基于FPGA的零误差大数阶乘算法的设计与实现

刘公绪,史凌峰,辛东金
(西安电子科技大学电子工程学院,陕西西安 710071)

摘 要: 随着大数据时代的到来,人们对超高精度科学计算的需求日益迫切,其中一个难点是大数阶乘问题.斯特林公式作为计算大数阶乘的传统近似方法,远不能在精度上满足要求,其它的阶乘算法可以实现较高的精度,但以牺牲大量存储空间为代价.本文提出一种具有零误差的大数阶乘算法,可以根据问题规模优化存储空间,利用并行计算的思想和FPGA的优势来提高计算速度,测试结果表明,所提出的算法具有较好的时空效率,可以应用在如大数阶乘计算器等诸多领域.

关键词: 阶乘算法;存储空间;并行计算;FPGA
中图分类号: TP311 **文献标识码:** A **文章编号:** 0372-2112 (2019)05-1180-05
电子学报URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2019.05.027

FPGA-Based Zero-Error Factorial Algorithm for Large Integer

LIU Gong-xu, SHI Ling-feng, XIN Dong-jin
(*School of Electronic Engineering, Xidian University, Xi'an, Shaanxi* 710071, *China*)

Abstract: With the advent of the era of big data, there is an increasingly urgent need for ultra-high precision scientific calculation. One of the difficulties is the large number factorial problem. Stirling formula as the traditional approximate method of calculating factorial of large numbers cannot meet the accuracy requirements. Other factorial algorithm can achieve high accuracy, but at the cost of a large amount of storage space. This paper puts forward a factorial algorithm of large numbers with zero error that can optimize storage space according to the problem size. Besides, the computing speed can be improved by making full use of parallel computing and the advantages of field programmable gate array (FPGA). The test results show that the proposed algorithm has better efficiency of time and space, which can be used in many fields such as large factorial calculator.

Key words: factorial algorithm; storage space; parallel computation; FPGA (field programmable gate array)

1 引言

对阶乘多项式的最初定义和基本特性的分析可以追溯到上世纪九十年代[1].文献[2]对阶乘的概念和性质做了一定补充.文献[3]提出了一种用于编号排列的阶乘数字系统,大数阶乘的应用由此可见一斑.众所周知,许多现实中的工业和经济领域的问题是排列组合和优化问题,这都离不开阶乘运算.此外,计算机对泰勒多项式和自然对数 e 的高阶近似,本质也是求整数 n 的阶乘.

当问题规模比较大,即计算大数 n 的阶乘时,计算过程不可避免的消耗大量时间,这显然不能满足实时超高精度解的要求.人们对超高精度解算的需求越来越迫切[4].事实上,阶乘算法常以迭代和递推的形式呈现,虽然其可以用各种语言来实现,但算法效率不高[5,6].

此外,整数在计算机中的表示形式有天花板,其最大表示值取决于编译器,如在64位编译器中,整数最大可以表示为 $2^{64}-1$,换句话说,其只能精确表示20的阶乘.因此,需要重新定义新的整数类型,或者研究新的算法,可以让计算机处理和表示更大整数的阶乘.

利用链表或动态数组可以表示大于20的整数的阶乘,因为对链表的访问普遍要比对数组的访问慢,并且构建链表往往需要更多的存储空间,在计算大数阶乘时,无论是运算速度还是存储空间,链表都不是最佳方案.

收稿日期:2018-03-15;修回日期:2018-10-15;责任编辑:孙瑶
基金项目:卫星导航系统与装备技术国家重点实验室开放基金(No. SKX182010927)

利用整数数组来构建一个可以表示尽可能大的整数,该整数的理论上限取决于存储空间,通常为了直观,数组中的每一个元素都用十进制数表示. 文献[4,7]利用了动态数组来存储和表示大数的阶乘,且每个数组元素表示的位数不同. 虽然这能够精确表示大数的阶乘,但是其运算速率一直是瓶颈,且对问题规模缺少自适应性,即或是因为预设一个足够大的静态存储空间而造成存储资源的浪费;或是因为存储空间不够而导致问题规模足够大时溢出.

