

引文格式:

孙鹏, 王云鹏, 吴琼, 等. 纳皮尔对数的数学证明及精度分析 [J]. 集成技术, 2024, 13(2): 3-14.

Sun P, Wang YP, Wu Q, et al. Proof of Napier's log and analysis of precision [J]. Journal of Integration Technology, 2024, 13(2): 3-14.

纳皮尔对数的数学证明及精度分析

孙鹏^{1*} 王云鹏¹ 吴琼¹ 宋德王¹ 张小飞² 杜娟² 斯白露³ 李慧云⁴

¹(百度自动驾驶技术部 北京 100085)

²(百度技术管理部 北京 100085)

³(北京师范大学系统科学学院 北京 100875)

⁴(中国科学院深圳先进技术研究院 深圳 518055)

摘要 对数函数在自动驾驶系统中有着广泛应用, 如自动驾驶感知系统所使用的深度学习或卷积神经网络通常会利用对数函数来设计损失函数, 故研究对数发明的历史对掌握对数的概念和应用具有重要意义。该文阐述了纳皮尔对数的定义及其 3 张表, 分析了前人的两类证明方法, 提出了新的基于指数函数构造的证明方法。同时, 该文还分析了纳皮尔的计算方法, 与对照方法相比, 给出了纳皮尔对精度范围的优化结果, 通过 MPRF 库进行了计算, 结果表明纳皮尔的方法的计算结果更接近真实值。

关键词 科学史; 纳皮尔; 对数; 自动驾驶; 深度学习

中图分类号 TP 391 文献标志码 A doi: 10.12146/j.issn.2095-3135.20230817001

Proof of Napier's Log and Analysis of Precision

SUN Peng^{1*} WANG Yunpeng¹ WU Qiong¹ SONG Dewang¹ ZHANG Xiaofei²
DU Juan² SI Bailu³ LI Huiyun⁴

¹(Department of Autonomous Driving Technology, Baidu Inc., Beijing 100085, China)

²(Department of Technology Management, Baidu Inc., Beijing 100085, China)

³(School of Systems Science, Beijing Normal University, Beijing 100875, China)

⁴(Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China)

*Corresponding Author: sunpeng14@baidu.com

Abstract In the automatic driving systems, logarithm function has been widely used. For example, logarithm function is often used to design loss function in deep learning or convolutional neural network, which serves as

收稿日期: 2023-08-17 修回日期: 2023-10-23

作者简介: 孙鹏(通讯作者), 博士, 研究方向为计算机系统结构、自动驾驶、高精地图与定位、空间数据库系统和数据挖掘、科学史等, E-mail: sunpeng14@baidu.com; 王云鹏, 博士, 研究方向为自动驾驶系统、高精地图与定位、计算机图形学等; 吴琼, 高级工程师, 研究方向为自动驾驶评测、仿真等; 宋德王, 硕士, 研究方向为环境感知、仿真; 张小飞, 博士, 研究方向为自动驾驶感知; 杜娟, 硕士, 研究方向为自动驾驶开源开放技术; 斯白露, 教授, 研究方向为计算神经科学、智能机器人、类脑智能; 李慧云, 研究员, 研究方向为自动驾驶、智能网联汽车等。

the basis for the automatic driving perception system. Therefore, studying the history of invention of logarithm is of great significance to master the concept and application. This paper studies the definition of Napier's logarithm and his three tables, analyzes two kinds of proof methods of predecessors, and puts forward new proof methods based on the exponential function. Meanwhile, this paper also analyzes Napier's calculation method. Compared with other alternative methods, the optimization results of Napier's interval approximation are given. The calculation by MPRF library shows that Napier's method is more convergent to the true value.

Keywords history of science; Napier; logarithm; autonomous driving; deep learning

1 引 言

在自动驾驶系统中，对数函数获得了广泛的应用。例如：在自动驾驶感知系统中，一些经典的目标检测、分割方法，如 YOLO^[1-4]、SSD^[5]、Faster R-CNN^[6]、CornerNet^[7]、RefineDet^[8]、PointNet^[9]、VoxelNet^[10]、SegNet^[11]、Mask R-CNN^[12]等，均为基于深度学习或神经网络的方法，而在神经网络中常会利用对数函数来设计损失函数^[13]。又例如：自动驾驶系统会使用到高精地图或导航地图^[14-16]，而在编绘地图时，需要将地球椭球面上的点表示到平面上，形成平面图。墨卡托投影是常用的地图投影之一，其将对数作为计算的数学基础^[17]。故研究对数的科学史对于掌握对数的概念和应用具有重要意义，已有大量的文献对其进行了研究^[18-22]。

苏格兰数学家约翰·纳皮尔(John Napier, 1550—1617)是对数的早期发明人之一，他有两本著作，分别是 1614 年出版的《论述对数的奇迹》(*Mirifici Logarithmorum Canonis Descriptio*, 以下简称 *Descriptio*)^[23]和 1619 年出版的《作出对数的奇迹》(*Mirifici Logarithmorum Canonis Constructio*, 以下简称 *Constructio*)^[24]。他在第一本书 *Descriptio* 中给出了对数的描述及如何使用三角对数表，在第二本书 *Constructio* 中，他给出了发明对数表背后的方法。

给出纳皮尔对数严格的数学表达式对理解纳皮尔对数具有重要意义，常见的有 C. H. Edwards 和 Denis Roegel 对纳皮尔对数的数学证明^[25-26]。C. H. Edwards 的证明基于速度和余下的距离相等，从 *Constructio* 中第 25 条的论断或推论出发，而不是从纳皮尔对数的定义出发，这使得证明与纳皮尔的定义有所偏离。Denis Roegel 的证明把纳皮尔定义中速度成几何比率下降的定义转换为余下的距离成几何比率下降，并依此作为证明的基础，同样使得该证明和纳皮尔的定义有所偏离。而本文直接利用指数函数来定义速度，尽管指数概念的出现晚于对数^[27]，但是作为证明的基础，和纳皮尔对数的定义是契合的，对于教学而言，更易于学生理解。

