

重力场反演中大型矩阵GPU加速运算的实现

周予涵¹, 简念川², 陈从颜¹

(1. 东南大学自动化学院, 南京 210096; 2. 中国科学院上海天文台, 上海 200030)

摘要: 针对重力场反演中大型矩阵解算任务量大、解算时间长的问题, 提出一种基于多片GPU并行的矩阵运算方法, 该方法将多片图形处理器 (Graphics Processing Unit, GPU) 和CUDA (Computer Unified Device Architecture) 相结合, 能实现密集运算的高度并行化, 通过加速矩阵乘法和矩阵求逆的运算, 极大缩短了行星重力场反演中大型矩阵求逆所需的时间, 计算速率为使用CPU计算时的191倍, 并且计算的精度较高, 反演的误差数量级在 10^{-17} 。将其应用到重力恢复与内部实验室 (Gravity Recovery and Interior Laboratory, GRAIL) 月球重力场反演的计算中, 在计算截断大小为50阶次和180阶次的矩阵时, 比使用CPU的运算方法时间分别缩短了94.63%、99.51%, 并在武汉大学高性能计算平台实现了900阶次矩阵的运算。采用的方法可有效缩短传统计算模型所需的时间, 从而有助于建立高阶次、高精度的重力场模型。

关键词: 行星重力场反演; GPU运算; 并行计算; CUDA

中图分类号: P222

文献标识码: A

文章编号: 2096-9287(2024)06-0587-07

DOI: 10.15982/j.issn.2096-9287.2024.20240012

引用格式: 周予涵, 简念川, 陈从颜. 重力场反演中大型矩阵GPU加速运算的实现[J]. 深空探测学报(中英文), 2024, 11(6): 587-593.

Reference format: ZHOU Y H, JIAN N C, CHEN C Y. Implementation of GPU-accelerated computation of large matrix in gravity field inversion[J]. Journal of Deep Space Exploration, 2024, 11(6): 587-593.

引言

近20年来, 卫星重力测量和卫星测高技术迅速取得了长足的发展, 这为高精度、高分辨率的重力场模型提供了大量的观测数据。地球重力场、月球重力场、火星重力场等的研制对于大地测量、行星勘测等相关领域以及交叉学科都有很大的应用潜力和重要性。随着对重力卫星跟踪技术的研究不断进步, 行星重力场模型经历了一个从低阶次到高阶次, 从低分辨率到高分辨率^[1]的发展过程, 以月球为例, 在重力恢复与内部实验室 (Gravity Recovery and Interior Laboratory, GRAIL)^[2]任务, 阶次最高的重力场模型可达1 500阶次^[3], 其最大分辨率可达3.6 km, 取得了较好的成果。因而对于重力场反演的研究, 改善重力数据并进行处理是极其重要的。

重力数据处理是一项计算密集型任务, 受困于解算任务量大、时间长的问題^[4], 往往需要超级计算机求解。目前, 国内重力数据并行处理主要围绕设计矩阵的计算、法方程的构建与求解等方面进行研究, 为获取反演参数的最佳估值, 需对大型法方程进行求解运算。因此为提高重力场反演计算效率, 很有必要缩短

大规模矩阵求逆计算所需的时间。

国内外诸多重力研究机构都建立了超级计算机、计算机集群等平台, 对重力数据处理进行研究, 美国得克萨斯大学空间研究中心 (Center for Space Research, CSR) 和德国地学研究中心 (Geo Forschungs Zenrum, GFZ) 等^[4-5]都是其中之一。格拉茨技术大学 (Graz University of Technology) 建立了计算机集群系统 “Graz Beowulf Cluster”, 完成了对地球重力场与海洋环流卫星 (Gravity field and steady-state Ocean Circulation Explorer, GOCE) 数据处理的任务^[6]。利用代尔夫特理工大学的CRAY T3E超级计算机, Klees 和他的团队成功实现了地球重力场的场景图生成 (Scene Graph Generation, SGG) 数据恢复^[7]。慕尼黑工业大学、格拉茨技术大学、波恩大学、奥地利科学院联合开发了 “Sub-processing Facility (SPF) 6000” 软件, 用于GOCE卫星数据的并行处理^[8-9]。武汉大学、同济大学、中国科学院计算与地球动力学实验室等国内多家研究机构都有参与重力数据处理的研究。

