

面向SW26010间断有限元算法的多级并行计算

王晓忠, 张祖雨

引用本文

王晓忠, 张祖雨. 面向SW26010间断有限元算法的多级并行计算[J]. 计算机科学, 2024, 51(11A): 240700055-5.

WANG Xiaozhong, ZHANG Zuyu. Multi Level Parallel Computing for SW26010 Discontinuous Galerkin Finite Element Algorithm [J]. Computer Science, 2024, 51(11A): 240700055-5.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[基于FPGA并行实现SVM训练的可重构计算系统](#)

Reconfigurable Computing System for Parallel Implementation of SVM Training Based on FPGA
计算机科学, 2024, 51(11A): 231100120-7. <https://doi.org/10.11896/jsjcx.231100120>

[基于双目估计的动态场景三维感知技术研究与实现](#)

Research and Implementation of Dynamic Scene 3D Perception Technology Based on Binocular Estimation
计算机科学, 2024, 51(11A): 240300045-8. <https://doi.org/10.11896/jsjcx.240300045>

[基于OPENMP的再入飞行器轨迹多重打靶法并行计算](#)

Parallel Computing of Reentry Vehicle Trajectory by Multiple Shooting Method Based on OPENMP
计算机科学, 2024, 51(11A): 231000019-6. <https://doi.org/10.11896/jsjcx.231000019>

[基于鲲鹏处理器的LU并行分解优化算法](#)

LU Parallel Decomposition Optimization Algorithm Based on Kunpeng Processor
计算机科学, 2024, 51(9): 51-58. <https://doi.org/10.11896/jsjcx.230900079>

[基于自适应修正模型的飞行器外流场三维计算方法](#)

Adaptive Modification Turbulence Model for Flow Field of Aircraft Calculating in Three Dimensions
计算机科学, 2024, 51(6A): 230900053-9. <https://doi.org/10.11896/jsjcx.230900053>

面向 SW26010 间断有限元算法的多级并行计算

王晓忠¹ 张祖雨²

¹ 江苏联合职业技术学院 江苏 无锡 214000

² 江南大学物联网工程学院 江苏 无锡 214122

摘要 间断有限元算法(Discontinuous Galerkin Finite Element Method, DGM)是一种高精度的数值求解算法,针对电磁工程应用中 DGM 并行计算效率低、计算量较大的问题,提出了基于 SW26010 平台的并行 DGM 算法。通过区域分解、数据结构重构、热点函数从核并行计算、计算与通信重叠及从核缓冲优化技术完成了 DGM 算法的并行优化。实验结果表明,与基于 MPI 进程级的 DGM 并行算法相比,可以获得 46.8 的平均加速比。

关键词 间断有限元;数值模拟;并行计算;区域分解

中图分类号 TP391

Multi Level Parallel Computing for SW26010 Discontinuous Galerkin Finite Element Algorithm

WANG Xiaozhong¹ and ZHANG Zuyu²

¹ Jiangsu Union Technical Institute, Wuxi, Jiangsu 214000, China

² School of Internet of Things Engineering, Jiangnan University, Wuxi, Jiangsu 214122, China

Abstract The discontinuous Galerkin finite element method(DGM) is a high-precision numerical solution algorithm. Aiming at the problems of low efficiency and high computational complexity of DGM parallel computing in electromagnetic engineering applications, a parallel DGM algorithm based on the SW26010 platform is proposed. The parallel optimization of the DGM algorithm is achieved through region decomposition, data structure reconstruction, kernel parallel computing of hotspot functions, computation and communication overlap, and kernel buffering optimization techniques. Experiment results show that compared with the DGM parallel algorithm based on MPI process level, the proposed algorithm can achieve an average acceleration ratio of 46.8.