文献[8]全面地介绍了阶乘算法的发展,包括斯特林公式近似法,除此之外,也提到其它五种高精度阶乘计算方法. 如 Split Recursive 算法,简单易行,且没有质因数的分解;Prime Swing 算法,基于摇摆数的概念,通过摇摆数的质因数乘式来计算整数 n 的阶乘;Moessner 算法,只用加法来实现,但缺少实际意义;Poor Man 算法可以使用阶乘查表法,可以快速计算 10000 的阶乘;Parallel Prime Swing 算法,是对 Prime Swing 算法的改进,使用并发程序设计和多核处理器来提高计算速率.

在前人研究的基础上,本文提出一种零误差大数阶乘算法. 该算法可在硬件资源有限的情况下,根据问题规模,自动分配最优的存储空间,给出没有误差的计算结果;同时利用 FPGA 并行处理的优势,模拟多核处理器并行处理过程,大大提高计算速率.

2　算法设计过程

算法提出的总框图如图 1 所示,相应的硬件测试平台如图 2 所示. 所提出的算法包括三部分:首先是问题规模的估计,即计算阶乘 n 的位数 NUM,显然 n 或者 NUM 越大,问题的规模越大;然后根据位数 NUM 的大小优化存储空间;最后使用并行计算的方法得到整数 n 的阶乘结果. 通过使用硬件描述语言,即 Verilog HDL 语言,借助 FPGA 平台实现该阶乘算法. 过程如下:利用锁相环 PLL 来管理全局时钟,并控制每一个 Always 语句,实现并行计算;将计算结果存储在同步动态随机存储器(SDRAM)中;FIFO 用来从 SDRAM 读取计算结果,并在液晶显示器上(LCD)显示,这里引入 FIFO,是为了提高对计算结果的读取速率. 下面将对算法各部分进行具体介绍.

2.1　问题规模估计

根据阶乘多项式的定义,n 的阶乘如式(1)所示:

$$n! = 1 \cdot 2 \cdot 3 \cdot \cdots \cdot n \tag{1}$$

其中,n 是正整数. 对式(1)两边同时以 10 为底取对数,可以准确估计 n 阶乘的位数 NUM,如式(2)所示:

$$NUM = \log_{10}1 + \log_{10}2 + \cdots + \log_{10}n \tag{2}$$

当 n 比较大时,对 NUM 的确定可能是耗时的. 可以利用斯特林公式如式(3)所示,来解决这一问题:

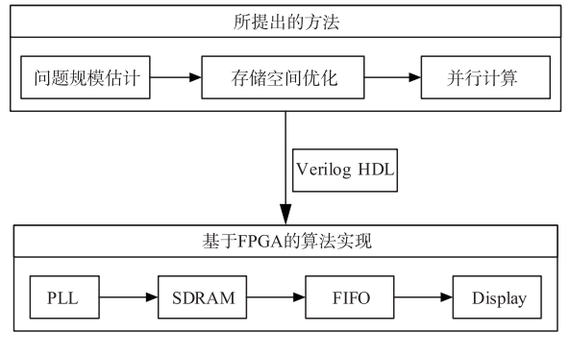

图1　所提出算法的总框架

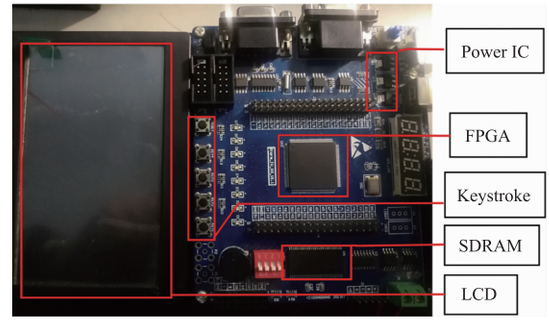

图2　算法测试的硬件平台

$$n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n \tag{3}$$

这里,e 是自然对数. 对式(3)两边同时以 10 为底取对数,得式(4):

$$\log_{10}n! = \frac{1}{2}\log_{10}(2\pi n) + n\log_{10}\left(\frac{n}{e}\right) \tag{4}$$

基于式(4),问题规模 NUM 的解析表达式如下:

$$NUM = \left\lfloor 1 + \frac{1}{2}\log_{10}(2\pi n) + n\log_{10}\left(\frac{n}{e}\right)\right\rfloor \tag{5}$$