对于重力数据的并行处理, 国内外的研究人员也采取了多种方法。Baboulin等^[10]运用MPI/OpenMP进行

混合编程, 实现了重力场反演中大规模线性最小二乘法的求解; Lee等^[11]开发了基于多核计算机集群的并行程序, 在减少重力场模型解算大地水准面高的时长方面做出了贡献; Couder-Castañeda等^[12]基于OpenMP和MPI的混合设计, 实现了重力矢量分量和张量分量并行计算。国内基于MPI和OpenMP方向开展研究的学者较多。2005年, 王正涛等^[13]首次将基于MPI矩阵并行运算应用到重力场恢复; 周浩等^[14-15]分别就设计矩阵的计算、法方程的构建与求解进行并行效率分析, 在利用最小二乘直接法反演高阶次位系数过程中, 基于MPI实现了并行读写、并行计算和分块存储; 陈秋杰等^[16]比较分析了高阶地球重力场反演中OpenMP、MKL的并行运行的效率; 吴怿昊等^[17]基于ScaLAPACK并行库, 实现了法方程并行构建及矩阵求逆。得益于GPU多核心、高集成的特点, 密集运算能实现高度并行化, 单片GPU乃至多片GPU并行的计算速度给矩阵运算速度进一步提升创造了空间。

针对目前处理高阶重力场模型需要的时间较长, 处理速度比较慢的问题, 本文提出采取CUDA (Computer Unified Device Architecture) 和多片图形处理器 (Graphics Processing Unit, GPU) 并行的方式, 对重力场反演中的大型矩阵进行求逆运算, 以进一步提高高阶次重力场模型反演的速度。

1 卫星重力场模型反演

由于卫星内部物质密度不均匀, 地表形状也不规则, 很难用一个理想的数学模型来描述, 为此采取有限元方法进行更稳定、更有效的重力问题计算, 即将观察平面分成一组矩形元素 $x_i = \xi_i, y_i = \eta_i (0 \leq i < N, 0 \leq j < M)$, 并计算场 $\Delta g(x_0, y_0)$ 作为平行六面体元素重力效应的总和。

$$\Delta g(x', y') = \gamma \Delta \sigma \sum_{ij} \Delta g_{ij}(x', y') \quad (1)$$

在重力场元的计算中, 边界重力效应计算实际上被有限物体的重力效应计算所取代, 因此可使用任何水平定位的平面来代替渐近线。重力场元各纬度带之间的计算是相互独立的, 适合采用数据并行的方法运算。并行算法的实现中包含了两个嵌套循环, 一为迭代观察点, 另一个迭代离散点, 其运算过程与CUDA的内部结构相似, 即将运算分为块和线程两个部分, 在 $N \times M$ 个线程中对 $N \times M$ 个块进行 Δg 的计算, 然后汇总每个块内所有线程的结果, 则块结果的组合将是所需的重力效应。

重力场模型的基本观测方程^[18-19]为

$$\Delta g(\gamma, \phi, \lambda) = \frac{GM}{R^3} \sum_{n=2}^{n_{\max}} (n+1)(n+2) \left(\frac{R}{\gamma}\right)^{n+3} \sum_{m=0}^n (\bar{C}_{nm} \cos m\lambda + \bar{S}_{nm} \sin m\lambda) \bar{P}_{nm}(\sin \phi) \quad (2)$$

其中: Δg 为行星的任意一点 (γ, ϕ, λ) 处的经过椭圆改正、大气改正、二阶径向梯度改正等运算后满足调和边值问题的格网平均重力异常值; GM 为地心引力常数; R 为行星的平均半径; n_{\max} 为模型展开的最大阶次; \bar{P}_{nm} 为规格化的缔合勒让德函数; $\bar{C}_{nm}, \bar{S}_{nm}$ 为待求的一组重力场模型位系数。

根据基本观测方程构建设计矩阵、建立法方程并解算, 从而计算重力场模型, 是进行重力场反演的重要步骤。

2 大型矩阵GPU加速运算

针对大型数据的处理, 采用在GPU密集线性代数加速的方式。从矩阵乘法、线性计算等方面将GPU和CUDA结合, 进行加速计算。

2.1 矩阵乘法的加速

优化矩阵乘法的主要工具是循环平铺技术。平铺将一个循环替换为两个循环: 内部循环将循环计数器增加一个, 外部循环将循环计数器增加一个平铺因子。矩阵乘法的平铺利用了曲面体积效应, 即在 $O(n^2)$ 个数据执行 $O(n^3)$ 个浮点运算。并应用循环展开技术, 用单个直线代码块 (单个基本代码块) 替换最内层的3个循环, 其目的是为减少循环的代价 (增加循环计数器、推进数据指针和分支的开销), 并通过创建独立指令序列来增加指令级的并行性, 填充处理器的管道。

在GPU的情况下, 矩阵被覆盖在线程块的2D网格上, 每个线程块负责计算单个块。在单个线程块的操作之外, 两个外部循环消失, 只剩下一个循环——沿着 k 维逐块前进的循环, 如图1所示。