Keywords Discontinuous Galerkin finite element method, Numerical simulation, Parallel computing, Domain decomposition

1 引言

在电磁学领域中,数值方法和计算机硬件共同促进了计算电磁学的发展。随着计算机性能的大幅度提升,计算电磁学的重点逐渐转移到数值方法的研究上。计算电磁学主要有如下方法:有限元法(Finite Element Method, FEM)^[1]、有限体积法(Finite Volume Method, FVM)^[2]和 DGM。DGM 结合了 FEM 和 FVM 两种方法的优点,既采用了 FEM 中非结构网格剖分计算区域以便于模拟各种复杂的模型,又引入了 FVM 中的数值通量(Numerical Flux, NF)的概念,采用 NF 来连接网格之间的场。因此,DGM 的所有操作都是基于网格局部的,具有天然的并行性。目前,大规模的 DGM 计算面临内存开销大和计算时间长的问题,所以,DGM 的加速计算与优化值得深入研究。

在新型高性能体系架构下,出现了 MPI+X 多层次并行计算模型,即节点计算间基于 MPI 并行,而节点内采用共享内存编程模型,例如 OpenACC, CUDA, OpenMP 等^[3]。目前,关于非结构网格 DGM 并行计算的研究已有较多成果,文献[4]提出了一种非结构化混合网格的网格重排序方法,有效提高了并行计算收敛速度和整体效率,但随着分区数的增多,

不仅通信开销加大,而且算法收敛效率也会急剧减少。文献[5]提出了一种非结构网格的并行计算方法,该方法基于神威众核架构,简化了网格拓扑结构,提高了数据并行性,但数据映射产生了较多的额外开销,导致并行效率不高。文献[6]提出了非结构网格并行计算预处理方法,采用 Metis 进行负载均衡和任务划分,有效降低了通信开销,但面对大规模计算时,大量的数据重排序,导致整体加速效果不佳。文献[7]根据 SW26010 处理器的体系结构及非结构网格特点,提出了一种适应于任何网格规模的从核并行排序算法,解决了数据离散访存问题,但增加了额外的数据同步开销。

尽管上述各种方法存在一定的优点,但对于 DGM 算法并行计算效率低的问题仍未解决。综合考虑非结构网格的特点及 DGM 算法中热点函数的数据结构,本文基于“神威·太湖之光”平台^[8-9],将 DGM 算法与 SW26010 众核架构充分融合,对 DGM 并行计算麦克斯韦方程组进行研究。通过调整 DGM 中热点函数的数据结构,建立主从加速模型,利用 SW26010 处理器从核阵列加速热点函数计算。实验结果显示,在对 233 万网格的数值模拟时,本文方法达到了 73% 的并行效率,有效提高了 DGM 的并行效率,对计算电磁学的持续发展有重要意义。

基金项目:高等学校学科创新引智计划项目(B23008);未来网络科研基金项目(FNSRFP2021YB11)

This work was supported by the Programme of Introducing Talents of Discipline to Universities(B23008) and Future Network Research Fund Project(FNSRFP2021YB11).

通信作者:王晓忠(57439353@qq.com)

2 背景

2.1 SW26010 处理器

“神威·太湖之光”超级计算机采用的处理器是我国自主研发的异构众核处理器,称为 SW26010^[10]。

该处理器的架构如图 1 所示,每个处理器由 4 个核组(Core Groups, CG)构成,CG 之间通过片上网络进行互联,通常作为 4 个单独的节点。每个核组由 1 个运算控制核心

