

DOI:10.11918/201910088

卫星轨道递推的 GPU 集成式并行加速方法

孔繁泽, 叶东, 柳子然, 孙兆伟

(哈尔滨工业大学 航天学院, 哈尔滨 150001)

摘要: 为克服传统卫星轨道模型预报方法的速度瓶颈, 为实现卫星在轨自主规划变轨奠定基础, 利用图形处理器(GPU)并行计算方法对多卫星轨道解算进行加速, 构建了轨道预报并行计算模块, 成功实现了卫星轨道预报的大幅加速. 为提高低计算量时解算速度, 提出了集成式 GPU 加速方法, 将简化常规摄动模型(SGP4)解算模型整体代入核函数, 计算机内存仅需与 GPU 进行一次调用及数据交互, 大大缩短调用核函数时间, 较模块化 GPU 加速方法在中低规模计算量时速度有明显提高. 本研究于两种设备上基于统一计算设备架构(CUDA)实现了集成式加速方法并进行了加速试验, 在小型嵌入式开发板 NIVIDA TX2 设备上可实现现在 5 s 内进行 500 颗星一天时间 86 400 步的轨道预报, 笔记本设备上 GPU 加速比也可达到中央处理器(CPU)的 4.6 倍, 且加速后精度损失极低. 实验结果表明: 集成式加速方法适用于中低规模星数(总步数小于 400 万步)的并行解算任务, 模块化加速方法适用于大规模星数(总步数大于 400 万步)的并行解算任务.

关键词: 图形处理器; SGP4/SDP4 模型; 轨道递推; 核函数; 集成式加速

中图分类号: TV19

文献标志码: A

文章编号: 0367-6234(2021)06-0013-08

GPU monolithic parallel acceleration method for satellite orbit prediction with SGP4/SDP4 model

KONG Fanze, YE Dong, LIU Ziran, SUN Zhaowei

(School of Astronautics, Harbin Institute of Technology, Harbin 150001, China)

Abstract: To overcome the speed limit of the traditional satellite orbit prediction method and lay foundation for independent orbital transfer planning of on-orbit satellites, the graphics processing unit (GPU) parallel computing method was utilized to accelerate the multi-satellite orbit calculation, and the parallel prediction module of orbit prediction was constructed, which realized the acceleration of satellite orbit prediction. In order to improve the calculation speed when the calculation amount is low, a monolithic GPU acceleration method was proposed, which substituted the simplified general perturbation version 4 (SGP4) calculation model into the kernel function. The computer memory only needed to interact with the GPU once, which greatly shortened the data transmission time between the memory and the GPU. Compared with the modular GPU acceleration method, the speed for medium or low scale calculations was increased greatly. The proposed monolithic acceleration method was implemented on two devices based on compute unified device architecture (CUDA) library. On NIVIDA TX2, a small embedded development board, it could realize the orbit prediction of 500 satellites for one day in 5 s (86 400 steps for each satellite), while the GPU acceleration ratio on the laptop was 4.6 times more than that of the central processing unit (CPU), and the precision loss after the acceleration was low. The experiment showed that the monolithic acceleration method was suitable for the parallel calculation of low and medium scale calculations (the number of steps is less than four million), and the modular acceleration method was suitable for the parallel calculation of large scale calculations (the number of steps is more than four million).

Keywords: graphics processing unit (GPU); SGP4/SDP4 model; orbit prediction; kernel function; monolithic acceleration

据美国空间监测网站给出的数据, 截至 2013 年, 地球轨道上尺寸大于 10 cm 的可编目空间物体

数量已超过 16 000 多个. 针对日益增多的空间物体, 在轨卫星在进行变轨机动时, 需对大量空间物体进行高精度的轨道预报计算, 以预防在轨航天器与其他空间天体的碰撞. 由于卫星高精度的轨道模型具有较高的计算复杂度, 而目前计算机上常用的 CPU (Central Processing Unit) 处理器仅能进行串行计算, 进行轨道预报需消耗大量的时间, 难以满足军事上及民用上卫星的快速响应需求, 因此有必要对

收稿日期: 2019-10-14

基金项目: 国家自然科学基金(62073102, 91638301); 国防基础科研计划(JCKY2017603B006)

作者简介: 孔繁泽(1998—), 男, 博士研究生;

叶东(1985—), 男, 副教授, 博士生导师;

孙兆伟(1963—), 男, 教授, 博士生导师

通信作者: 叶东, yed@hit.edu.cn

轨道模型计算加速方法展开研究. 如表 1 所示为各类星载计算机的 CPU 型号与性能参数.

表 1 On board computer (OBC) 及 CPU 参数

Tab. 1 On board computer (OBC) and CPU parameters

OBC	CPU	CPU 参数
Galileo CDS	RCACosmac 1802	~1.6 MHz, 8 bit ~25 MIPS@
Mars Pathfinder AIM	RAD6000	20 MHz, 32 bit RISC ~300 MIPS@
Deep Impact	RadLite750	133 MHz
Curiosity	BAE RAD 750	133 MHz,
中巴地球资源卫星	80C86&80C31	
实践五号小卫星	80386EX	
天宫一号	SPARC 兼容格式的 ERC32 处理器	10 MHz, 32 bit