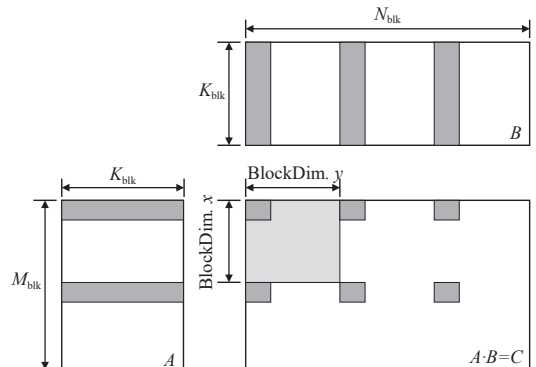


图1 矩阵乘法在GPU中运算示意图

Fig. 1 Diagram of matrix multiplication operation in GPU

2.2 矩阵求逆的加速

在对重力场反演中的协方差矩阵进行处理的过程中, 将矩阵求逆分解为两大任务执行: LU分解、利用LU分解的结果计算逆矩阵, 具体求解过程如图2所示。将整体求解分解成多个任务, 在CUDA分配动态内存, 将适合高效GPU计算的大型、高数据并行任务分配到GPU执行。采用两层嵌套并行的方式, 其中粗粒度的并行性是通过将计算分成多个块, 以便在GPU之间并发执行(遵循PLASMA的框架); 细粒度的并行性通过将工作负载划分到一个tile进一步提供, 以便在GPU高效计算, 而且在某些情况下, 还可以从使用GPU和CPU的混合计算中获益。以此实现GPU的并行计算, 充分发挥多核GPU加速器系统的高性能和可扩展性。同时, 对矩阵进行LU分解, 可有效降低矩阵的条件数, 改善在求解线性方程组时可能存在的病态矩阵问题。

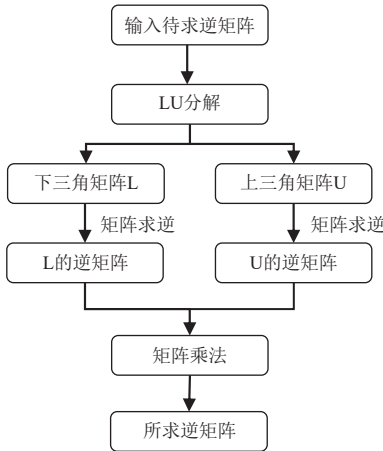


图 2 矩阵求逆过程图
Fig. 2 Solve process chart of matrix inversion

2.3 矩阵分块

在重力场模型中, 计算阶次达到180阶次, 所需内存即为3.99 GB, 而900阶次的矩阵, 所需要的内存达到了2.4 TB, 直接计算的方法是不可取的。为计算阶次更大的法矩阵时, 为减少GPU内存的耗用量, 需对矩阵进行分块处理, 通过并行化处理子矩阵块, 提高计算效率, 实现矩阵的分块计算。

计算过程中, 需对求逆矩阵进行行列划分, 采用内存映射的方法读取矩阵数据, 根据硬件结构和性能将矩阵分为 $n \times n$ 个子矩阵, 如式(3)所示。对分块后的子矩阵进行求逆, 减少单次计算所需要的内存。按顺序读取子矩阵, 通过LU分解计算每个子矩阵块的逆, 待所有子矩阵计算完毕, 将计算得到的子矩阵块合并为整个矩阵的逆矩阵, 从而有效减少计算所需的内存损耗。

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} & \cdots & A_{1n} \\ A_{21} & A_{22} & A_{23} & \cdots & A_{2n} \\ A_{31} & A_{32} & A_{33} & \cdots & A_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & A_{n3} & \cdots & A_{nn} \end{bmatrix} \quad (3)$$

3 实验验证分析

本研究采用MAGMA 2.6.2版本的软件进行计算。MAGMA是目前在GPU加速密集线性代数的最佳设计和实现实践^[20]。任务执行是通过静态调度系统在多核GPU和GPU加速器混合系统的计算组件调度的。

本文的实验平台为上海天文台服务器和武汉大学的高性能计算平台。由于硬件相关性能的限制, 仿真实验和部分数据测试在上海天文台服务器进行, 高阶重力场反演测试在武汉大学的高性能计算平台进行。

3.1 仿真数据测试与分析

仿真实验的平台为上海天文台的服务器, 搭载了linux 4.4.0-189-generic x86_64操作系统, 所用的GPU型号为Tesla K80, 服务器一共有4片GPU, 编译语言为C++和Fortran。