纳皮尔在计算他的 3 张对数表时，没有仅仅计算表中对数的单一估计值，而是使用了对数值的范围的估计，或区间估计(interval approximation)^[25]，他的这一思想贯穿在 3 张对数表的计算过程中。例如：在计算他的第 3 张表的第 0 列第 1 个数(9 995 000.000 0)的对数值时，就没有直接使用相邻的第 2 张表中的最后一个数(9 995 001.224 80)进行求解，而是首先使用 *Constructio* 中第 36 条的论断，即“若两组数的比相同，则它们的对数的差相同”，然后再使用他的表 1 中的数据进行求解，进而获得对数的区间估计值。在计算纳皮尔的第 3 张表和小表(乘

数表)时, Denis Roegel 给出了纳皮尔方法与未优化方法的对比, 本文借助于 MPRF 库给出了 200 位精度上的比较结果(为方便显示, 保留为 20 位精度), 并增加了在计算纳皮尔第 2 张表时与外插方式的精度对比, 外插的方法见 C. H. Edwards 的方法^[25]。

本文在分析两类纳皮尔对数证明方法的基础上, 提出了新的基于指数函数构造的证明方法。同时, 本文还分析了纳皮尔在计算他的第 2 张表和三角对数表中使用的优化方法, 借助于 MPRF 库, 在 20 位精度上给出了纳皮尔方法的精度优化结果, 结果表明纳皮尔的方法均优于其他对照方法。

2 纳皮尔对数的数学证明

2.1 纳皮尔的 3 张表

纳皮尔一开始也从等差数列和几何数列之间的对应关系来研究, 他构建的第一张表如表 1 所示, 为方便读者阅读, 本文使用现代符号基于纳皮尔的方法重新编制该表(下文中的表和图类似)。在纳皮尔的第 1 张表中, 他选择 10^7 作为第一项, 然后用一个非常接近于 1, 但又小于 1 的公比($1-10^{-7}$, 即 0.999 999 9)来构建等比数列, 第 1 张表共有 101 个数(为方便展示, 本文将整正弦 10 000 000 也放入其中), 本文采用 Denis Roegel 使用的符号^[25], 将表 1 中的数用

表 1 纳皮尔的第 1 张表

Table 1 Napier's first table

编号	数(等比数列)	
	用等比表示	用小数表示
a_0	10^7	10 000 000.000 000 0
a_1	$a_0 \cdot 0.999\ 999\ 9^1$	9 999 999.000 000 0
a_2	$a_0 \cdot 0.999\ 999\ 9^2$	9 999 998.000 000 1
...
a_{99}	$a_0 \cdot 0.999\ 999\ 9^{99}$	9 999 901.000 485 1
a_{100}	$a_0 \cdot 0.999\ 999\ 9^{100}$	9 999 900.000 495 0

a_0, a_1, \dots, a_{100} 来表示。

纳皮尔的第 2 张表如表 2 所示, 他把公比设为 $0.999\ 99(1-10^{-5})$, 表中共有 51 个数(为方便展示, 本文将整正弦 10 000 000 也放入其中), 这里用 b_0, b_1, \dots, b_{50} 表示, 表中的第一个数 b_0 接近于纳皮尔的第一个表中的最后一个数 a_{100} 。需要提及的是, 纳皮尔原表的最后一个数 b_{50} 的计算有误(他的计算结果为 9 995 001.222 927), 本文表 2 中的数据根据纳皮尔的方法计算求得, 和纳皮尔原表中的值不同。

表 2 纳皮尔的第 2 张表

Table 2 Napier's second table

编号	数(等比数列)	
	用等比表示	用小数表示
b_0	10^7	10 000 000.000 000 0
b_1	$b_0 \cdot 0.999\ 99^1$	9 999 900.000 000 0
b_2	$b_0 \cdot 0.999\ 99^2$	9 999 800.001 000 0
...
b_{49}	$b_0 \cdot 0.999\ 99^{49}$	9 995 101.175 816
b_{50}	$b_0 \cdot 0.999\ 99^{50}$	9 995 001.224 804

纳皮尔的第 3 张表略显复杂, 如表 3 所示。该表共有 69 列数据(表中编号从 0 到 68), 每列有 21 行(编号从 0 到 20), 这里每个数用 $c_{i,j}$ 表示, 其中, $0 \leq i \leq 20, 0 \leq j \leq 68$ 。

2.2 纳皮尔对数的定义及法则

纳皮尔在吸纳前人等比数列和等差数列对应的思想的基础上, 用几何和连续运动来定义对数, 如图 1 所示。对于等差数列来说, 纳皮尔用一条射线 α 和一个匀速运动的点 A 来表示, 点 A 的速度恒为 10^7 。对于对应的等比数列来说, 纳皮尔用一条线段 $\beta(B_0, B_e)$ 分别为起点和终点, $B_0 B_e$ 的长度为 10^7) 和另一个变速运动的点 B 来表示, 在初始位置 B_0 处, 速度为 10^7 , 然后要求 B 的速度按几何级数的形式下降, 即在同样的一段时间内, 速度降低同样的一个比率(假设降低的比率为实数 $r, 0 < r < 1$)。然后纳皮尔给出了对数的定义: 在同一时刻出发后, 在 β 线段上,

表 3 纳皮尔的第 3 张表

Table 3 Napier's third table

数(等比数列)						
第 0 列			第 1 列			...
编号	用等比表示	用小数表示	编号	用等比表示	用小数表示	...
$c_{0,0}$	$10^7 \cdot 0.99^0$	10 000 000.000 0	$c_{0,1}$	$10^7 \cdot 0.99^1$	9 900 000.000 0	...
$c_{1,0}$	$c_{0,0} \cdot 0.999 5^1$	9 995 000.000 0	$c_{1,1}$	$c_{0,1} \cdot 0.999 5^1$	9 895 050.000 0	...
$c_{2,0}$	$c_{0,0} \cdot 0.999 5^2$	9 990 002.500 0	$c_{2,1}$	$c_{0,1} \cdot 0.999 5^2$	9 890 102.475 0	...
...
$c_{19,0}$	$c_{0,0} \cdot 0.999 5^{19}$	9 905 426.291 2	$c_{19,1}$	$c_{0,1} \cdot 0.999 5^{19}$	9 806 372.028 3	...
$c_{20,0}$	$c_{0,0} \cdot 0.999 5^{20}$	9 900 473.578 0	$c_{20,1}$	$c_{0,1} \cdot 0.999 5^{20}$	9 801 468.842 2	...