对此,国内外都进行了相关的研究. 计天阳^[1]采用模块化加速方法及英伟达公司提供的 CUDA 库, 配套 GPU (Graphics Processing Unit) 对 SGP4/SDP4 轨道模型进行了加速试验, 填补了国内在轨道模型 GPU 并行加速领域研究的空白. Liu 等^[2]利用 GPU 并行计算加速方法进行大量空间碎片的轨道预报, 通过 PMC 简化模型 (The global point mascon model), 可进行 15 000 颗空间碎片轨道的并行计算, 其精度较 SGP4/SDP4 低, 运算速度快, 加速比可达 10~15 倍. Lin 等^[3]基于 CUDA 的并行架构, 采用块分解算法, 对空间碎片进行分批次处理, 相邻批次的计算和数据传输操作互相交叠. 基于此算法, 150 个飞行器 3 d 内的碰撞预报不超过 3 h, 在大型高性能 GPU 显卡上计算速度可达 CPU 加速的 30 倍. 郭松等^[4]在 FPGA 平台上设计并实现了基于 VLIW 的 SGP4/SDP4 轨道预报加速器, 在单片 FGPA 上用 8 个处理单元组成计算阵列, 采用指令级并行和任务级并行相结合的方式, 将轨道预报速度提升到了 CPU 的 8 倍以上. 冯森^[5]利用 CUDA 并行架构和 GPU 硬件作为 CPU 的协处理器, 实现了 SGP4 轨道预报模型计算时间的大幅缩小, 实验结果证明其加速比达到 15 倍. 综上所述, 卫星轨道预报并行递推面临解算速度以及解算精度这两个相互冲突的问题, 也即如何在保证解算速度的同时尽可能的提高轨道预测精度. 目前精度较高的解算方法都为外推法, 如四阶龙格库塔法以及 HPOP 高精度外推模型^[6]. 但由于外推法计算量极大, 且其外推的特点在保证精度的同时却无法进行并行计算, 外推法难以应用于快速解算之中. 因此, 本文基于 SGP4/SDP4 模型, 考虑摄动模型的高精度性质及解析特性, 即同一卫星在不同时间点的轨道模型解可

独立解得, 符合并行计算的特点, 提出 GPU 并行计算方法, 使不同卫星的星历、同一颗星不同时间点的星历在不同的线程中同步计算, 以达到数倍于 CPU 处理器速度的加速效果.

Lin 等^[3]和冯森^[5]都在高性能工作站上进行并行解算实验, 加速性能达到较高程度. 本文拟在小型板卡 NVIDIA TX2 上完成并行轨道模型加速解算的部署与优化, 大大有利于推进基于 GPU 的卫星轨道并行计算的学术研究与工程应用.

1 轨道预报模型

轨道预报是指已知卫星某一时刻位置、速度状态, 根据卫星的动力学微分方程来预测卫星未来一段时间内的位置和速度. 为了表示卫星的位置、速度信息, 北美防空司令部 (The north amerian aerospace defense command, NORAD) 开发了双行元 (two-line element, TLE)^[7]. 其综合考虑了大气阻力、日月引力的长期和周期摄动、地球扁率等影响. TLE 作为平均根数, 使用特定的方法消除了周期扰动项, 解算时必须使用同样的方法重构, 因此 TLE 只适用于特定的轨道预报模型.

针对不同轨道高度的空间目标, NORAD 开发了与 TLE 相应的近地或深空模型, 目前定期更新的 TLE 针对的模型是 SGP4/SDP4 模型. SGP4 模型 (Simplified general perturbation version 4) 能够对轨道高度较低或绕地飞行周期小于 225 min 的航天器进行轨道预报, SDP4 模型 (Simplified deep space perturbation version 4) 能够对轨道高度较高或绕地飞行周期大于 225 min 的航天器进行轨道预报. SGP4/SDP4 模型预报精度较高, 模型为解析模型, 计算复杂度较递推模型低, 综合性能高, 因此其在卫星轨道预报领域被人们广泛使用^[8-9].

SGP4 预报模型解算采用解析法, 需要将输入数据转换为初始轨道位置. 由于 TLE 中采用平均运动, 解算时需要将其重新转换为半长轴, 转换方法与正常轨道根数转换方法有差异.

首先, 求解平均运动对应半长轴, 即

$$a_1 = \left(\frac{\mu}{n_0} \right)^{\frac{2}{3}}. \quad (1)$$

式中: μ 为地球引力常数; n_0 为平均运动.

J_2 摄动下的轨道倾角和偏心率分别为:

$$\delta_1 = \frac{3}{2} \frac{k_2}{a_1^2} \frac{(3 \cos^2 i_0 - 1)}{(1 - e_0^2)^{3/2}}, \quad (2)$$

$$a_0 = a_1 \left(1 - \frac{1}{3} \delta_1 - \delta_1^2 - \frac{134}{81} \delta_1^3 \right). \quad (3)$$

式中: $k_2 = 0.5 J_2 a_E^2$, J_2 为地球二阶引力带谐波项; i_0 、

e_0 分别为平均轨道倾角和平均偏心率.

由此,对半长轴进行修正:

$$\delta_0 = \frac{3}{2} \frac{k_2 (3 \cos^2 i_0 - 1)}{a_0^2 (1 - e_0^2)^{3/2}}, \quad (4)$$

$$n'_0 = \frac{n_0}{1 + \delta_0}, \quad (5)$$

$$a'_0 = \frac{a_0}{1 - \delta_0}. \quad (6)$$

式中: a'_0 、 n'_0 分别为改进的半长轴和周期参数^[10-11].