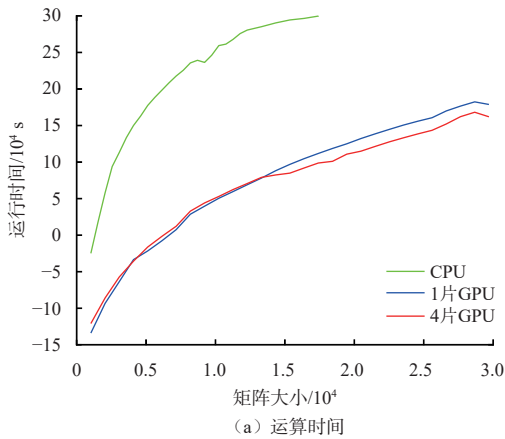
实验对随机生成的30 000 × 30 000大小的稠密矩阵进行求逆, 采用3种方案: 一是运用LAPACK库函数, 采用CPU进行矩阵运算; 二是通过MAGMA的dgetrf和dgetri两个函数对矩阵进行求逆, 该过程在一片GPU进行; 三是将方案二中所描述的过程分为4个线程, 分别在4片GPU同时进行, 用GPU并行的方式, 达到缩短计算所需时间的效果。实验结果如表1所示, 统计了3种算例方案所需的计算时间和每秒所执行的浮点运算次数, 可以看出在该算例中, 使用GPU的速度远大于使用CPU进行运算的速度, 单片GPU运算速度可以达到CPU的约119倍, 加速计算效果显著。而对于多片GPU并行的效果相比于单片GPU也有所提升, 运算时间缩短了37.53%。这里选用了Gflops指标衡量在本算例中的GPU的计算能力, 从表1可以看出, 多GPU并行运算将GPU的计算能力提高了13.11%, 在大规模矩阵的运算中, 多线程的运算使得GPU的计算能力能够更好地发挥出来。

表 1 矩阵求逆运算效率
Table 1 Efficiency of matrix inversion operation

硬件	计算时间/s	指标量Gflops	加速因子
CPU	8 449.785 81 (140 + minutes)	—	1x
1 GPU	70.928 47	761.311 39	119x
4 GPU	44.308 85	1 218.702 32	191x

此外, 本文对不同规模矩阵的处理效率进行了测试。针对不同大小的稠密矩阵, 上述3种方案的效果如

图3所示,对于大型矩阵, GPU进行加速运算的方法有绝对的优势,大大缩短了运算时间,提高了计算效



率。而就GPU并行计算而言,在矩阵规模较小的情况下,单片GPU要略优于多片GPU并行运算。

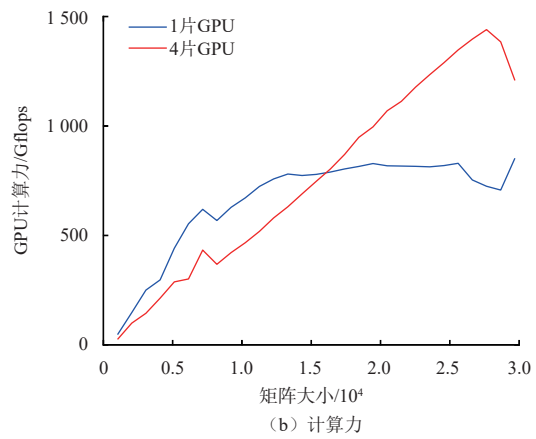


图3 CPU、单片GPU、4片GPU对比图

Fig. 3 Comparison of CPU single GPU and 4 GPUs

硬件方面,由于服务器的散热系统不够完善,在多片GPU并行运行的情况下,温度较高,其计算潜力未正常发挥;软件方面,矩阵较小时运算时间较短,单片GPU的调度较快,4片GPU运算则存在一定的通信时间。而在矩阵大于 15000×15000 的算例中,多片GPU并行计算具有显著的优势,能有效缩短计算的时间,且提高了GPU的运算能力,运算效率得到了提升。

3.2 实际数据测试与分析

本文采用的实际数据来源于NASA的GRAIL月球重力场探测任务。为JPL月球重力场GL0900D的相关数据(pds-geosciences.wustl.edu/-grail/grail-l-lgrs-5-rdr-v1/grail_1001/shbdr/),这是一个900阶的球谐模型。所测得的重力场包括主要任务(2012年3月1日16:30—5月29日16:36)和GRAIL跟踪数据的扩展任务(2012年8月30日16:20—12月14日20:56)。实验选取了其中截断阶次分别为50和180阶次的数,数据中包含球谐系数以及其与其它参数的协方差矩阵,对其中的协方差矩阵进行求逆,并进行重力场反演的相关操