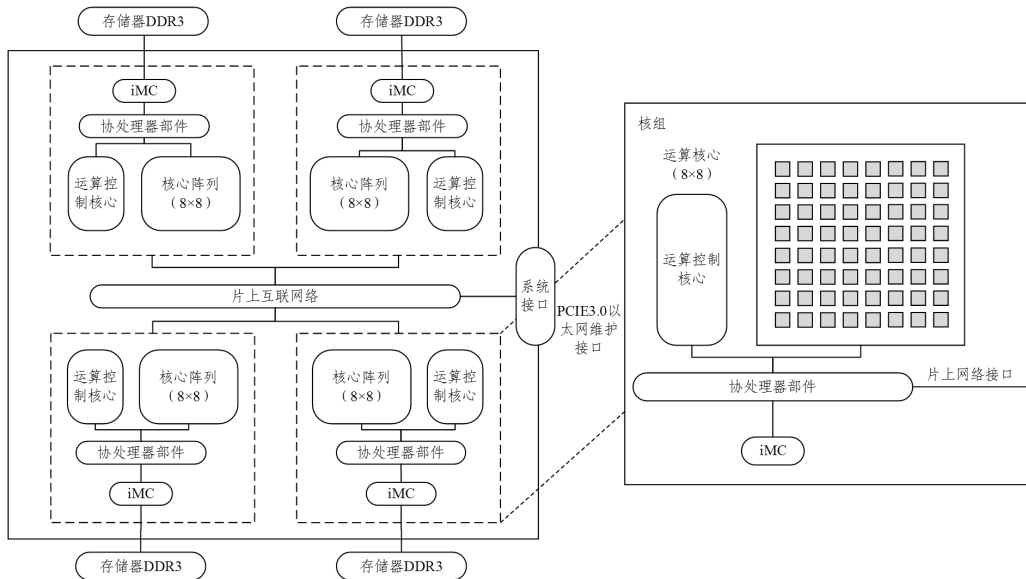


图 1 SW26010 处理器

Fig. 1 SW26010 processor

2.2 主从异步并行模型

根据 SW26010 处理器主从并行编程模型的特点,神威平台提供了 Athread 加速线程库。在该模型中, MPE 负责完成应用程序的通信、I/O 和部分串行代码的计算,而 CPEs 负责核心段的加速计算。在该核心段的计算任务完成之前, MPE 处于等待状态。主从异步模型如图 2 所示。

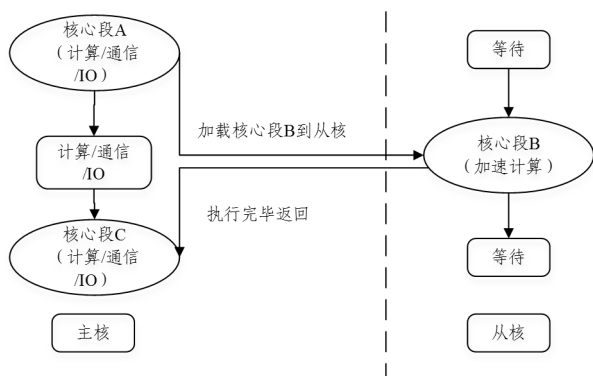


图 2 主从异构并行模型

Fig. 2 Master-Slave heterogeneous parallel model

3 DGM 的基本理论

从 Maxwell 方程组出发给出 DGM 的基本理论 Maxwell 旋度方程为:

$$\epsilon \frac{\partial E}{\partial t} - \nabla \times H + \sigma E = -J \quad (1)$$

$$\mu \frac{\partial H}{\partial t} - \nabla \times E + \sigma_m H = -J_m \quad (2)$$

式(1)和式(2)中, ϵ 为介电系数, μ 为磁导系数, σ 为电导

(Main Processing Element, MPE) 和 1 个从核阵列 (Computing Processing Elements, CPEs) 组成。CPEs 按照 8×8 的拓扑网络进行布局, 可以通过寄存器进行通信。整个芯片上集成了 260 个处理单元, 每个从核有 1 个大小为 64kB 的可重构局部数据存储空间 (Local Data Memory, LDM)。CPE 可以通过直接存储器访问 (Direct Memory Access, DMA) 方式批量访问主存。同时, DMA 是异步的, 可以通过计算与通信的重叠^[11] 来提升程序的性能。

率, σ_m 为磁导率, E 为电场强度, H 为磁场强度, J 为电流, J_m 为磁流。

通过 DGM 方法求解 Maxwell 方程, 得到一组离散化的矩阵方程如下:

$$\epsilon M^e \frac{\partial E^e}{\partial t} - S^e H^e - F h^e + \sigma M^e E^e = 0 \quad (3)$$

$$\mu M^e \frac{\partial H^e}{\partial t} - S^e E^e - F e^e + \sigma_m M^e H^e = 0 \quad (4)$$