对于低轨道卫星,即地心距小于 98 km,SGP4 功率函数中常变量 s 的表达式为

$$s = \frac{20}{R_{\oplus}} + a_E. \quad (7)$$

计算出如下参数:

$$\theta = \cos i_0, \quad (8)$$

$$\zeta = \frac{1}{a_0'' - s}, \quad (9)$$

$$\eta = a_0'' e_0 \zeta. \quad (10)$$

利用初始参数,进行 SGP4 轨道预报模型的递推求解. SGP4 模型包含大气阻力摄动、非球形轨道摄动、长周期摄动项和短周期摄动项的计算^[12]. 对于低轨道目标,主要摄动项为大气阻力摄动和非球形摄动,表达式如下:

$$M_{DF} = M_0 + \left[1 + \frac{3k_2(-1 + 3\theta^2)}{2a_0''\beta_0^3} + \frac{3k_2^2(13 - 78\theta^2 + 137\theta^4)}{16a_0''\beta_0^7} \right] n_0''(t - t_0), \quad (11)$$

$$\omega_{DF} = \omega_0 + \left[-\frac{3k_2(1 - 5\theta^2)}{2a_0''\beta_0^4} + \frac{3k_2^2(7 - 114\theta^2 + 395\theta^4)}{16a_0''\beta_0^8} + \frac{5k_4(3 - 36\theta^2 + 49\theta^4)}{4a_0''\beta_0^8} \right] n_0''(t - t_0), \quad (12)$$

$$\Omega_{DF} = \Omega_0 + \left[-\frac{3k_2\theta}{a_0''\beta_0^4} + \frac{3k_2^2(4\theta - 19\theta^3)}{2a_0''\beta_0^8} + \frac{5k_4\theta(3 - 7\theta^2)}{2a_0''\beta_0^8} \right] n_0''(t - t_0). \quad (13)$$

式中: M_0 为某段时间内的平均平近角; Ω_0 为某段时间内的平均升交点赤经.

大气阻力摄动和非球形摄动造成轨道根数发生变化,有:

$$\delta\omega = B^* C_3 (\cos \omega_0) (t - t_0), \quad (14)$$

$$\delta M = -\frac{2}{3} (q_0 - s)^4 B^* \xi^4 \frac{a_E}{e_0 \eta} \left[(1 + \eta \cos M_{DF})^3 - (1 + \eta \cos M_0)^3 \right], \quad (15)$$

$$M_p = M_{DF} + \delta\omega + \delta M, \quad (16)$$

$$\omega = \omega_{DF} - \delta\omega - \delta M, \quad (17)$$

$$\Omega = \Omega_{DF} - \frac{21}{2} \frac{n_0'' k_2 \theta}{a_0'' \beta^2} C_1 (t - t_0)^2, \quad (18)$$

$$e = e_0 - B^* C_4 (t - t_0) - B^* C_5 (\sin M_p - \sin M_0), \quad (19)$$

$$a = a_0'' \left[1 - C_1 (t - t_0) - D_2 (t - t_0)^2 - D_3 (t - t_0)^3 - D_4 (t - t_0)^4 \right]^2, \quad (20)$$

$$IL = M_p + \omega + \Omega + n_0'' \left[\frac{3}{2} C_1 (t - t_0)^2 + (D_2 + 2C_1^2) (t - t_0)^3 + \frac{1}{4} (3D_3 + 12C_1 D_2 + 10C_1^2) (t - t_0)^4 + \frac{1}{5} (3D_4 + 12C_1 D_3 + 6D_2^2 + 30C_1^2 D_2 + 15C_1^{25}) (t - t_0)^5 \right], \quad (21)$$

$$\beta = \sqrt{(1 - e^2)}, \quad (22)$$

$$n = \frac{k_e}{a^{3/2}}, \quad (23)$$

式中, B^* 为 SGP4 典型阻力系数. 在近心点距小于 220 km 时, a 和 IL 的表达式中 C_1 后面的项可舍去,含有 C_5 、 $\delta\omega$ 、 δM 的项可忽略^[13-15].

2 基于 GPU 的并行计算流程设计

2.1 GPU 并行计算

GPU 是一种在个人电脑、工作站、游戏机和一些移动设备(如平板电脑、智能手机等)上专门进行图像运算工作的微处理器. 如图 1 所示, GPU 与 CPU 组成相同,都由运算基本单元、控制器、寄存器组成. CPU 内大部分空间被控制器和寄存器占据,其可利用的临时缓存较大,擅长逻辑控制和串行计算,多用于处理包含不同数据类型的复杂操作. 相比于 CPU 处理器, GPU 拥有更多的 ALU(算术逻辑单元,是实现多组算术运算和逻辑运算的组合逻辑电路)用于数据处理,而非数据高速缓存和流控制,此类结构适合于对密集型数据进行并行处理,例如本文中所处理的多卫星轨道数据.

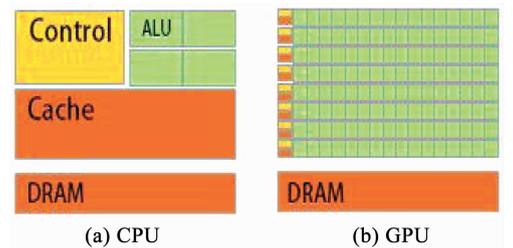


图 1 CPU 与 GPU 结构对比示意

Fig. 1 Comparison of structures of CPU and GPU

因此, GPU 可同时调用大量相同类型的数据,并将它们分配入不同线程中进行相同类型的计算操

作,其耗费的时间和单独计算一个数据的时间一致,这就是 GPU 并行计算的概念,即“以空间换时间”的处理方式.