作,得到相应结果。对协方差矩阵进行求逆所花费的时间如表2所示。

表2 月球重力场数据求逆运算所需时间

Table 2 Time required to invert the lunar gravity field data				
模型	截断阶次	规模	计算时间/s	
			CPU	GPU
GL0900D	50	$2\,598 \times 2\,598$	4.641 100	0.248 80
	180	$32\,758 \times 32\,758$	11 292.211 238 (188 + minutes)	55.601 01

从表2中可以看出,与CPU运算相比, GPU并行的运算速率分别提高了94.63%, 99.51%,在实际的重力场反演运算有极大的运用价值。

为衡量求逆算法的精度,采取原矩阵 N 乘以求得的逆矩阵 N^{-1} 并与单位阵 E 相减,得到求逆误差,记为 n_error ,计算结果计入表3中。结果表明,求逆误差的量级为 10^{-9} 左右。由误差传播定律,可认为该部分误差是由于数据发送接收误差、计算机截断误差等引起的,对求逆精度的影响极小。

表3 月球重力场数据求逆运算精度分析

Table 3 Precision analysis of inversion operation for lunar gravity field data

误差类型	样本大小	最大值	最小值	平均值	均方根
n_error	2 598	1.07×10^{-6}	0	2.63×10^{-11}	2.30×10^{-9}
	32 758	1.35×10^{-6}	0	4.89×10^{-10}	3.52×10^{-9}
q_error	2 598	$9.094\,95 \times 10^{-11}$	$3.970\,47 \times 10^{-23}$	$3.500\,76 \times 10^{-14}$	$1.784\,35 \times 10^{-12}$
	32 758	$1.209\,63 \times 10^{-11}$	$1.075\,33 \times 10^{-23}$	$3.692\,65 \times 10^{-14}$	$6.683\,33 \times 10^{-12}$

为了衡量反演运算的精度,根据法方程,即

$$(A^T P A) X = A^T P L^b \quad (4)$$

其中: A 为设计矩阵; P 为单位矩阵; X 为要求的球谐

系数; L 为观测值向量。

采用矩阵求逆的方法,计算法方程的解 X ,即球谐系数。将求得的球谐系数与实际数据给出的球谐系数

进行对比, 求得其中的误差, 记为 q_error , 计入表3中 (其中, 实际数据中给出的球谐系数的数量级为 $10^{-5} \sim 10^{-9}$)。其误差结果如图4所示。

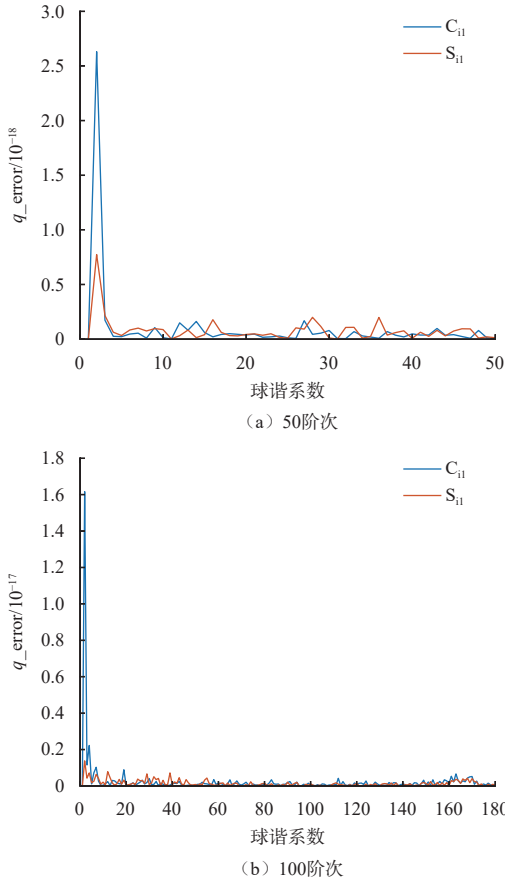


图 4 球谐系数的误差 (以 C_{ii}, S_{ii} 为例)

Fig. 4 Error of spherical harmonics coefficients (Take C_{ii}, S_{ii} as examples)

结果表明, 反演误差的量级为 10^{-17} 左右, 误差的重力场径向平均功率谱如图5所示, 反演误差造成的重力场误差几乎可以忽略不记。上述计算结果验证了本文运算方法的可靠性和准确性。

3.3 高阶次卫星重力场反演测试

为测试GPU并行的计算方式在高阶重力场反演中的表现, 本文在武汉大学的高性能计算平台进行了高阶重力场求逆运算实验。服务器搭载了4片Nvidia Tesla A100型号的GPU, 每片GPU内存大小为40 GB, 对900阶次的月球重力场中的协方差矩阵求逆, 实际数据同样来源于JPL月球重力场模型GL0900D。实验中对 $811\ 798 \times 811\ 798$ 的对称矩阵分块, 共分为400个子矩阵, 大小为 $40\ 589 \times 40\ 589$, 采用分块矩阵求逆的方法对整体大矩阵进行求逆, 计算结果如表4所示。