2.2　存储空间优化

在 Verilog HDL 语言中,十进制数可以由一个数组来表示,如 a 为数组名,M 为数组大小如式(6)和式(7)所示:

$$a[M] = a_{M-1}a_{M-2}\cdots a_0 \tag{6}$$

$$\gamma * w * M = NUM \tag{7}$$

$$\underbrace{99\cdots9}_{w} * \underbrace{99\cdots9}_{w} \leq 2^{64} - 1 \tag{8}$$

$$w = \min\{\underbrace{99\cdots9}_{w} * \underbrace{99\cdots9}_{w} - 2^{64} + 1\} \tag{9}$$

其中,$a_i, i \in (0, M-1)$ 表示一个数组元素代表一个 w 位十进制数,γ 是与数据在 SDRAM 存储时数据的对齐方式相关的参数,通常 γ 的典型值为 1. 因此,数组 a 可以表示一个有 $\gamma * w * M$ 位的十进制数. 考虑到 LCD 硬件资源有限,我们假设问题规模的天花板是 $n = n_{max} = 25000$,即这里 n 是整数,且 $n \in (0, 25000)$. 事实上,如图 3 所示,针对同一个 w,计算时间随着问题规模的增大成倍增加. 值得注意的是,针对同一个问题规模,计算

时间随着 w 的增大而显著减少. 其原因不难理解,w 越大,数组中每一个元素的权重越大,相当于其能装载数据而不至于溢出的能力越强,数组中对应的各元素刷新的平均频率也越低,这样就加快了计算速度. 然而,w 不能一直大下去,其理论上限由式(8)和式(9)决定. 同时,式(6)~(9)也表明,数组中各元素表示的位数越多,针对相同的问题规模,其所需要的存储空间越小. 这样,存储空间就可以根据问题规模的大小自适应的分配,实现优化. 至此,所有关于存储空间和问题规模的参数都已经确定. 下面即开始利用这些参数,和并行计算方法,求 n 的阶乘.

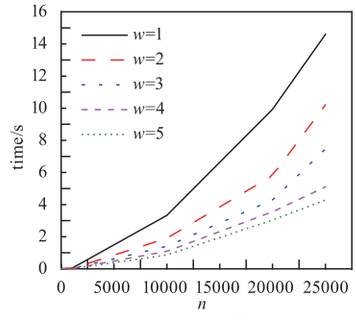

图3 不同w下的解算时间

2.3 并行计算

图 4 描述了给定总存储空间(NUM)时,w 和 M 关系的示意图,其中相同颜色的单元格表示 w,表示该位置存储一个 w 位的十进制数;M 不仅表示数组的大小,还表示数组内部各相邻元素之间的逻辑关系,即权重,示意图中为突出权重的不同而用不同的颜色表示. 如:从右到左,红色的表示的权重为 10^{0*w},蓝色的表示的权重为 10^{1*w},紫色的表示的权重为 10^{2*w},依次类推. 明白这一点,对基于存储空间划分的并行计算的理解非常重要. 严格来讲,所提出的方法不是并行计算,而是一种类并行计算方法,这里只是利用 FPGA 并行计算的优势. Verilog HDL 的 Always 语句是并行语句,不同权重单元同时进行阶乘计算,在权重单元溢出时再进行迭代更新. 在每个子 processing 中,可以采用最简单的阶乘递推算法,只是在溢出时,将溢出值更新到后一个存储空间,同时对当前存储空间求模,具体如图 5 所示.