2.2 CUDA 计算平台

CUDA (Compute unified device architecture)是由 NVIDIA 公司推出的一种通用并行计算框架,仅适用于 NVIDIA 公司的产品(以显卡产品为主).其架构包含软硬件体系,硬件为 CUDA 支持的 GPU 处理器,软件则包括显卡驱动程序、nvcc 编译器、调试模块等组成的开发环境,支持 C 语言及 C++ 语言编写,通用性较强,易于上手.可将 CUDA 框架简单理解为一个可通过特定编程模型实现 GPU 并行加速的 C 语言函数库.本文中测试程序均于 Visual Studio 2013 与 CUDA 9.0 开发平台完成.

CUDA 框架结构中,存在两个必要的程序组成部分:主机程序以及内核函数 kernel.主机程序运行在 CPU 处理器中,实现了 CPU 部分的计算、CPU 与内存间数据传输、CPU 与 GPU 间内存交互、设置 GPU 并行计算参数的功能.内核函数仅运行于 GPU 各线程中,为实际并行计算的运算部分,各 GPU 线程内的相同内核函数带入不同参数同步进行计算.如图 2 所示,CUDA 框架内 GPU 线程结构分为 3 个层次:网格(Grid)、线程块(Block)和线程(Thread).每当一个内核函数被发送到 GPU 时,即生成一个自定义网格,网格内存在若干自定义维度的线程块,每个线程块内存在同等数量的线程.每个线程都分配有相应的寄存器和本地内存,在其所在的线程块内有一共享内存空间可供该线程块内所有线程读取,同时对于整个网格有可供所有线程访问读取的全局常量及内存,GPU 并行计算可根据此原理进行参数设置以达到最优加速效果.

2.3 集成式加速流程设计

本文集成式框架的输入项为一 TXT 格式的双行元文件,其包含任意数目卫星的某一时刻双行元数据,表示卫星初始的轨道数据,多颗卫星的双行元数据被统一读取并存入内存.

卫星初始数据读入后,从内存中依次调出每一颗星的双行元数据,并根据双行元数据进行轨道初始参数(轨道六根数)的计算并存入内存,由于此过程单颗卫星参数计算时间小于 SGP4 模型解算时间的 1/10,因此本文中并未对此计算流程进行并行加速.此过程为在 CPU 运算单元中按顺序进行每一颗星的轨道参数初始化操作.

如图 3 所示,完成所有双行元数据的轨道初始参数计算后,程序进入第 1 次数据传输阶段,即 CPU 至 GPU 的数据传输.根据双行元数据的个数开辟相

应大小的 GPU 内存空间,每个双行元数据对应的 GPU 内存空间内分别储存以下 4 种数据:起始时间、结束时间、解算步长以及卫星星历.其中卫星星历在程序中是以一个结构体变量形式存储的,它包含了每一颗星位置、速度与轨道参数信息. CPU 到 GPU 的数据传输过程是将 CPU 内存中每颗星起始时间、结束时间、解算步长及起始时刻表示卫星状态的结构体变量数据打包传输到 GPU 内存空间中,等待 GPU 调用.

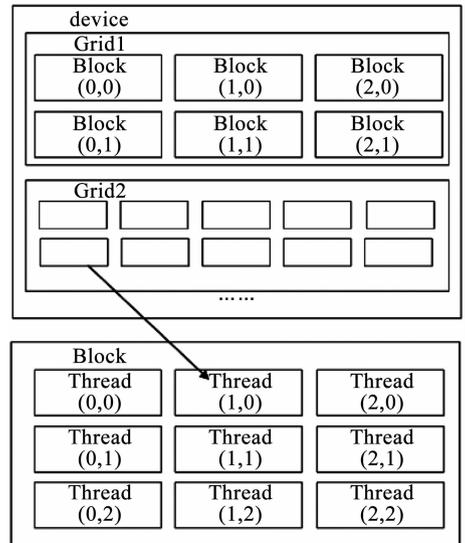


图 2 CUDA 线程结构示意图

Fig. 2 Schematic of CUDA thread structure

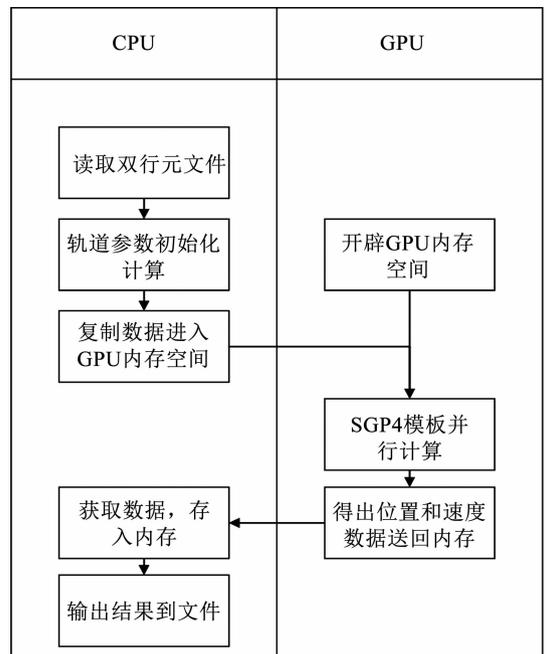


图 3 程序流程图

Fig. 3 Program flowchart

第 1 次数据传输完成后, GPU 将每一颗星对应的结构体变量带入多个独立线程中的核函数进行并

行计算. GPU 内每一个运算单元为一独立线程,每一独立线程中计算一颗星在某一时间点的星历,则每一颗星的结构体变量需带入计算的线程数量为

$$n = \frac{t_s - t_0}{t_p} \quad (24)$$

式中: n 为单颗星所需线程数量; t_s 为结束时间; t_0 为起始时间; t_p 为步长

在并行计算过程中,多颗星的计算步骤不是顺序进行的,而是同步进行的,即多颗星在独立线程中的解算是同时进行的,以此达到加速效果. 所需的并行线程数量为

$$N = n_s \times n. \quad (25)$$

式中: N 为总线程数量; n_s 为卫星数; n 为单星线程数量.