本文中的矩阵求逆是通过矩阵进行LU分解, 并求出逆矩阵, 计算复杂度为 $O(n^3)$, 武汉大学高性能计

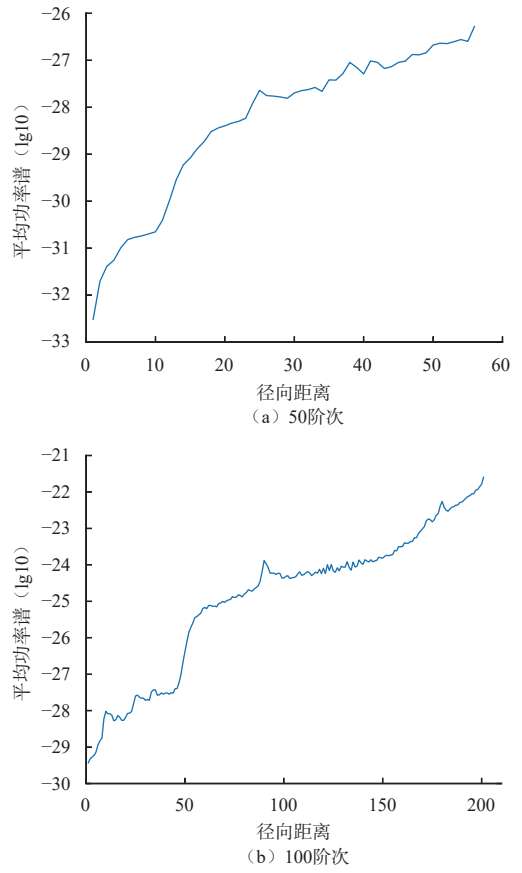


图 5 反演误差的重力场径向平均功率谱

Fig. 5 Radial average power spectrum of gravity field of inversion error

表 4 高阶月球重力场数据求逆运算所需时间

Table 4 Time required to invert high level lunar gravity field data

模型	规模	计算时间/s	
		4 GPUs	
GL0900D	$40\ 589 \times 40\ 589$	18.348 04	
	$811\ 798 \times 811\ 798$	8 177.306 54 (136.288 44 min)	

算平台搭载的CPU为AMD EPYC 7543, 其主频为2.8GHz, 核心数为32, 线程数为8, 理论的Gflops为5 734.4。

$$T = \frac{O(n^3)}{X} \quad (5)$$

其中: T 为运算时间; X 为CPU的Gflops, 可粗略估算使用CPU计算900阶次矩阵求逆所需要的时间54 d, 使用GPU并行加速计算的方法可使计算速度提高570倍左右。由此可见, GPU并行计算适用于高阶重力场计算, 可极大地缩短计算所需的时间。

采用上文计算球谐系数的方法, 求得本文算法运算900阶次重力场所得的球谐系数误差 q_error , 得到 q_error 的最大值为 $2.983\ 14 \times 10^{-10}$, 最小值为0, 平均值为 $3.679\ 07 \times 10^{-16}$ 。均方根为 $3.310\ 93 \times 10^{-13}$, 其运算

所得球谐系数的计算误差功率谱图如图6所示,反演误差造成的重力场误差可以忽略不计,因此GPU并行运算在高阶重力场计算也具有可靠性和准确性。

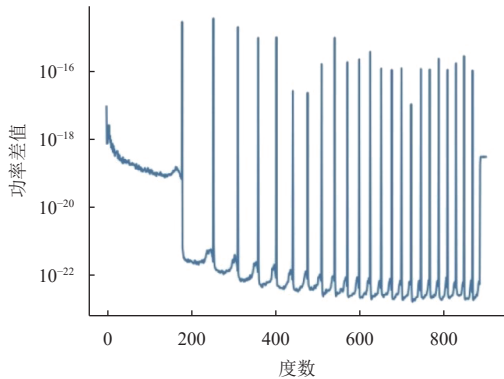


图6 900阶次球谐系数计算误差功率谱图

Fig. 6 900-order spherical harmonic coefficient calculation error power spectrum

4 结论

本文针对重力场反演中大型矩阵解算任务量大、解算时间长的问题,实现了基于多片GPU并行的大型矩阵运算方法,将GPU和CUDA相结合,能够实现密集运算的高度并行化,极大缩短了行星重力场反演中大型矩阵求逆所需的时间。在仿真计算中,采用4片GPU并行的计算方式是CPU计算速率的191倍。在GARIL月球重力场反演的计算当中,采用GPU并行的计算时间比使用CPU的运算方法时间分别缩短了94.63%,99.51%,可有效缩短传统计算模型所需的时间,求逆误差的量级为 10^{-9} 左右,反演误差的量级为 10^{-17} 左右,误差较小,适用于高阶卫星重力场反演的运算,将其运用到900阶次的月球重力场计算中,在武汉大学高性能计算平台运算,其求逆时间缩短为136 min,计算速率有极大的提高。