式(3)和式(4)中, ϵ 为介电系数, μ 为磁导系数, e 为网格编号, M^e 为质量矩阵, S^e 为刚度矩阵, E^e 为电场自由度, H^e 为磁场自由度。 $F h^e$ 和 $F e^e$ 为数值通量, 且需要导入邻居网格数据, 其定义为:

$$F h^e = K_H^e [M f^e] \{H^{e+} - H^e\} - V_E^e [M g^e] \{E^{e+} - E^e\} \quad (5)$$

$$F e^e = K_E^e [M f^e] \{E^{e+} - E^e\} - V_H^e [M g^e] \{H^{e+} - H^e\} \quad (6)$$

式(5)和式(6)中, K_H^e , K_E^e , V_E^e , V_H^e 为数值通量系数, $[M f^e]$ 和 $[M g^e]$ 为相应矩阵, E^{e+} 为邻居网格的电场数据, H^{e+} 为邻居网格的磁场数据, E^e 为本网格的电场数据, H^e 为本网格的磁场数据。如果两个互为邻居的网格在同一个进程内, 可以直接使用邻居网格的数据, 如果不在同一个进程内, 则需要数据通信, 传输 E^{e+} 和 H^{e+} 。

4 基于 SW26010 的 DGM 多级并行策略

4.1 Metis 区域分解

DGM 并行计算电磁场量的流程如图 3 所示。并行计算的首要任务是区域分解^[10-12], 即将计算区域分解为若干子域, 通过 MPI 实现进程级并行。在每个时间迭代步内, 各进程分别迭代计算相应子域, 计算区域的边界进行数据交换。

本文使用图剖分软件 Metis 进行区域分解,以确保互为邻接的网格尽可能分配在同一个进程内,从而减少通信开销。在 Metis 完成分块后,还需要对网格数据进行重新排布,以保证每个进程的网格数据连续存储。

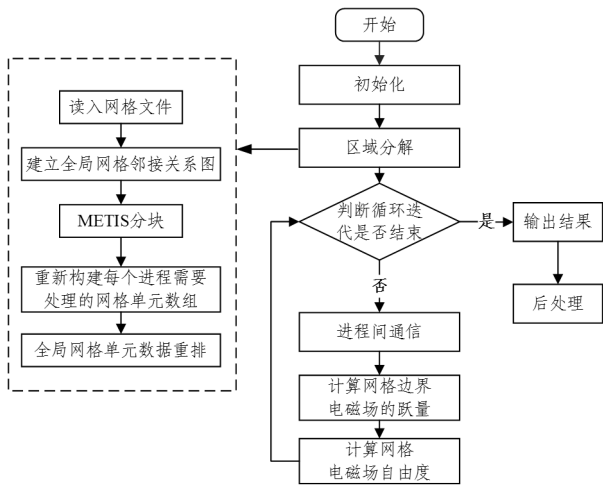


图 3 DGM 并行计算电磁场量的流程

Fig. 3 Process of DGM parallel calculation of electromagnetic field quantity

基于 Metis 区域分解的主要步骤如下:

- (1) 根据网格之间的邻接关系构建全局邻接矩阵。
- (2) 使用 Metis 图分割方法,将整体网格分割为 N 个子域,其中 N 为任务使用的进程数。
- (3) 使用划分好的网格集合重新构建每个进程所需要处理的网格数组。
- (4) 重排网格原始数据,使得每个进程的网格数据连续存储。

在区域分解过程中,需要特别注意网格之间的交界面信息。区域间通信映射是区域与区域之间相邻网格的信息交换,如图 4 所示,每次迭代时,区域 A 需要从区域 B 中获取与 A 相邻的电磁场信息,而区域 B 需要从区域 A 获取与 B 相邻的电磁场信息,因此需要交换相邻的网格电磁场信息。

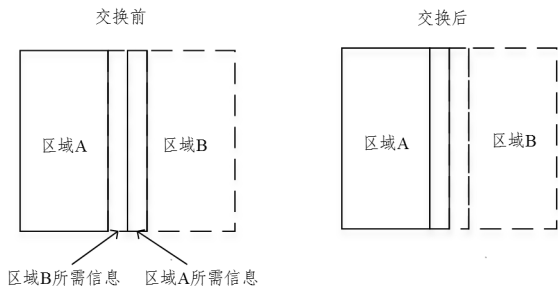


图 4 区域间通信映射

Fig. 4 Interarea communication mapping

在处理区域间的通信映射时,需要对进程内的所有网格进行遍历,通过邻接矩阵确定邻接网格的数据是否需要通信,将需要通信的数据保存在通信缓冲区。之后,使用 MPI_Sendrecv 接口进行数据交换。