以计算一颗卫星 24 h 的星历为例,以 1 s 为解算步长,所需线程数为 86 500 个,计算 200 颗星即为 17 300 000 个线程,所需的运算单元数超过 CPU 串行运算 2~3 个数量级,符合本文“以空间换时间”的理念. 但因此,使用本方法进行并行解算的星数会受 GPU 内存空间大小的限制. 例如,在 Jetson TX2 设备上可同步解算 500 颗星,但在 PC 电脑端本方法只可同步递推 50 颗星的轨道.

GPU 并行计算完成后得到卫星星历数据,进行第 2 次数据传输(GPU 至 CPU),将星历数据传输回 CPU 内存,再输出到文件中结束解算.

3 实验加速效果与分析

为了验证所提集成式加速方法的有效性和适应性,即本文提出方法是否达到加速效果,若达到加速效果,在何种计算条件及硬件设备在能达到最好的使用效果,本文分别在两种仿真环境下进行 GPU 并行加速运行试验:笔记本电脑 PC 端环境以及 JETSON TX2 开发板环境.

在笔记本电脑 PC 端进行的实验对比了 CPU 解算速度与 GPU 集成式加速方法解算速度,验证了集成式加速方法的有效性. 又将两种环境下的集成式加速结果与模块化加速结果进行比较,得出两种加速方法各自的适用场景. 最后通过对比两种环境下集成式加速方法的加速效果,讨论了 GPU 并行加速方法的设备适用性,提供了卫星在轨解算平台的硬件选择建议.

3.1 集成式加速方法有效性分析

本文在笔记本电脑 PC 端环境下首先进行了 CPU 处理器上的 SGP4/SDP4 轨道模型解算实验,又于同一台设备上的 GPU 处理器进行集成式方法的并行加速实验,两次实验结果如表 2 和图 4 所示.

仿真过程中每颗星解算步长为 1 s,解算时长为 24 h,则每颗星计算 86 400 步,输出 86 400 组解算结果.

试验验证环境 1(笔记本电脑):

操作系统:Windows 10 中文版.

CPU:英特尔 core i7-6700HQ, 2.6 Hz 主频, 8 GB 内存.

GPU:NVIDIA GeForce GTX 950 M, 包含 640 个 CUDA 单元,4 096 MB 显存.

表 2 仿真环境 1 下仿真结果

Tab. 2 Results of the first simulation environment

星数	CPU 实现时间/ms	GPU 实现时间/ms	加速比
1	214	631	0.197
10	2 643	809	3.270
30	7 568	1 701	4.450
50	12 732	2 773	4.590

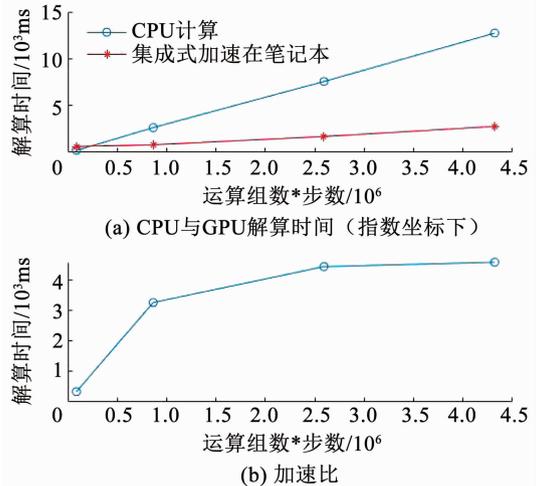


图 4 CPU 和 GPU 解算速度对比

Fig. 4 Comparison of calculation speed in CPU and GPU

集成式方法需较大的数据储存空间进行运算,鉴于 GPU 显卡显存及运算单元数量限制,笔记本电脑环境仅能并行解算 50 颗星星历数据,而 TX2 开发板能并行解算 500 颗星的数据. 显存空间的大小及 GPU 运行单元数量决定并行计算所能计算的星数上限,并影响加速效果.

由实验结果如表 2 及图 4 可知,在星数较少时,总运算步数也较少,CPU 解算时间要低于 GPU 并行解算时间. 这是由于 CPU 处理器在进行单次解算时解算速度为 GPU 处理器速度的数倍;且进行 GPU 加速时,CPU 调用 GPU 核函数进入线程的时间为主要消耗时间,真正的 GPU 线程解算时间都仅为解算一个星历时间点的解算时间. 两者的共同因素导致在总运算步数较低时,采用 CPU 处理器运算效率较高.