本文的算法在矩阵乘法和矩阵求逆方面实现了并行计算,但在设计矩阵计算和矩阵分块部分仍有优化空间。在今后的研究中,将针对这些内容,进一步优化算法性能。此外,本文将会把GPU加速计算方法应用于国内高阶行星重力场的反演研究中,其中包括火星重力场反演等。通过利用GPU的高性能并行计算能力,加速高阶行星重力场模型的计算过程,大幅缩短了传统计算模型所需的时间,从而有助于建立高阶次、高精度的重力场模型。

致谢

感谢武汉大学超级计算中心对本论文的数值计算支持和帮助。

参考文献

- [1] 李斐,郑翔,郝卫峰,等.月球重力场的确定与月面重力测量[J].中国科学基金,2022,36(6):851-858.
LI F, ZHENG C, HAO W F, et al. Determination of lunar gravity field and lunar surface gravity measurement[J]. Science Foundation of China, 2022, 36(6): 851-858.
- [2] ZUBER T M, SMITH D E, WATKINS M M, et al. Gravity field of the Moon from the Gravity Recovery and Interior Laboratory(GRAIL) mission[J]. Science, 2013, 339(6120): 668-671.
- [3] PARK R, KONOPLIV A, YUAN D, et al. A high-resolution spherical harmonic degree 1500 lunar gravity field from the GRAIL mission[C]//Proceedings of American Geophysical Union, Fall Meeting 2015. [S. l.]: AGU, 2015.
- [4] 邹贤才,李建成,汪海洪,等. OpenMP并行计算在卫星重力数据处理中的应用[J].测绘学报,2010,39(6):636-641.
ZOU X C, LI J C, WANG H H, et al. Application of OpenMP parallel computing in satellite gravity data processing[J]. Chinese Journal of Surveying and Mapping, 2010, 39(6): 636-641.
- [5] TAPLEY B D, BETTADPUR S, WATKINS M, et al. The gravity recovery and climate experiment: mission overview and early results[J]. Geophysical Research Letters, 2004, 31(9): L09607.
- [6] PAIL R, PLANK G. Assessment of three numerical solution strategies for gravity field recovery from GOCE satellite gravity gradiometry implemented on a parallel platform[J]. Journal of Geodesy, 2002, 76(8): 462-474.
- [7] KLEES R, KOOP R, VAN GEEMERT R, et al. GOCE gravity field recovery using massive parallel computing[M]. Berlin, Heidelberg: Springer, 2001: 109-116.
- [8] PAIL R, METZLER B, LACKNER B, et al. GOCE gravity field analysis in the framework of HPF: operational software system and simulation results[C]//Proceedings 3rd GOCE User Workshop. Frascati, Italy: ESA, 2006.
- [9] PAIL R, GOIGINGER H, MAYRHOFER R, et al. GOCE gravity field model derived from orbit and gradiometry data applying the time-wise method[C]//Proceedings of the ESA Living Planet Symposium. Bergen, Norway: ESA, 2010.
- [10] BABOULIN M. Solving large dense linear least squares problems on parallel distributed computers. Application to the Earth's gravity field computation[D]. France: Institut National Polytechnique de Toulouse, 2006.
- [11] LEE S, KIM J, JUNG Y, et al. Implementation of the distributed parallel program for geoid heights computation using MPI and OpenMP[J]. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2012, 39: B4.
- [12] COUDER-CASTAÑEDA C, ORTÍZ-ALEMÁN C J, OROZCO-DEL-CASTILLO M, et al. Forward modeling of gravitational fields on hybrid multi-threaded cluster[J]. Geofisica Internacional, 2015, 54(1): 31-48.
- [13] 王正涛. 卫星跟踪卫星测量确定地球重力场的理论与方法[D]. 武汉: 武汉大学, 2005.
WANG Z T. The theory and method of determining the Earth's gravity field by satellite tracking satellite measurements[D]. Wuhan: Wuhan University, 2005.
- [14] 周浩, 罗志才, 钟波. 大规模矩阵的MPI并行求逆算法设计与分析[J]. 大地测量与地球动力学, 2014, 34(5): 120-124.
ZHOU H, LUO Z C, ZHONG B. Design and analysis of MPI parallel