4.2 DGM 的热点函数从核并行计算

4.2.1 数据结构重构

非结构网格数据关系的依赖性和对数据访问的离散性,导致其在并行计算时出现局部性差、数据相关性强和离散访存等问题,从而使得 DGM 在众核处理器上进行并行计算的

难度增大。为解决上述问题,本文根据非结构网格的数据依赖性提出了面向从核架构的数据重构模型,在预处理阶段新建了一个数据结构,该数据结构存放计算电磁场量所需的网格信息。

本文重构数据结构的步骤如下:

- (1) 遍历每个核心计算部分,找到所有与计算网格电磁场量的相关数据,如进程编号、邻接关系表、电磁场自由度等。
- (2) 针对网格循环新建数据结构,结构成员包括步骤 1 中核心计算所需的相关数据。
- (3) 在预处理阶段,根据原拓扑关系,将相应的数据重新映射至新的数据结构中对应的成员变量。
- (4) 从核使用新的数据结构进行并行计算。

在数据重构之后,进程内网格信息按照新的结构体连续存储。在从核并行计算阶段,将该结构体数组进行任务划分,从核通过 DMA 将划分后的数据段加载至 LDM 中,从而解决了非结构网格局部性差、数据相关性强及离散访存的问题。

4.2.2 任务划分

由于从核的 LDM 有限,不可能存放所有的网格数据,因此,实现从核并行计算的关键任务是任务划分。每个从核只需要将负责计算的网格的相关数据循环从主存通过 DMA 加载至 LDM 中,计算结束之后再通过 DMA 将结果加载至主存。

划分示意图如图 5 所示,针对最外层网格循环进行任务划分,每个从核负责计算的网格编号为 $elem_no$,其计算式为:

$$elem_no = i \times 64 + cpe_no \quad (7)$$

其中, i 代表循环次数, cpe_no 代表从核编号。

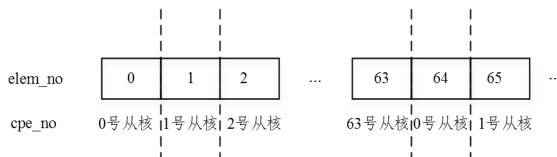


图 5 任务划分

Fig. 5 Task partitioning

4.2.3 循环融合

本文采用的网格模型为三维四面体,在核心计算段对数值通量的求解过程中,网格在 X, Y, Z 维的数值通量的循环模式相同,依赖数据(如参考单元微分矩阵)部分重复,从而导致数据局部性下降。通常可以采用循环融合的优化策略以改善数据局部性,因此,在从核并行计算电磁场自由度阶段,我们将多个数值通量的计算融合到一个核心段中。循环体数目的减少可消除循环并行时冗余的同步开销,减少中间数据的主从传输,进一步提升计算效率。

4.3 计算与通信重叠

随着运行规模的增大,进程间通信开销会越来越大,因此,降低程序的通信开销是性能提升的关键。通常,如果网格至少有一个网格面需要通信,则称为通信网格;反之,则称为非通信网格。因此,计算网格的电磁场自由度可分为两部分:计算通信网格的电磁场自由度和计算非通信网格的电磁场自由度。

在单次迭代步的计算阶段,计算非通信网格的电磁场自由度无需等待通信结果,而计算通信网格的电磁场自由度则需要等待目标进程通信的数据。因此,本文采用了“神威·太湖之光”的主从异步运行。在主核进行边界通信的同时,从核阵列计算非通信网格的电磁场自由度,将通信时间隐藏在从核计算中,从而缩短程序运行时间。当以上两个过程结束后,