数(等比数列)						
...	第 67 列			第 68 列		
...	编号	用等比表示	用小数表示	编号	用等比表示	用小数表示
...	$c_{0,67}$	$10^7 \cdot 0.99^{67}$	5 099 857.462 5	$c_{0,68}$	$10^7 \cdot 0.99^{68}$	5 048 858.887 9
...	$c_{1,67}$	$c_{0,67} \cdot 0.999 5^1$	5 097 307.533 8	$c_{1,68}$	$c_{0,68} \cdot 0.999 5^1$	5 046 334.458 4
...	$c_{2,67}$	$c_{0,67} \cdot 0.999 5^2$	5 094 758.880 0	$c_{2,68}$	$c_{0,68} \cdot 0.999 5^2$	5 043 811.291 2
...
...	$c_{19,67}$	$c_{0,67} \cdot 0.999 5^{19}$	5 051 626.219 0	$c_{19,68}$	$c_{0,68} \cdot 0.999 5^{19}$	5 001 109.956 8
...	$c_{20,67}$	$c_{0,67} \cdot 0.999 5^{20}$	5 049 100.405 9	$c_{20,68}$	$c_{0,68} \cdot 0.999 5^{20}$	4 998 609.401 9

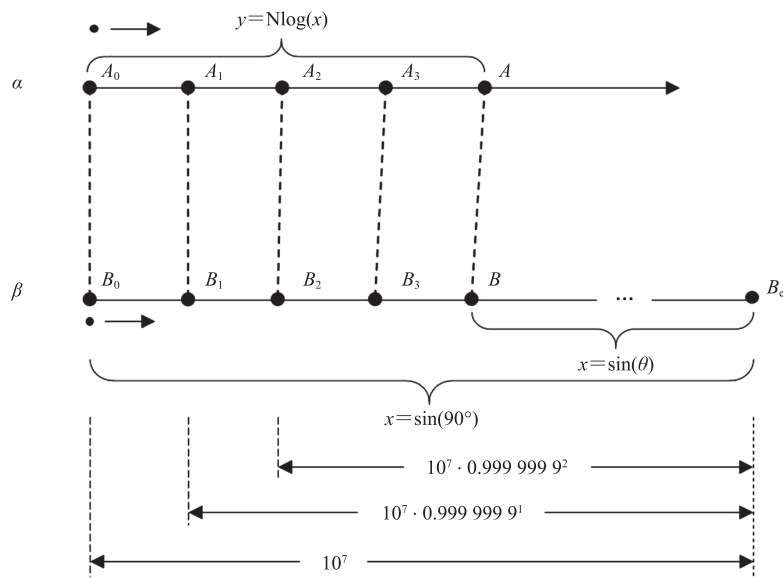


图 1 纳皮尔利用几何和连续运动来定义对数

Fig. 1 Napier utilized geometry and continuous motion to define logarithm

B 点距终点的距离 BB_c 。其对数值定义为在 α 线上, A 点经过的距离 A_0A 。例如, 经过相同的时间, 点 B 从 B_0 到达 B_1 处, 点 A 从 A_0 到达 A_1

处, 则 B_1B_c 的对数为 A_0A_1 。

为表述方便, 本文用 $N\log(x)$ 来表示正实数 x 对应的纳皮尔对数。

2.2.1 *Constructio* 中的第 28、30 条论断

纳皮尔为求解他的第 1 张表中的第 1 个数 a_1 的对数(毫无疑问, a_0 的纳皮尔对数为 0, 无须计算), 首先给出了第 1 个论断, 即 *Constructio* 中的第 28、30 条^[26], 其原理如图 2 所示。

本文把该论断描述为“初始数的对数的区间估计”, 即“在线段 β 上, 从起点出发一个微小距离 B_0B_1 , 其余下距离对应的对数的大小位于 $(B_0B_1, B_0B_1 \cdot \frac{1}{r})$ 之间”。其中, r 同上文定义, 为速度降低的比率。读者很容易根据速度的大小关系推导出该论断, 此处不再给出证明, 用现代符号表示就是:

$$B_0B_1 < A_0A_1 < B_0B_1 \cdot \frac{1}{r} \quad (1)$$

代入 a_1 (值为 $10^7 \cdot 0.999\ 999\ 9^1$) 和 r (值为 $0.999\ 999\ 9$), 可求得:

$$(10^7 - 10^7 \cdot 0.999\ 999\ 9^1) < N\log(a_1) < (10^7 - 10^7 \cdot 0.999\ 999\ 9^1) \cdot \frac{1}{0.999\ 999\ 9}$$

即: $1 < N\log(a_1) < 1.000\ 000\ 05$ 。

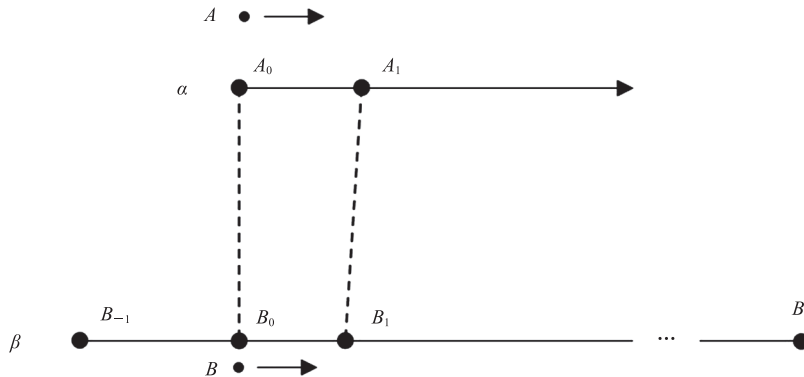


图 2 纳皮尔对他的第 1 张表中的第 1 个数的对数进行区间估计

Fig. 2 Napier gives the interval approximation for the first number in his first table



图 3 纳皮尔由已知数的对数求解相邻数的对数的方法

Fig. 3 Napier gives the solving method for a number based on adjacent known logarithm value