- inversion algorithm for large-scale matrix[J]. *Geodesy and Geodynamics*, 2014, 34(5): 120-124.
- [15] 周浩, 罗志才, 钟波, 等. 利用最小二乘直接法反演卫星重力场模型的MPI并行算法[J]. *测绘学报*. 2015, 44(8): 833-839.
ZHOU H, LUO Z C, ZHONG B, et al. MPI parallel algorithm for inversion of Satellite Gravity Field Model using least square direct method[J]. *Journal of Surveying and Mapping*. 2015, 44(8): 833-839.
- [16] 陈秋杰, 沈云中, 张兴福. MKL和OpenMP多核并行算法解算高阶地球重力场的效率分析[J]. *大地测量与地球动力学*, 2012, 32(5): 118-123.
CHEN Q J, SHEN Y Z, ZHANG X F. Efficiency analysis of MKL and OpenMP multi-core parallel algorithm for solving high-order Earth's gravity field[J]. *Journal of Geodesy and Geodynamics*, 2012, 32(5): 118-123.
- [17] 吴烽昊. 基于泊松小波径向基函数融合多源数据的局部重力场建模方法研究[D]. 武汉: 武汉大学, 2016.
WU Y H. Region gravity field modeling from heterogeneous data sets by using Possion wavelets radial basis functions[D]. Wuhan: Wuhan University, 2016.
- [18] DU S, GAO Y. Inertial aided cycle slip detection and identification for integrated PPP GPS and INS[J]. *Sensors*, 2012, 12(11): 14344-14362.
- [19] GREHNER-BRZEZINSKA A D, DA R, TOTH C. GPS error modeling and OTF ambiguity resolution for high-accuracy GPS/INS integrated system[J]. *Journal of Geodesy*, 1998, 72(11): 626-638.
- [20] DONGARRA J, GATES M, HAIDAR A, et al. Accelerating numerical dense linear algebra calculations with GPUs[M]. New York: Springer, 2014.

作者简介:

周予涵(1999-), 女, 硕士研究生, 主要研究方向: 行星探测器的多普勒信号高精度测量及在重力场反演应用中的研究。

通信地址: 南京市东南大学自动化学院(210096)

电话: 13813927281

E-mail: 573530322@qq.com

陈从颜(1969-), 男, 教授, 博士生导师, 主要研究方向: 深空探测和信号处理。**本文通信作者。**

通信地址: 南京市东南大学自动化学院(210096)

电话: 13815880379

E-mail: chency@seu.edu.cn

Implementation of GPU-Accelerated Computation of Large Matrix in Gravity Field Inversion

ZHOU Yuhan¹, JIAN Nianchuan², CHEN Congyan¹

(1. School of Automatic, Southeast University, Nanjing 210096, China;

2. Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai 200030, China)

Abstract: To address the problem of heavy computational tasks and lengthy processing times in gravity field inversion, a parallel matrix computation method based on multi-GPU integration with CUDA was proposed. This method achieved highly parallel dense computation, significantly reducing the time required for inverting large matrices in planetary gravity field inversion by accelerating matrix multiplication and inversion operations. The computation rate was 191 times faster than that of using CPU. Moreover, it offered high computational accuracy, with inversion precision at the level of 10^{-17} . Applied to the computation of GRAIL lunar gravity field inversion, the proposed method, when computing matrices of truncation orders 50 and 180 respectively reduced the processing time by 94.63% and 99.51% respectively compared to CPU-based methods. Furthermore, the method successfully computed a matrix of 900th order on the High-Performance Computing Platform at Wuhan University. The method employed in this paper can effectively reduce the time needed by traditional computing models, thereby conducive to aiding in the establishment of high-order, high-precision gravity field models.

Keywords: planetary gravity field inversion; GPU computing; parallel computing; CUDA

Highlights:

- The computing method of multi-GPU parallel processing, combining GPU and CUDA, was used to accelerate the multiplication and inversion of large matrices.
- GPU-accelerated computing was applied to the solution of the forward equations in gravity field inversion, solving the spherical harmonic coefficients of the gravity field through matrix inversion.
- Dense matrices randomly generated randomly were computed using three methods: CPU single-GPU and four-GPU showing significant differences in computational speed. The four-GPU computation was 191 times faster than the CPU computation.
- Real data of lunar gravity field were computed using the GPU parallel computing method resulting in a reduction of 94.63% and 99.51% in processing time compared to the CPU method.
- The precision of GPU computation was analyzed with inversion errors at the level of 10^{-9} and inversion accuracy at the level of 10^{-17} .

[责任编辑: 杨晓燕, 英文审校: 宋利辉]