当星数增加时,数据量增大,CPU 调用核函数的时间随之增加,但调用函数耗费时间增长率较无并行计算的 CPU 解算时间增长率低,此时 GPU 并行解算开始呈现速度优势,加速比(即 GPU 解算速度除以 CPU 解算速度)最终逐渐稳定于 4.6,达到较高加速比。

3.2 集成式加速方法适应性分析

上文分析了集成式方法的加速有效性,以下对集成式方法和模块化方法进行对比,提出两种方法各自的适用范围,以及加速方法的设备适用性。

本文于 NVIDIA Jetson TX2 小型开发板上进行了集成式加速方法的 SGP4/SDP4 轨道递推模型并行计算实验,将其结果与计天阳^[1]中模块化加速方法实验结果进行对比,得出两者方法的适用计算范围。又通过集成式方法在试验环境 1、2 上加速效果的对比,分析提出轨道模型 GPU 并行加速的设备适用性。试验结果对比见图 5 和表 3。

试验验证环境 2(NVIDIA TX2 小型开发板):

操作系统:ARM-LINUX Ubuntu 16.04

GPU:NVIDIA Pascal,256 CUDA 核心

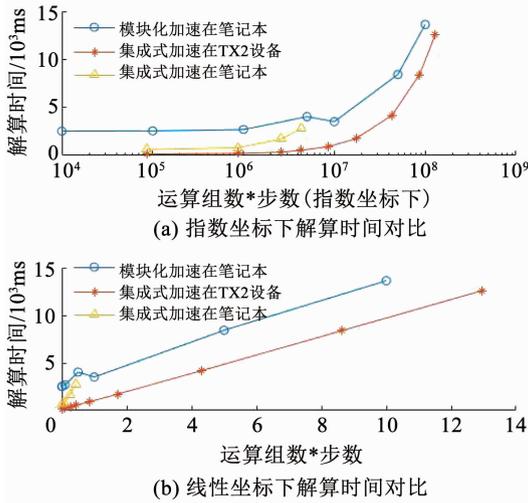


图 5 不同方法 GPU 解算速度对比

Fig. 5 Comparison of calculation speed in GPU with different methods

表 3 不同 GPU 解算方法运算时间表

Tab. 3 Calculation time of different GPU computing methods

序号	集成式加速方法		模块化加速方法	
	总步数 (步数 * 星数)	解算时 间/ms	总步数 (步数 * 星数)	解算时 间/ms
1	86 400	154.37	10 000	2 529
2	864 000	220.67	100 000	2 552
3	2 592 000	348.97	1 000 000	2 697
4	4 320 000	547.44	5 000 000	4 035
5	8 640 000	926.12	10 000 000	3 516
6	17 280 000	1 737.47	50 000 000	8 422
7	43 200 000	4 195.39	100 000 000	13 625

在进行 GPU 并行解算过程中,较 GPU 的解算速度,CPU 调用核函数的速度更为缓慢,如图 6 所示为一颗星 86 400 步数下使用 Visual Profiler 分析程序得出的各进程占用时间比图,可见建立加速进程的过程较核函数运行时间长。为提高解算速度,本文提出减少调用 GPU 并行计算核函数次数以达到更高速度的集成式加速方法。

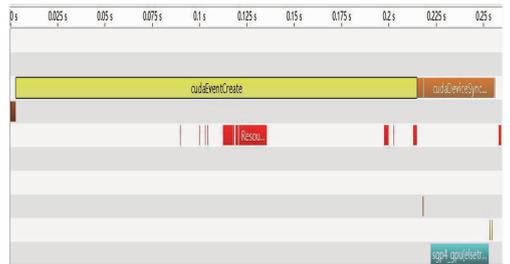


图 6 GPU 加速后进程时间线(左边黄色为调用创建进程的时间)

Fig. 6 Process timeline after GPU acceleration (the left yellow block is the process of calling kernel function)

集成式解算方法将 SGP4 解算模型(包括初始常量定义)作为整体加入核函数计算过程中。计算机内存仅需与 GPU 内存进行一次交互数据传输和一次核函数调用及显存空间分配,相比模块化方法省去多次参数交互和函数调用的时间。但此加速方法单次计算操作复杂,占用显存空间大,当并行计算的星数达到一定数量时,会由于缓存不足出现溢出错误,造成解算失败。

模块化解算方法则将解算过程分为 4 个模块:重力摄动常数初始化模块、双行元与定轨参数转换模块、SGP4 轨道模型初始化模块、SGP4 轨道预报模块。除重力摄动常数初始化模块外,其余 3 个模块在 GPU 中进行并行计算。分模块进行解算对显存使用需求小,可以实现大规模卫星轨道并行计算。但此方法在 CPU 与 GPU 间参数交互的过程中耗费大量时间,加速效果不够理想。

由图 5 及表 3,综合分析集成式和模块化两种并行计算方法的试验数据可发现,在运算数较小(本文试验中运算组数 * 步数 < 400 万步)时,宜采用集成式解算方法,在运算数较大(本文试验中运算组数 * 步数 > 400 万步)时,宜采用模块化解算方法。两种加速方法对比及适用范围可见表 4。

在得出加速方法最佳适用范围后,还需讨论加速方法所适用的计算设备,为搭建卫星在轨解算平台提供参考。

由实验结果图 5 和表 4 可知,低计算量时相同计算量下集成式加速方法解算时间低于模块化加速方法,集成式加速方法在 TX2 设备上实现解算速度

最高. TX2 开发板 GPU 计算性能优越, CPU 性能较普通笔记本计算机低, 其搭载的专用 ARM-LINUX 操作系统内核使得其上内存与 GPU 内存数据传输速率较普通计算高, 因此在 GPU 并行运算时间差距不大的情况下, TX2 开发板能够实现最高的解算速度.