再次使用从核阵列并行计算网格边界场的跃量及通信网络的电磁场自由度。计算与通信重叠的示意图如图 6 所示。

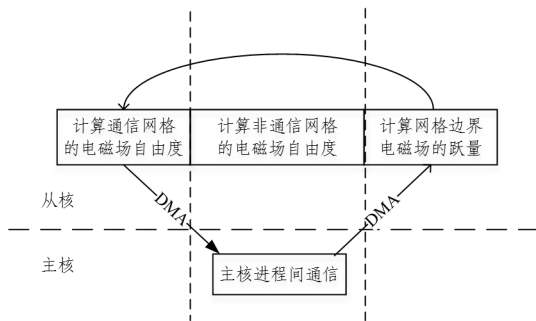


图 6 计算与通信重叠示意图

Fig. 6 Diagram of Computation and communication overlap

计算与通信重叠^[13-15]的步骤如下:

(1)将网格编号按照是否需要通信分别存储到两个下标队列中。

(2)根据两个下标队列将计算电磁场自由度的函数分为计算通信网络磁场自由度和计算非通信网络的电磁场自由度。

(3)在使用从核函数计算时,将两个下标队列作为参数传递给对应的从核函数。

(4)修改计算电磁场量循环的调用。由于计算非通信网络的电磁场自由度时需要网格边界电磁场的跃量,因此在计算电磁场量循环之前需要进行一次预处理,即边界通信和计算网格边界电磁场量。计算电磁场量的循环如图 7 所示。

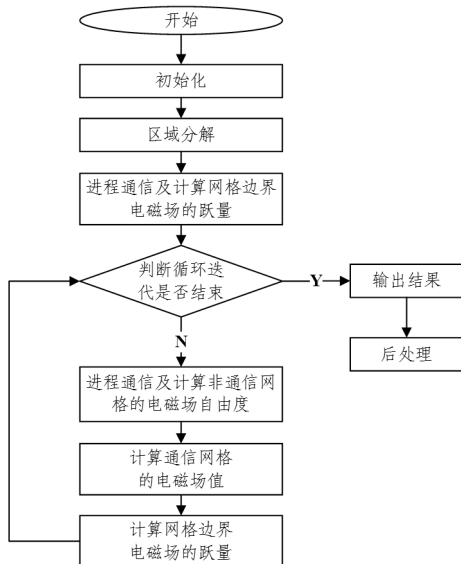


图 7 计算电磁场量流程图

Fig. 7 Flow chart of calculating electromagnetic field quantities

4.4 软件 Cache 的优化

网格数据是按照进程内网格编号连续存储的。然而,对于单一的网格,其邻接网格编号并不连续^[16]。例如,0 号网格相邻的网格编号分别为 24787,24784,33465,41514。因此,邻接网格的相关数据是离散存储的,计算一个网格电磁场的跃量,需要访问 5 次主存,即自身的电磁场数据及 4 个邻接网格的电磁场数据,使得访存开销更大。为了解决这一问题,本文在从核 LDM 中设计类似于 x86 的 Cache 的数据结构,通过 DMA 进行数据装填,从而减少了从主存取数据的次数,提高了访存效率。这种设计本质上是对被访问数据进行了重排序。

基于软件控制的从核缓冲优化步骤如下:

(1)数据分块。在已经完成进程间基于 Metis 区域分解的基础上,进程内部再次使用 Metis 对数据进行分块,分块大小 θ 以 LDM 存储空间为标准适当选取,子块上的网格数 Num 的计算公式为:

$$Num = \frac{\theta}{E_{buffer}} \quad (8)$$

其中, E_{buffer} 为每个网格的电磁场数据所需的存储空间。

每个进程内的子块数为:

$$P = \frac{E_{total}}{Num} \quad (9)$$

其中, E_{total} 为进程内网格总数。

每个从核需要负责计算的子块编号为:

$$PartNo = i \times Cpe + CpeNo \quad (10)$$

其中, i 为循环次数, Cpe 为计算使用的从核数, $CpeNo$ ($0 \leq CpeNo < 64$) 为从核编号。

(2)局部重排。根据 Metis 分块文件建立一个索引表,用于指示每个子块需要处理的网格编号,按照索引表指示,将相应的网格数据并行分段拷入 LDM,从而使得子块内网格数据在从核 LDM 中是连续存储的。