2.2.2 *Constructio* 中的第 36 条论断

本文把 *Constructio* 中的第 36 条论断描述为“若两组数的比相同, 则它们的纳皮尔对数的差相等”^[26]。用现代符号表示为:

$$\text{若 } \frac{\alpha}{\beta} = \frac{\gamma}{\delta}, \quad (2)$$

$$\text{则 } N\log(\alpha) - N\log(\beta) = N\log(\gamma) - N\log(\delta)$$

其中, α 、 β 、 γ 、 δ 为正实数。

2.2.3 *Constructio* 中的第 39 条论断

根据已知数的对数, 求得相邻数的对数, 纳皮尔针对这一问题在 *Constructio* 中的第 39 条论断给出了相应的计算方法, 如图 3 所示。

在图 3 中, B_0 为起点, 假设 B_2 处的对数已知, 现求解与 B_2 相邻的数 B_3 处的对数。纳皮尔通过在初始点 B_0 两侧构造两个点 B_{-1} 、 B_1 来辅助求解, 使得满足:

$$\frac{B_{-1}B_0}{B_0B_c} = \frac{B_2B_3}{B_3B_c}, \quad \frac{B_0B_1}{B_0B_c} = \frac{B_2B_3}{B_2B_c}$$

读者很容易结合公式(2)得出纳皮尔的结论, 此处不再给出证明。纳皮尔的公式用现代符号表示为:

$$10^7 \cdot \frac{B_2 B_c - B_3 B_c}{B_2 B_c} < N\log(B_3 B_c) - N\log(B_2 B_c) < 10^7 \cdot \frac{B_2 B_c - B_3 B_c}{B_3 B_c} \quad (3)$$

2.3 前人的证明

此处列出前人有代表性的两个证明，分别为 C. H. Edwards 的方法和 Denis Roegel 的方法^[25-26]。

2.3.1 C. H. Edwards 的证明

C. H. Edwards 从纳皮尔的一个推论开始求解，即纳皮尔推出 B 点的速度大小等于余下的距离，如图 4 所示。

图中 x 轴在 β 上，起点为 B_c ，方向向左，而速度的方向向右，与 x 轴方向相反，即有：

$$\frac{dx}{dt} = -x$$

化简后，有 $t = -\ln x + c$ 。其中， c 为常数； x 为 β 上运动到 B 点余下的距离 BB_c 。

然后根据初始条件，即在 $t=0$ 时刻， $x=10^7$ ，代入：

$$0 = -\ln 10^7 + c$$

求得 c ，故：

$$t = -\ln x + \ln 10^7$$

而根据定义，纳皮尔对数 y 为：

$$y = 10^7 t = 10^7 (-\ln x + \ln 10^7)$$

即：

$$N\log(x) = -10^7 \ln x + 10^7 \ln 10^7 \quad (4)$$

证毕。

2.3.2 Denis Roegel 的证明

Denis Roegel 证明用到的符号如图 5 所示。

设 x 轴在 β 上，原点为 B_0 ， B 为运动的点， x 为 B 经过的距离， BB_c 为剩余的距离， T 为时间段。根据纳皮尔的“速度成几何比率下降”的定义，以及“余下的距离和速度相等”的论断：

$$\frac{10^7 - x((n+1)T)}{10^7 - x(nT)} = r \quad (5)$$

其中， r 同上定义，为速度降低的比率。公式(5)进一步化简为：

$$10^7 - x(nT) = r^n (10^7 - x(0)) = r^n 10^7$$

$$\text{即：} x(nT) = 10^7 - r^n 10^7 = 10^7 (1 - r^n)$$

$$\text{设 } t = nT, a = r^{1/T}, \text{ 则：} x(t) = 10^7 (1 - a^t)$$

$$\text{对上式求导：} x'(t) = -10^7 a^t \ln a$$

$$\text{带入 } t=0, \text{ 得：} v(0) = x'(0) = -10^7 a^0 \ln a = -10^7 \ln a$$

而根据定义，初始速度为 10^7 ，故 $a = e^{-1}$ 。

$$\text{故：} x(t) = 10^7 (1 - e^{-t})$$

余下的距离为 $10^7 - x(t) = 10^7 e^{-t}$ ，其对数定义为 $10^7 t$ ，即： $N\log(10^7 e^{-t}) = 10^7 t$ ，