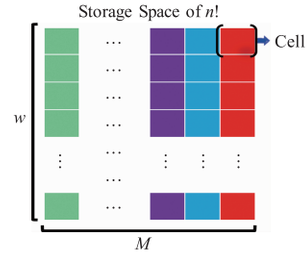

图4 在给定总存储空间(NUM)时,描述w和M关系的示意图

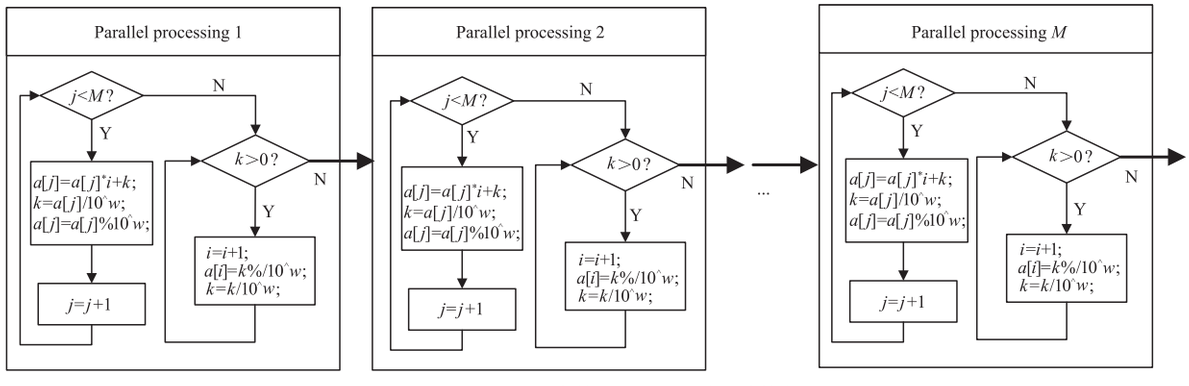

图5 基于图4所示的存储空间划分的并行计算方法

3 实验验证和讨论

事实上,关于存储空间优化的仿真和讨论在2.2节已经涉及,下面主要是对整个算法的运算速度和精度进行对比分析. 可以利用 Verilog HDL 的时序控制和计时器获得提出算法的运行时间. Verilog HDL 演示实验如图 6 所示,为便于展示,以 10 的阶乘为例. 图 6 所示结果模拟了在时钟控制下,阶乘计算结果更新的机制. 此外我们测试了在 $n=10000$ ($NUM=35660$)时各算法的时间效率,如图 7 所示. 显然,相比于已有的阶乘算法,所提出的算法用时最少. 为了测试所提出的算法在不同复杂度下的时间效率,可以通过改变 n 的大小来改变复杂度. 我们测试了不同复杂度下各算法的时间效率,测试结果如图 8 所示. 该对比反映了不同复杂度下,所提出的算法较传统算法有较大改善. 同时,可以定量分析在各测试复杂度下,相比于对比算法,所提出的算法的时间效率提高 40% 以上.

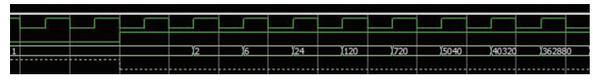

图6 VERILOG HDL阶乘仿真

图 9 表明所提出的算法计算的位数与真实值完全

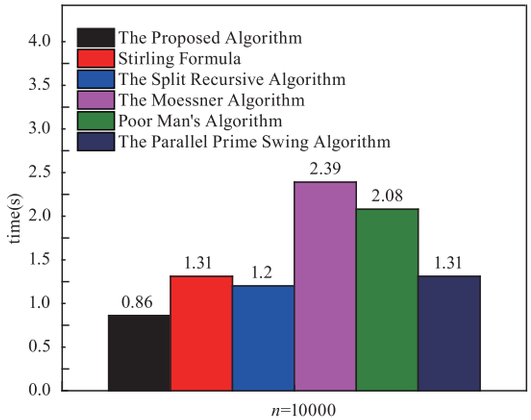

图7　所提出的算法与现有算法的比较

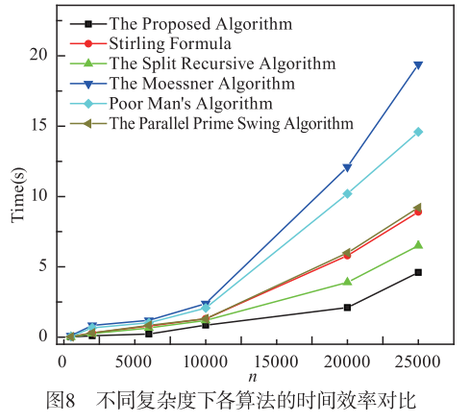

图8　不同复杂度下各算法的时间效率对比

重合,而其它算法均出现不同程度的截断现象,因此所提出的算法可以精确到个位,即零误差. 此外,利用本文提出的算法,基于 C + + 语言,我们开发出了多功能阶乘计算器,如图 10 所示.

