对噪声较敏感. 因篇幅所限,将另外探讨.

参考文献:

[1] Zhang Y N, Jiang D C, Wang J. A recurrent Neural Network for Solving Sylvester Equation with Time-varying Coefficients[J]. IEEE Transactions on Neural Networks, 2002, 13(5): 1 053-1 063.

[2] Zhang Y N, Ge S S. Design and Analysis of a General Recurrent Neural Network Model for Time-varying Matrix Inversion[J]. IEEE Transactions on Neural Networks, 2005, 16(6): 1 477-1 490.

[3] Zhang Y N, Chen K. Comparison on Zhang Neural Network and Gradient Neural Network for Time-varying Linear Matrix Equation AXB = C Solving[A]. Proceedings of International Conference on Industrial Technology[C]. Chengdu, China, 2008.

[4] Ma W M, Zhang Y N, Wang J. MATLAB Simulink Modeling and Simulation of Zhang Neural Networks for Online Time-varying Sylvester Equation Solving[A]. Proceedings of International Joint Conference on Neural Networks[C], Hong Kong, China, 2008: 286-290.

[5] Mathews J H, Fink K D. Numerical Methods Using MATLAB[M]. Beijin: Electlonic Industry Press, 2005.

[6] Li J P. General Explicit Difference Formulas for Numerical Differentiation[J]. Journal of Computational and Applied Mathematics, 2005, 183: 29-52.

[7] Khan I R, Ohba R. Close-form Expressions for the Finite Difference Approximations of First and Higher Derivatives Based on Taylor Series[J]. Journal of Computational and Applied Mathematics, 1999, 107: 179-193.

[8] Khan I R, Ohba R. New Finite Difference Formulas for Numerical Differentiation[J]. Journal of Computational and Applied Mathematics, 2000, 126: 269-276.

[9] Khan I R, Ohba R, Hozumi N. Mathematical Proof of Closed Form Expressions for Finite Difference Approximations Based on Taylor Series[J]. Journal of Computational and Applied Mathematics, 2003, 150: 303-309.

[10] 孙建平,赵亚红,梁若筠. 二阶差分方程边值问题的多解性[J]. 甘肃科学学报, 2002, 14(2): 1-5.

[11] 田瑞,李明,滑斌. 有限差分法求解方同轴线的特性[J]. 甘肃科学学报, 2007, 19(1): 111-113.

[12] 张雨浓,陈扬文,刘巍,等. 改造前向神经网络结构以求网络权值直接确定-Jacobi 正交基神经网络实例[J]. 自动化与信息工程, 2008, 29(1): 1-5.

[13] 李中华,张雨浓. 自动控制原理与设计[M]. 北京: 人民邮电出版社, 2007.

作者简介:

张雨浓 (1973-) 男,河南省信阳人,2002 年毕业于香港中文大学机械与自动化工程专业,获博士学位,现任中山大学信息科学与技术学院博士生导师,研究方向为神经网络、机器人、高斯过程及其科学计算与优化.