设余下的距离 $10^7 e^{-t}$ 用 y 来表示，则 $t = \ln 10^7 - \ln y$ ，

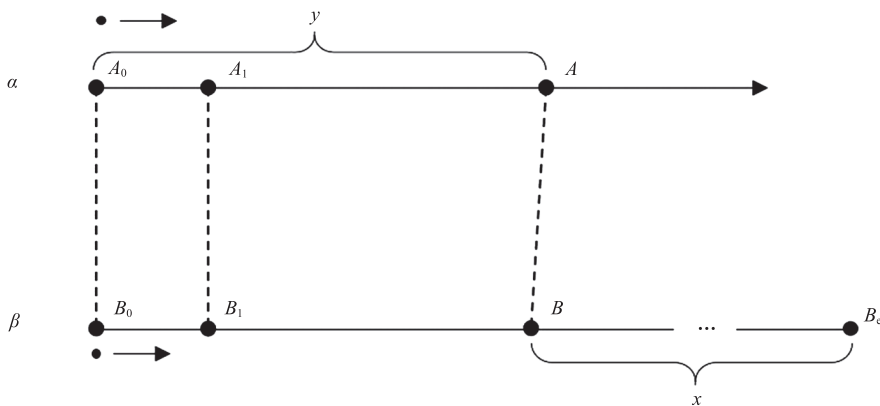


图 4 C. H. Edwards 的证明方法

Fig. 4 C. H. Edwards's proof method

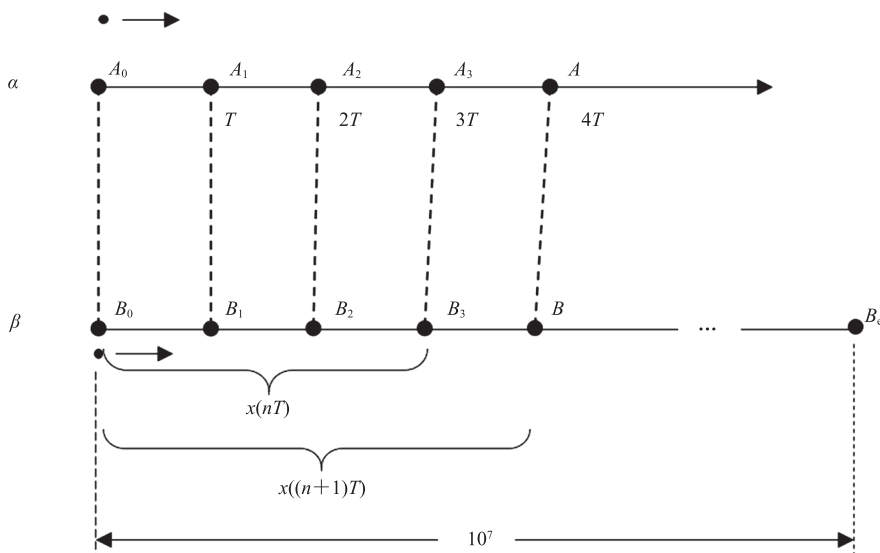


图 5 Denis Roegel 的证明方法

Fig. 5 Denis Roegel's proof method

故: $N\log(y) = 10^7 (\ln 10^7 - \ln y)$ 。

即: $N\log(y) = -10^7 \ln y + 10^7 \ln 10^7$ 。

证毕。

2.4 基于指数函数构造的方法

本文直接从指数函数的构造开始, 证明用到的符号如图 6 所示。

尽管纳皮尔没有为速度和长度设置相应的单

位, 但为了易于理解, 这里假设速度的单位都为 m/s, 长度的单位都为 m。按照纳皮尔的定义, B 点在 B_0 的初速度 $v_0 = 10^7$ m/s, 整个线段 B_0B_c 的长度为 10^7 m, 速度降低的比率 r 取为表 1 中的 0.999 999, 为便于表述, 本文把时间 t 放在 10^{-7} s 这样的单位下观测。用现代的观点来看, B 的速度 v 为时间 t 的指数函数:

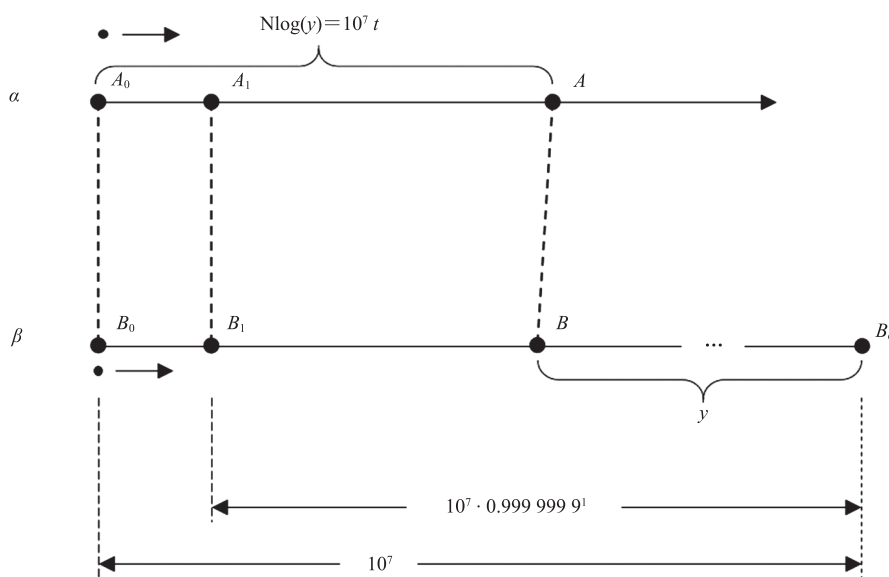


图 6 基于指数函数的证明方法

Fig. 6 The proof method based on exponential function

$$v(t)=10^7 \cdot 0.999\ 999\ 9^{10^7 t}$$

求出时间 t 关于 v 的函数:

$$t(v)=10^{-7} \cdot \log_{0.999\ 999\ 9} \left(\frac{v}{10^7} \right)$$

进而可以得到时间 t 内经过的距离 $B_0 B$ 为:

$$\begin{aligned} BB_c &= \int_0^t v(t) dt \\ &= \int_0^t 10^7 \cdot 0.999\ 999\ 9^{10^7 t} \cdot dt \\ &= \frac{1}{\ln 0.999\ 999\ 9} \cdot \left(\frac{v}{10^7} - 1 \right) \end{aligned}$$

余下的距离 BB_c 为:

$$BB_c = B_0 B_c - B_0 B = 10^7 - \frac{1}{\ln 0.999\ 999\ 9} \cdot \left(\frac{v}{10^7} - 1 \right)$$

化简后得: $v = -10^7 \cdot \ln 0.999\ 999\ 9 \cdot BB_c + 10^{14} \cdot \ln 0.999\ 999\ 9 + 10^7$.

根据纳皮尔对数的定义:

$$\begin{aligned} N\log(BB_c) &= A_0 A \\ &= 10^7 t \\ &= 10^7 \cdot 10^{-7} \cdot \log_{0.999\ 999\ 9} \left(\frac{v}{10^7} \right) \\ &= \log_{0.999\ 999\ 9} \left(\frac{v}{10^7} \right) \end{aligned}$$

把 v 代入:

$$N\log(BB_c) = \log_{0.999\ 999\ 9} \left(-\ln 0.999\ 999\ 9 \cdot BB_c + 10^7 \cdot \ln 0.999\ 999\ 9 + 1 \right),$$

用 y 表示 BB_c , 故上式为:

$$N\log(y) = \log_{0.999\ 999\ 9} \left(-\ln 0.999\ 999\ 9 \cdot y + 10^7 \cdot \ln 0.999\ 999\ 9 + 1 \right).$$