表 4 两种解算方法对比及适用范围

Tab. 4 Comparison and range of application of two calculating methods

解算方法	集成式解算方法	模块化解算方法
优点	并行解算速度快	显存需求小
缺点	显存需求大, 无法进行超大规模并行解算	并行解算速度低
适用范围	本文试验中运算组数 * 步数 < 400 万步	本文试验中运算组数 * 步数 > 400 万步

由图 5 可见, 两种解算方法解算时间与计算量都呈现出线性关系, 解算时间随计算量线性增长. 集成式加速在笔记本上解算时间随时间增长速率最高, 经分析认为是 WINDOWS 操作系统下 Visual Studio 开发环境下, CUDA 程序中由 CPU 内存至 GPU 内存的数据传输过程需经由数次内存转移到显卡设备, 导致解算速度较低, 低于 LINUX 系统环境效率. 此现象也说明, 在进行在轨卫星轨道递推解算时, 平台应尽量选用搭载结构简约的 LINUX 系统的计算机, 以降低数据传输带来的时间耗费.

4 GPU 运算数据精度分析

获取在试验环境 1 下 CPU 解算和并行解算得出的两份星历递推数据, 递推时间为 24 h, 即 86 400 步. 将数据导入 MATLAB 中对比, 进行二者卫星星历 xyz 轴上位置误差及直线距离误差计算, 因本文实验中笔记本计算机上显卡设备与 TX2 设备的 GPU 双精度运算位数相同, 试验环境 1 与试验环境 2 并行计算解算得到的星历数据一致. 用于仿真的双行元数据见表 5, 精度仿真结果如图 8 所示.

表 5 仿真实验中使用的双行元数据

Tab. 5 TLE used in simulation experiment

行数	数据	行数	数据
1	35683U	2	35683
1	09041C	2	98.022 1
1	12289.23158813	2	185.368 2
1	0.00000484	2	0001499
1	00000-0	2	100.529 5
1	89219-4	2	259.608 8
1	0	2	14.698 195 87
1	5863	2	172 294

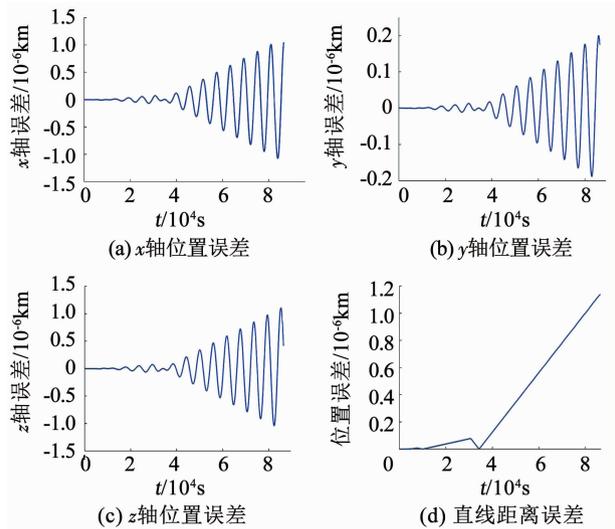


图 7 GPU 运算位置与 CPU 运算对比

Fig. 7 Comparison of position calculated by CPU and GPU

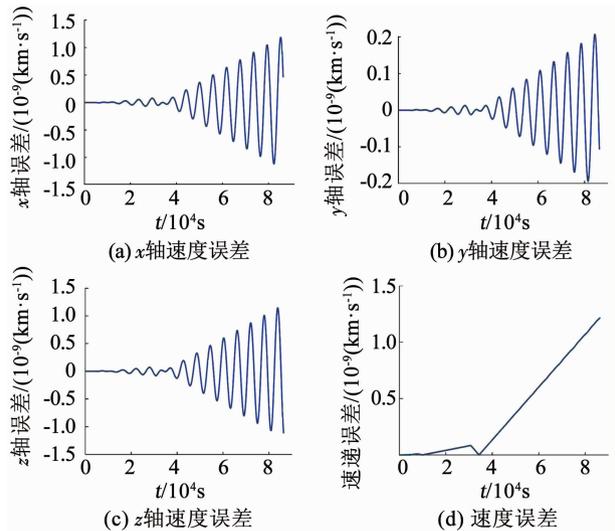


图 8 GPU 计算速度与 CPU 计算对比

Fig. 8 Comparison of velocity calculated by CPU and GPU

与 CPU 运算相比, GPU 运算的位置误差在 10^{-6} km 量级, 速度误差在 10^{-9} km 量级, 误差较小, 精度较高, 但误差并不收敛, 随着运行时间的增长逐渐发散. 误差发散的主要原因在于 CPU 和 GPU 中的变量精度不同. 本文对 CPU 变量及 GPU 核函数中变量都以双精度形式保存. CPU 中双精度运算位数为 80 位, GPU 中双精度运算位数为 64 位, 随着解算过程的进行, 代入解析解的数值增大, GPU 双精度计算产生的误差随之增大, 结果误差也越大, 但总体在可接受范围.

5 结论

1) 该方法将完整的 SGP4/SDP4 轨道预报模型解算过程写入 GPU 核函数中进行运算. 在中小规模计算量下的加速效果优于模块化加速方法.

2)通过实验仿真对比,集成式加速方法较模块化加速方法在中小规模计算量下加速比更高,且精度损失小.

3)由于集成式方法占用较大显存容量,因此难以用于超大规模计算量的加速解算,此外,由于GPU内部计算条件限制,导致误差会随着运行时间增长而逐渐发散.针对这两个问题,在后续工作中将采用新的算法尝试进行优化.