经测试,当 $Num=20$ 时,有 68% 互为邻接的网格分在同一个子块内,在计算子块内的网格电磁场的跃量的阶段中,有 68% 的可能性在从核 LDM 访问到邻接网格,从而有效提高了访存率。

5 实验结果分析

本文基于 SW26010 处理器,采用 swgcc710 编译器编译 C/C++ 程序,使用 16~256 个核组测试了 233 万网格规模的数值模拟。

5.1 热点函数的加速效果

性能测试指标主从加速比 S_n 的计算公式为:

$$S_n = \frac{T_s}{T_n} \quad (11)$$

其中, T_s 为纯主核耗时, T_n 为主从核耗时。

为了验证加速效果,采用数据结构重构及从核并行计算后,计算电磁场自由度的主从加速比,如表 1 所列。

表 1 计算电磁场自由度的主从加速比

Table 1 Calculation of master-slave acceleration ratio of degrees of freedom of electromagnetic field

核组数	T_s/s	T_n/s	S_n
4	4103.451	86.855	47.245
16	1024.834	21.808	46.993
32	512.330	10.912	46.951
64	256.134	5.466	46.859
128	128.161	2.738	46.802
256	64.044	1.375	46.575

实验结果显示,计算电磁场自由度的主从加速比平均为 46.8,单核组内有 64 个从核,并行效率可达 73%,热点函数的加速效果比较理想。随着进程数的增加,单个进程分配的网格数大幅度减少,从核加速效率会相应降低。

5.2 通信优化效果

在区域分解时按照网格编号将所有网格大致均匀地分配到不同的进程中,这种分配策略虽然实现了计算量上的负载均衡,但未考虑网格之间的邻接关系,大量共用网格面的网格被分到不同的进程中,在后续的计算过程中需要大量的通信。采

用 Metis 优化前后通信时间在总时间中的占比情况如图 8 所示。

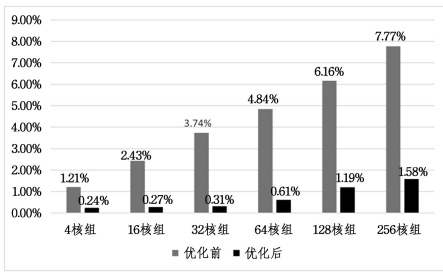


图 8 Metis 优化前后通信时间占比总时间的情况

Fig. 8 Proportion of communication time before and after Metis optimization

通过图 8 可以看出,使用 Metis 进行区域分解能够明显缩短通信时间,但随着进程数的增加,通信开销依旧逐渐增大。经过 Metis 区域分解后,通信时间小于计算非通信网格电磁场自由度的时间,所以,本文利用“神威·太湖之光”的主从异步运行,采用计算与通信重叠方法,将通信时间隐藏在从核计算中,从而解决通信开销大的问题。

5.3 软件 Cache 的优化效果

在软件 Cache 优化后,计算网格边界电磁场的跃量的平均主从加速比由 3.7 增加到 10.6。当 $Num = 20$ 时,从核缓冲优化效果如表 2 所列。

表 2 从核缓冲优化前后的主从加速比

Table 2 Master-Slave acceleration ratio before and after optimizing from kernel buffer

核组数	优化前	优化后
4	3.783	10.734
16	3.771	10.625
32	3.731	10.504
64	3.705	10.554
128	3.621	10.770
256	3.534	10.609

经过从核缓冲优化,在计算网格边界电磁场的跃量时,68%的网格可以在从核 LDM 上访问到邻接网格,这类网格无需再次访问主存,有效提高了访存率。从表 2 可以看出,经过从核缓冲优化,计算网格边界电磁场的跃量的主从加速比有了显著提高。

结束语 本文从提高 DGM 求解麦克斯韦方程组计算速度的角度出发,基于国产超级计算机“神威·太湖之光”,利用其异构众核体系结构的优势,实现了 DGM 算法的并行加速,充分发挥了国产异构众核平台的特点和优势。通过 Metis 区域分解及计算与通信重叠方法,将进程间的通信开销隐藏在从核计算中,通过对非结构网格的数据重构及合理的任务划分,使得 DGM 算法中热点函数的平均主从加速比达到了 46.8,且充分利用了从核的 LDM,有效提高了访存率。但随着计算节点的增多,节点间数据传输将达到性能瓶颈,因此下一步将进行强扩展性研究。