证毕。

2.5 进一步改进的方法

2.4 小节中的证明用了一些具体的数值(如 r 取 0.999 999 9), 可以进一步优化, 根据 $v(t) = 10^7 \cdot 0.999\ 999\ 9^{10^7 t}$, 设 $a = 0.999\ 999\ 9^{10^7}$, 而 n 设为

10^7 , 则 $0.999\ 999\ 9 = 1 - \frac{1}{n}$, 即: $a = \left(1 - \frac{1}{n} \right)^n$ 。取

极限, 则: $a = \lim_{n \rightarrow \infty} \left(1 - \frac{1}{n} \right)^n = e^{-1}$ 。

故: $v(t) = 10^7 \cdot e^{-t}$ 。

设余下的距离为 $y(t)$, 则经过的距离为:

$$10^7 - y(t) = \int_0^t v(t) dt.$$

$$\begin{aligned} \text{故: } y(t) &= 10^7 - \int_0^t v(t) dt \\ &= 10^7 - \int_0^t 10^7 \cdot e^{-t} \cdot dt \\ &= 10^7 + 10^7 (e^{-t} + c). \end{aligned}$$

其中, c 为常数。代入初始值 $y(0) = 10^7$, 得: $c = -1$ 。

故: $y = 10^7 + 10^7 (e^{-t} - 1) = 10^7 \cdot e^{-t}$ 。

即: $t = -\ln y + \ln 10^7$ 。

$$\begin{aligned} \text{故: } N\log(y) &= 10^7 t \\ &= 10^7 (-\ln y + \ln 10^7) \\ &= -10^7 \ln y + 10^7 \ln 10^7. \end{aligned}$$

证毕。

3 纳皮尔对数的精度分析

纳皮尔为了取得更好的对数的区间估计值的精度, 他在几个地方提出了独特的方法, 分别如下。

在计算 b_1 时, 没有用通过 a_{99} 、 a_{100} 进行外插的方法(下文称为替代方法), 而是用公式(3)的方法。

在计算 $c_{1,0}$ 时, 没有仅仅用公式(3), 即通过相邻数 b_{50} 计算的方法(下文称为替代方法), 而是综合运用公式(2)~(3)进行了计算。

接下来, 本文将公式(4)计算的结果作为真值, 以分析纳皮尔求得的对数的精度的情况。

3.1 计算 b_1

3.1.1 替代方法

首先, 在计算 b_1 时, 一个替代方法是可以通
过 a_{99} 和 a_{100} , 向外线性插值得到 b_1 的对数。例
如: 通过 $N\log(a_{99})$ 的下边界(99)和 $N\log(a_{100})$ 的
上边界(100.000 010 000 001 000 000 10, 为精确
起见采用 20 位小数, 下同)进行外插, 可以得到
 $N\log(b_1)$ 的上边界为 100.000 505 008 234 545 745
69, 即:

$$\frac{N\log(b_1)=100.000\ 010\ 000\ 001\ 000\ 000\ 10+}{9\ 999\ 900.000\ 494\ 998\ 383\ 003\ 921\ 22-9\ 999\ 900} \times \frac{9\ 999\ 901.000\ 485\ 098\ 431\ 513\ 764\ 37-9\ 999\ 900.000\ 494\ 998\ 383\ 003\ 921\ 22}{(100.000\ 010\ 000\ 001\ 000\ 000\ 10-99)}$$

同理, 通过 $N\log(a_{99})$ 的上边界(99.000
009 900 000 990 000 10)和 $N\log(a_{100})$ 的下边界
(100)进行外插, 可以得到 $N\log(b_1)$ 的下边界为
100.000 494 998 382 979 418 63。

有了 $N\log(b_1)$ 的上、下边界后, 可以求得二
者的算术平均值为 100.000 500 003 308 762 582 16。

3.1.2 纳皮尔的方法

而纳皮尔的方法运用公式(3), 即把 $B_2B_e =$
 a_{100} , $B_3B_e = b_1$ 分别代入, 得:

$$\begin{aligned} &0.000\ 495\ 003\ 333\ 012\ 748\ 76 < \\ &N\log(B_3B_e) - N\log(B_2B_e) < \\ &0.000\ 495\ 003\ 333\ 037\ 251\ 59 \end{aligned}$$

故用 $N\log(a_{100})$ 下边界 100 加上上式中较
小的值 0.000 495 003 333 012 748 76, 用上边界
100.000 010 000 001 000 000 10 加上上式中较大
的值 0.000 495 003 333 037 251 59, 得:

$$\begin{aligned} &100.000\ 495\ 003\ 333\ 012\ 748\ 76 < N\log(B_3B_e) < \\ &100.000\ 505\ 003\ 334\ 037\ 251\ 69 \end{aligned},$$

也即:

$$\begin{aligned} &100.000\ 495\ 003\ 333\ 012\ 748\ 76 < N\log(b_1) < \\ &100.000\ 505\ 003\ 334\ 037\ 251\ 69 \end{aligned}。$$

3.1.3 精度比较

通过公式(4)求得真值 $N\log(b_1)$, 即 $N\log(b_1) =$
100.000 500 003 333 358 333 53。

纳皮尔的方法和上述替代方法(通过外插方
法)的比较如表 4 所示。从表中可以看到, 与替
代方法相比, 纳皮尔方法计算的上、下边界及算
术平均值更为接近真值。

3.2 计算 $c_{1,0}$

3.2.1 替代方法

计算 $c_{1,0}$ 时, 一个替代方法是直接通过相邻的
 b_{50} , 利用公式(3)进行计算, 下面是计算过程。

将 $b_{50} = 9\ 995\ 001.224\ 804\ 023\ 027\ 881\ 398\ 90$
代入公式(3), 则:

$$\begin{aligned} &10^7 \cdot \frac{9\ 995\ 001.224\ 804\ 023\ 027\ 881\ 398\ 90 - 9\ 995\ 000.000\ 0}{9\ 995\ 001.224\ 804\ 023\ 027\ 881\ 398\ 90} < \\ &N\log(c_{1,0}) - N\log(b_{50}) < \\ &10^7 \cdot \frac{9\ 995\ 001.224\ 804\ 023\ 027\ 881\ 398\ 90 - 9\ 995\ 000.000\ 0}{9\ 995\ 000.000\ 0} \end{aligned}$$