参考文献

- [1] 计天阳. 基于 GPU 加速的低轨卫星星座覆盖性能计算与分析 [D]. 哈尔滨: 哈尔滨工业大学, 2018
JI Tianyang. Calculation and analysis of coverage performance of LEO satellite constellations based on GPU acceleration [D]. Harbin: Harbin Institute of Technology, 2018. DOI: 10.7666/d.D01587155
- [2] LIU J, WANG W, GAO Y, et al. Paralleled geopotential computing methods based on GPU [C]//Proceedings of the China Satellite Navigation Conference. Singapore: Springer, 2016: 87. DOI: 10.1007/978-981-10-0940-2_8.
- [3] LIN Mingpei, XU Ming, FU Xiaoyu. A parallel algorithm for the initial screening of space debris collisions prediction using the SGP4/SDP4 models and GPU acceleration [J]. Advances in Space Research, 2017, 59(9): 2398. DOI: 10.1016/j.asr.2017.02.023.
- [4] 郭松, 雷元武, 窦勇. 基于 VLIW 的 SGP4/SDP4 轨道预测加速器的设计与实现 [C]//第十四届计算机工程与工艺会议 (NCCET'10) 论文集. 扬州: 中国计算机学会, 2010: 147
GUO Song, LEI Yuanwu, DOU Yong. Design and realization of a SGP4/SDP4 orbit prediction accelerator based on VLIW [C]//Proceedings of the 14th Computer Engineering and Technology Conference (NCCET 10). Yangzhou: China Computer Federation, 2010: 147
- [5] 冯森. 基于 CUDA 异构并行加速的两行根数算法的研究 [J]. 无线互联科技, 2016(9): 96
FENG Sen. Research on heterogeneous parallel CUDA accelerated two lines of the number of algorithm [J]. Wireless Internet Technology, 2016(9): 96. DOI: 10.3969/j.issn.1672-6944.2016.09.044.
- [6] 车通宇, 杨力, 张传定, 等. 利用 HPOP 轨道仿真模型实现接收机自主历书外推 [J]. 全球定位系统, 2015, 40(4): 18
CHE Tongyu, YANG Li, ZHANG Chuanding, et al. Executing receiver autonomous ephemeris extrapolation by HPOP model [J]. GNSS World of China, 2015, 40(4): 18. DOI: 10.13442/j.gnss.1008-9268.2015.04.004

- [7] HOOTS F R, ROEHRICH R L. Space track report No. 1-models for propagation of NORAD element sets [R]. Peterson: NORAD, 1980
- [8] 杨嘉墀, 范秦鸿, 张云彤, 等. 航天器轨道动力学与控制 [M]. 北京: 中国宇航出版社, 2009: 15
YANG Jiachi, FAN Qinhong, ZHANG Yuntong, et al. Orbit dynamics and control of spacecraft [M]. Beijing: China Astronautic Publishing House, 2009: 15
- [9] 刘林. 航天器轨道理论 [M]. 北京: 国防工业出版社, 2000: 5
LIU Lin. Orbit theory of spacecraft [M]. Beijing: National Defense Industry Press, 2000: 5
- [10] 蒋方华, 李俊峰, 宝音贺西. 高精度卫星轨道摄动模型 [C]. 中国第十三届空间及运动体控制技术学术年会论文集. 宜昌: 中国自动化学会, 2008
JIANG Fanghua, LI Junfeng, BAOYIN Hexi. High precision orbit perturbation model [C]//Proceedings of the China's 13th Annual Conference on Space and Sports Control Technology. Yichang: Chinese Association of Automation, 2008
- [11] 刘一帆. 基于 SGP4 模型的低轨道航天器轨道预报方法研究 [D]. 哈尔滨: 哈尔滨工业大学, 2009
LIU Yifan. Research on low earth orbit spacecraft orbit prediction strategy based on SGP4 model [D]. Harbin: Harbin Institute of Technology, 2009. DOI: 10.7666/d.D270797
- [12] 韦栋, 赵长印. SGP4/SDP4 模型精度分析 [J]. 天文学报, 2009, 50(3): 332
WEI Dong, ZHAO Changyin. Analysis on the accuracy of the SGP4/SDP4 model [J]. Acta Astronomica Sinica, 2009, 50(3): 332. DOI: 10.3321/j.issn:0001-5245.2009.03.010
- [13] 李梦奇. 基于大气参数修正的低轨航天器 SGP4/SDP4 模型精度分析轨道预报方法研究 [D]. 哈尔滨: 哈尔滨工业大学, 2017
LI Mengqi. Research on low earth orbit spacecraft orbit prediction strategy based on atmospheric coefficient modification [D]. Harbin: Harbin Institute of Technology, 2017. DOI: 10.7666/d.D01333050
- [14] 王明明, 罗建军, 马卫华. IAU1976、1980 及 2000A 岁差章动模型比较 [J]. 中国空间科学技术, 2009, 29(5): 42
WANG Mingming, LUO Jianjun, MA Weihua. Comparison of IAU1976, 1980 and IAU2000A precession-nutation model [J]. Chinese Space Science and Technology, 2009, 29(5): 42. DOI: 10.3321/j.issn:1000-758X.2009.05.007
- [15] 郭俊义. 同时顾及章动和极移的地球自转方程 [J]. 武汉大学学报(信息科学版), 2000, 25(5): 391
GUO Junyi. Earth rotation equations containing both nutation and polar motion [J]. Geomatics and Information Science of Wuhan University, 2000, 25(5): 391. DOI: 10.3969/j.issn.1671-8860.2000.05.003

(编辑 张红)