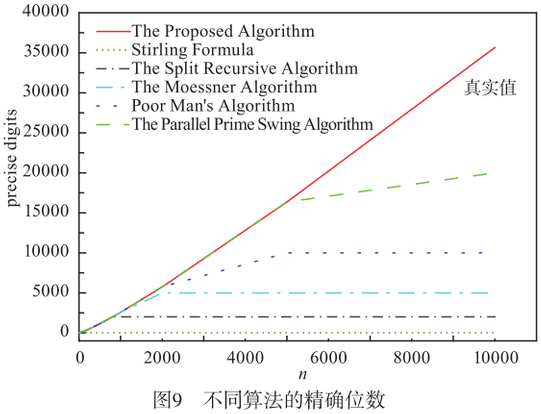

图9　不同算法的精确位数

最后,有必要讨论一下大数阶乘的应用. 学者们对大数阶乘算法的研究从未停止,因为阶乘作为一种常见的数学运算形式,与工业与经济领域的排列组合和优化、泰勒多项式和自然对数的高阶近似等问题息息相关. 其应用领域也十分广泛,具体如下所述.

图10　多功能阶乘计算器

直接应用领域是高精度乘法器[9],多功能计算器等科学计算器,其解决方案一般是通过 C/C + +/Matlab 以及其它编程语言,开发上位机软件;解决方案 2 是利用嵌入式、FPGA 等开发平台对上位机软件进行硬件实现.

间接应用领域包括以下四点.

(1)数据传输与加密领域[10],将大数阶乘作为产生秘钥的核函数.

(2)人工智能方面,如利用卷积神经网络训练高精度目标跟踪算法[11].

(3)大数据方面,如高精度多维计数器[12]、海量数据反演算法[13]等.

(4)高精度导航定位领域[14~16];尤其是捷联惯性导航,位置误差会随时间累积,且其与姿态误差的三次方成正比,因此需要研究高精细姿态计算方法. 大数阶乘可以作为一种惩罚因子,抑制误差的漂移,提供长期的位置服务.

关于各应用领域的解决方案,在底层实现上,大数阶乘往往依靠数组、链表、堆等存储,采用近似方法、快速傅里叶变换方法、以及并行计算方法等进行实现;在间接应用领域,大数阶乘的计算结果往往取导数、加权等,在建模和仿真中作为算法处理中的核心步骤,解决各种工程实际问题.

4　结论

本文提出一种零误差大数阶乘算法,在硬件资源有限的情况下,可以估计问题规模,并根据问题规模自动分配最优的存储空间,给出零误差的计算结果,同时利用 FPGA 并行处理的优势,模拟多核处理器并行处理,大大提高计算速率. 此外,基于该算法开发多功能阶乘计算器上位机,测试结果表明,所提出的算法具有较好的时空效率,可以应用在如大数阶乘计算器、组合与优化、导航与定位等诸多领域.