得:

$$\begin{aligned} &1.225\ 416\ 581\ 228\ 980\ 031\ 48 < \\ &N\log(c_{1,0}) - N\log(b_{50}) < \\ &1.225\ 416\ 731\ 393\ 578\ 187\ 99 \end{aligned}。$$

故用 $N\log(b_{50})$ 下边界 5 000.024 750 166 650

表 4 计算 b_1 时, 纳皮尔的方法与替代方法(外插方法)的比较

Table 4 Napier's method being compared with alternative method (extrapolation) when calculating b_1

	求解量	求解值	求解值与真值 $N\log(b_1)$ 的差
纳皮尔的方法(通过公式(3)求得), 表示为 $N\log(b_1)$ '	下边界	100.000 495 003 333 012 748 76	-0.000 005 000 000 345 584 77
	上边界	100.000 505 003 334 037 251 69	0.000 005 000 000 678 918 16
	算术平均值	100.000 500 003 333 525 000 23	0.000 000 000 000 166 666 7 = 1.666 667 × 10 ⁻¹³
替代方法(通过 a_{99} 、 a_{100} 进行外插 的方法求得), 表示为 $N\log(b_1)$ "	下边界	100.000 494 998 382 979 418 63	-0.000 005 004 950 378 914 9
	上边界	100.000 505 008 234 545 745 69	0.000 005 004 901 187 412 16
	算术平均值	100.000 500 003 308 762 582 16	-0.000 000 000 024 595 751 3 = -2.459 575 13 × 10 ⁻¹¹

637 438 00 加上上式中较小的值 1.225 416 581 228 980 031 48, 用 $N\log(b_{50})$ 上边界 5 000.025 250 166 701 862 584 50 加上上式中较大的值 1.225 416 731 393 578 187 99, 得:

$$5\ 001.250\ 166\ 747\ 879\ 617\ 469\ 48 < N\log(c_{1,0}) < \\ 5\ 001.250\ 666\ 898\ 095\ 440\ 772\ 49 \quad \circ$$

3.2.2 纳皮尔的方法

纳皮尔的方法综合运用公式(2)~(3)进行计算。

纳皮尔首先利用公式(2), 代入 $\frac{c_{1,0}}{b_{50}} = \frac{\gamma}{10^7}$, 求得:

$$\gamma = 10^7 \cdot \frac{c_{1,0}}{b_{50}} \\ = 10^7 \cdot \frac{9\ 995\ 000.000\ 0}{9\ 995\ 001.224\ 804\ 023\ 027\ 881\ 398\ 90} \\ = 9\ 999\ 998.774\ 583\ 418\ 771\ 019\ 968\ 52$$

那么,

$$N\log(c_{1,0}) - N\log(b_{50}) = N\log(\gamma) - N\log(10^7) \\ = N\log(\gamma) \quad \circ$$

然后通过查找纳皮尔的第一个表, 找到最接近 γ 的数为 a_1 , 即 9 999 999, 然后再运用公式(3), 得:

$$10^7 \cdot \frac{9\ 999\ 999 - 9\ 999\ 998.774\ 583\ 418\ 771\ 019\ 968\ 52}{9\ 999\ 999} < \\ N\log(\gamma) - N\log(a_1) < \\ 10^7 \cdot \frac{9\ 999\ 999 - 9\ 999\ 998.774\ 583\ 418\ 771\ 019\ 968\ 52}{9\ 999\ 998.774\ 583\ 418\ 771\ 019\ 968\ 52}$$

而 $1 < N\log(a_1) < 1.000\ 000\ 100\ 000\ 01$,

故: 1.225 416 603 770 640 408 54 <

$$N\log(\gamma) <$$

1.225 416 708 851 915 048 63 ,

也即: 1.225 416 603 770 640 408 54 <

$$N\log(c_{1,0}) - N\log(b_{50}) <$$

1.225 416 708 851 915 048 63 .

这样再根据 $N\log(b_{50})$:

$$5\ 000.024\ 750\ 166\ 650\ 637\ 438\ 00 <$$

$$N\log(b_{50}) <$$

$$5\ 000.025\ 250\ 166\ 701\ 862\ 584\ 50 ,$$

得: 5 001.250 166 770 421 277 846 54 <

$$N\log(c_{1,0}) <$$

5 001.250 666 875 553 777 633 13 .

3.2.3 精度比较

通过公式(4)求得真值 $N\log(c_{1,0})$, 即 $N\log(c_{1,0}) = 5\ 001.250\ 416\ 822\ 979\ 192\ 719\ 50$ 。

比较纳皮尔的方法和替代方法(仅通过相邻数 b_{50} 用公式(3)计算), 如表 5 所示。从表中可以看到, 与替代方法相比, 纳皮尔的方法计算的上、下边界及算术平均值更为收敛到真值。

4 计算说明

文中的计算采用 MPFR 库^[28], 需要说明的是, 初始化 mpfr_t 变量时使用了 mpfr_set_str(), 而不是 mpfr_set_d(.) 函数。以设置比例数 0.999 999 9 为例, 当使用 mpfr_set_d(r, 0.999 999 9, MPFR_RNDD) 时, 实际存储为:

$$9.999\ 999\ 000\ 000\ 000\ 526\ 355\ 847\ 796\ 367\ 015 \\ 689\ 611\ 434\ 936\ 523\ 437\ 500\ 000\ 000\ 0 \times 10^{-1}.$$

而使用 mpfr_set_str(r, "0.999 999 9", 10, GMP_RNDN) 时, 实际存储为:

$$9.999\ 998\ 999\ 999\ 999\ 999\ 999\ 999\ 999\ 999 \\ 999\ 999\ 999\ 999\ 999\ 999\ 999\ 997\ 0 \times 10^{-1}.$$

显然, 后者具有稍高的精度。

5 结 论

本文阐述了纳皮尔对数的定义及其 3 张表, 在分析 C. H. Edwards 和 Denis Roegel 的两类经典的纳皮尔对数证明方法的基础上, 提出了新

表 5 计算 $c_{1,0}$ 时, 纳皮尔的方法与替代方法 (仅通过相邻数 b_{50} 使用公式 (3) 计算) 的比较Table 5 Napier's method being compared with alternative method (only being calculated by adjacent number b_{50} using formula (3)) when calculating $c_{1,0}$