参考文献

[1] CAI Y M, ZHANG J B, YU W Z. A Predictor-Corrector Method for Power System Variable Step Numerical Simulation[J]. IEEE Transactions on Power Systems, 2019, 34(4): 3283-3285.
 [2] ZHOU N, ZHOU H, HOPPE D. Containerization for High Per-

formance Computing Systems: Survey and Prospects[J]. IEEE Transactions on Software Engineering, 2022, 49(4): 2722-2740.
 [3] MARTINEZ-FERRER P J, YZELMAN A N, BELTRAN V. A Native Tensor-Vector Multiplication Algorithm for High Performance Computing[J]. IEEE Transactions on Parallel and Distributed Systems, 2022, 33(12): 3363-3374.
 [4] ZAHID F, TAHERKORDI A, GRAN E G, et al. A Self-Adaptive Network for HPC Clouds: Architecture, Framework, and Implementation[J]. IEEE Transactions on Parallel and Distributed Systems, 2018, 29(12): 2658-2671.
 [5] XIAO G Q, LI K I, CHEN Y D, et al. CASpMV: A Customized and Accelerative SpMV Framework for the Sunway TaihuLight[J]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(1): 131-146.
 [6] LI X, XU L, YANG Z, LI B. The PML Boundary Application in the Implicit Hybridizable Discontinuous Galerkin Time-Domain Method for Waveguides[J]. IEEE Microwave and Wireless Components Letters, 2021, 31(4): 337-340.
 [7] JIANG C T, XIA M M, ZHOU H. Dispersion Analysis and 3-D Wavefield Modeling of Lattice Boltzmann Model[J]. IEEE Transactions on Geoscience and Remote Sensing, 2024, 62: 1-12.
 [8] CHEN Y D, XIAO G Q, YANG W D, et al. Exploiting Hierarchical Parallelism for Sparse Tensor-Vector Multiplication on Heterogeneous Parallel Systems[J]. Chinese Journal of Computers, 2024, 47(2): 441-455.
 [9] XU S, WANG W, ZHANG J, et al. High Performance Computing Algorithm and Software for Heterogeneous Computing[J]. Journal of Software, 2021, 32(8): 2365-2376.
 [10] XIA T, FU G L, QU S R, et al. Optimization of Parallel Computation on Sparse Matrix-Vector Multiplication with High Predictability[J]. Journal of Computer Research and Development, 2023, 60(9): 1973-1987.
 [11] CREMONESI M, FRANCI A, IDELSOHN S, et al. A state of the art review of the particle finite element method(PFEM)[J]. Archives of Computational Methods in Engineering, 2020, 27(5): 1709-1735.
 [12] ALHIJAZI M, ZEESHAN Q, QIN Z, et al. Finite element analysis of natural fibers composites: A review[J]. Nanotechnology Reviews, 2020, 9(1): 853-875.
 [13] WU S S, DONG X S, WANG Y F, et al. UPPA: Unified Parallel Programming Architecture for Heterogeneous Systems[J]. Chinese Journal of Computers, 2020, 43(6): 990-1009.
 [14] AN C Q, DU H, LI Q, et al. Memcached Optimization on High Performance I/O Technology[J]. Journal of Computer Research and Development, 2018, 55(4): 864-874.
 [15] LI Y Y, XUE W, CHEN D X, et al. Performance Optimization for Sparse Matrix-Vector Multiplication on Sunway Architecture[J]. Chinese Journal of Computers, 2020, 43(6): 1010-1024.
 [16] SONG G H, GUO S Z, ZHAO J, et al. Automatic Mixed Precision Optimization for Stencil Computation[J]. Journal of Software, 2023, 34(12): 5704-5723.



WANG Xiaozhong, born in 1981, M. S., associate professor. His main research interests include control theory and control engineering, and high performance computing.