参考文献

[1] SCHEID F J. Schaum's Outline of Theory and Problems of

Numerical Analysis[M]. New York, NY, USA: McGraw-Hill, 1989. 7 – 50.
[2] WANG C, YUEH, YIN C, et al. Arithmetic operations beyond floating point number precision [J]. International Journal of Computational Science and Engineering, 2010, 6(3): 206 – 215.
[3] MEZMAZ M, LEROY R, MELAB N, et al. A multi-core parallel branch-and-bound algorithm using factorial number system[A]. Proceedings of IEEE 28th International Parallel and Distributed Processing Symposium[C]. Phoenix, AZ, USA: IEEE, 2014. 1203 – 1212.
[4] 马旭,王大勇. 大数的阶乘与自然对数的超高精度求解[J]. 计算机与现代化,2017,(3):51 – 53.
MA X, WANG D Y. Factorial of large number and super high precision solution of natural logarithm[J]. Computer and Modernization, 2017, (3): 51 – 53. (in Chinese)
[5] UY R L, MARCOS N. On the complexity of implementation of recursion on Factorial and Fibonacci algorithms using RISC-based MIPS64 instruction set architecture[A]. Proceedings of TENCON 2015 IEEE Region 10 Conference[C]. Macao, China: IEEE, 2016. 1 – 6.
[6] FATEMAN R J. Comments on Factorial Programs, 2006 [OL]. https://www. pdf-archive. com/2016/10/07/ factorial-pdf0/factorial-pdf0. pdf, 2018-10-16.
[7] 英昌盛,周喜龙. 大整数乘法的数据结构及算法选择探究[J]. 长春工业大学学报,2008,29(2):204 – 207.
YING C S, ZHOU X L. Data structure and algorithm selection of big integer multiplication[J]. Journal of Changchun University of Tchnology (Natural Science Edition), 2008, 29(2): 204 – 207. (in Chinese)
[8] PETER L. Factorial Prime Swing[OL]. http://www. luschny. de/math/factorial/, 2018-10-16.
[9] 张柳,崔晓平,董文雯. 高性能并行全冗余十进制乘法器的设计[J]. 电子学报,2018,46(6):1519 – 1523.
ZHANG L, CUI X-P, DONG W-W. High-performance parallel fully redundant decimal multiplier[J]. Acta Electronica Sinica, 2018, 46(6): 1519 – 1523. (in Chinese)
[10] 李俊志,关杰. 非线性反馈移存器型序列密码的完全性通用算法研究[J]. 电子学报, 2018, 46(9): 2075 – 2080.
LI J-J, GUAN J. Universal algorithm of full diffusion of stream cipher based on nonlinear feedback shift register [J]. Acta Electronica Sinica, 2018, 46(9): 2075 – 2080. (in Chinese)
[11] 李康,李亚敏,胡学敏,邵芳. 基于卷积神经网络的鲁棒高精度目标跟踪算法[J]. 电子学报,2018,46(9):2087 – 2093.
LI K, LI Y-M, HU X-M, SHAO F. A robust and accurate object tracking algorithm based on convolutional neural network[J]. Acta Electronica Sinica, 2018, 46(9): 2087 – 2093. (in Chinese)
[12] 李玮,张大方,黄昆,等. 面向大数据处理的高精度多维计数布鲁姆过滤器[J]. 电子学报,2015,43(4):652 – 657.
LI W, ZHANG D-F, HUANG K, et al. Accurate multi-dimension counting bloom filter for big data processing[J]. Acta Electronica Sinica, 2015, 43(4): 652 – 657. (in Chinese)
[13] 蒋帅,汪丙南,李银伟. 一种用于 InSAR/INS 组合导航的姿态角反演方法[J]. 电子学报,2018,46(3):513 – 519.
JIANG S, WANG B-N, LI Y-W. An inversion method of the attitude for InSAR/INS integrated navigation[J]. Acta Electronica Sinica, 2018, 46(3): 513 – 519. (in Chinese)
[14] LIU G, SHI L. Adaptive algorithm of magnetic heading detection [J]. Measurement Science and Technology, 2017, 28(11): 115101.
[15] LIU G, SHI L-F, XUN J, CHEN S, ZHAO L, SHI Y. Orientation estimation algorithm based on multi-source information fusion[J]. Measurement Science and Technology, 2018, 29(11): 115101.
[16] 刘公绪,史凌峰. 室内导航与定位技术发展综述[J]. 导航定位学报,2018,6(2):7 – 14.
LIU G X, SHI L F. An overview about development of indoor navigation and positioning technology[J]. Journal of Navigation and Positioning, 2018, 6(2): 7 – 14. (in Chinese)

作者简介

刘公绪 男,1992 年出生于河南兰考. 2015 年毕业于中北大学仪器与电子学院,同年保送至西安电子科技大学电子工程学院,现为硕博连读生. 研究方向为惯性导航、室内定位、多传感器融合算法等.
E-mail: liugx@ stu. xidian. edu. cn

史凌峰 男,1970 年生于陕西省宝鸡市. 1995 年,分别于 1995 年、2003 年和 2008 年从中国计量学院获得无线电学士、西安电子科技大学获得电路与系统硕士和博士学位. 现为西安电子科技大学电子工程学院教授/博导. 主要研究兴趣为导航与定位、射频与微波技术等.
E-mail: lfshi@ mail. xidian. edu. cn

辛东金 男,1986 年出生于山东济宁. 西安电子科技大学超高速电路与电磁兼容教育部重点实验室博士研究生,研究方向为物联网通信及滤波器.
E-mail: xindj@ stu. xidian. edu