	求解量	求解值	求解值与真值 $N\log(c_{1,0})$ 的差
纳皮尔的方法(综合运用公式(2)和公式(3)进行计算), 表示为 $N\log(c_{1,0})'$	下边界	5 001.250 166 770 421 277 846 54	-0.000 250 052 557 914 872 96
	上边界	5 001.250 666 875 553 777 633 13	0.000 250 052 574 584 913 63
	算术平均值	5 001.250 416 822 987 527 739 84	0.000 000 000 008 335 020 34=8.335 020 34 $\times 10^{-12}$
替代方法(仅通过相邻数 b_{50} 用公式(3)计算), 表示为 $N\log(c_{1,0})''$	下边界	5 001.250 166 747 879 617 469 48	-0.000 250 075 099 575 250 02
	上边界	5 001.250 666 898 095 440 772 49	0.000 250 075 116 248 052 99
	算术平均值	5 001.250 416 822 987 529 120 99	0.000 000 000 008 336 401 49=8.336 401 49 $\times 10^{-12}$

的基于指数函数构造的证明方法, 同时还给出了基于自然常数 e 的更为优化的方法。此外, 本文还分析了纳皮尔在计算 b_1 和 $c_{1,0}$ 时的方法, 以及其他对照可选的计算方法, 借助于 MPRF 库, 在 20 位精度上给出了纳皮尔的方法的精度优化结果, 结果表明纳皮尔的方法优于对照方法。

参 考 文 献

- [1] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 779-788.
- [2] Redmon J, Farhadi A. YOLO9000: better, faster, stronger [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 7263-7271.
- [3] Redmon J, Farhadi A. YOLOv3: an incremental improvement [Z/OL]. arXiv Preprint, arXiv: 1804.02767, 2018.
- [4] Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: optimal speed and accuracy of object detection [Z/OL]. arXiv Preprint, arXiv: 2004.10934, 2020.
- [5] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector [C] // Proceedings of the European Conference on Computer Vision, 2016: 21-37.
- [6] Ren SQ, He KM, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.
- [7] Law H, Deng J. CornerNet: detecting objects as paired keypoints [C] // Proceedings of the European Conference on Computer Vision, 2018: 734-750.
- [8] Zhang SF, Wen LY, Bian X, et al. Single-shot refinement neural network for object detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 4203-4212.
- [9] Qi CR, Su H, Mo KC, et al. PointNet: deep learning on point sets for 3D classification and segmentation [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 652-660.
- [10] Zhou Y, Tuzel O. VoxelNet: end-to-end learning for point cloud based 3D object detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 4490-4499.
- [11] Badrinarayanan V, Kendall A, Cipolla R. SegNet: a deep convolutional encoder-decoder architecture for image segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.
- [12] He KM, Gkioxari G, Dollár P, et al. Mask R-CNN [C] // Proceedings of the IEEE International Conference on Computer Vision, 2017: 2961-2969.
- [13] Tian YJ, Su D, Lauria S, et al. Recent advances on loss functions in deep learning for computer vision [J]. Neurocomputing, 2022, 497: 129-158.

- [14] Sun P, Wang YP, He P, et al. GCD-L: a novel method for geometric change detection in HD maps using low-cost sensors [J]. *Automotive Innovation*, 2022, 5: 324-332.
- [15] Sun P, Dong FF, Wang YP, et al. SPIL: a shape preserving method for improving lane width accuracy of HD maps [C] // *Proceedings of the 2021 6th International Conference on Control, Robotics and Cybernetics*, 2021: 250-256.
- [16] 姜娜娜, 汤咏林, 黄鹤, 等. 自动驾驶高精地图相对精度验证方法研究 [J]. *南京信息工程大学学报: 自然科学版*, 2023, 15(2): 193-200.
Jiang NN, Tang YL, Huang H, et al. Relative accuracy verification of high-definition map for autonomous driving [J]. *Journal of Nanjing University of Information Science & Technology: Natural Science Edition*, 2023, 15(2): 193-200.
- [17] 孙达, 蒲英霞. 地图投影 [M]. 2 版. 南京: 南京大学出版社, 2012.
Sun D, Pu YX. *Map projections* [M]. 2nd ed. Nanjing: Nanjing University Press, 2012.
- [18] 陈少丽. 对数的发明及其相关历史分析 [D]. 太原: 山西师范大学, 2012.
Chen SL. *Invention and relevant history analysis of the logarithm* [D]. Taiyuan: Shanxi Normal University, 2012.
- [19] Rice B, González-Velasco E, Corrigan A. *The life and works of John Napier* [M]. Berlin: Springer, 2017.
- [20] Raugh M. John Napier and Henry Briggs at the threshold of analysis [EB/OL]. (2022-5-15)[2023-10-11]. <https://www.mikeraugh.org/NapierBriggs/NapierBriggs-20220515-ZDwnld.pdf>.
- [21] Havil J. *John Napier: life, logarithms, and legacy* [M]. Princeton: Princeton University Press, 2014.
- [22] Villarreal-Calderon R. Chopping logs: a look at the history and uses of logarithms [J]. *The Mathematics Enthusiast*, 2008, 5(2-3): 337-344.
- [23] Napier J. *Mirifici logarithmorum canonis descriptio* [M]. Edinburgh: Andrew Hart, 1614.
- [24] Napier J. *The construction of the wonderful canon of logarithms* [M]. Edinburgh: Blackwood & Sons, 1620.
- [25] Roegel D. Napier's ideal construction of the logarithms [EB/OL]. (2020-11-6)[2023-10-11]. <https://inria.hal.science/inria-00543934/document>.
- [26] Edwards CH. *The historical development of the calculus* [M]. New York: Springer Science & Business Media, 1979.
- [27] Cajori F. History of the exponential and logarithmic concepts [J]. *The American Mathematical Monthly*, 1913, 20(2): 35-47.
- [28] The GNU MPFR library 4.2.1. [EB/OL]. (2023-8-22)[2023-10-11]. <https://www.mpfr.org/>.