

MPI+OpenMP 混合编程在三维电磁辐射计算中的应用

唐 龙, 何 冰, 张 武

(上海大学 计算机工程与科学学院, 上海, 200444)

摘 要: 在多核 CPU 集群环境下, 采用 MPI+OpenMP 混合算法, 实现了 FDTD 算法电磁场点源辐射数值模拟计算, 有效解决了传统的 FDTD 算法在大数据电磁辐射计算中空间和时间上的不足的问题, 并运用区域分割、子区域数据通讯、合并嵌套等方法提高程序的并行性, 最后对计算结果进行正确性验证. 通过在上海大学高性能集群上与 MPI 算法进行性能对比, 结果表明, 利用 MPI+OpenMP 获得了较高的加速比, 并有效地节省计算资源, 加快计算速度.

关键词: 有限时域差分法; MPI+OpenMP; 电磁辐射计算; 并行计算

中图分类号: TN402

文献标识码: A

文章编号: 1000-7180(2014)08-0127-05

Application and Research on Three-dimensional Electromagnetic Radiation Calculation Using MPI+OpenMP Paradigms

TANG Long, HE Bing, ZHANG Wu

(School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China)

Abstract: We simulate the electromagnetic radiation calculation based on the multi-core CPU cluster parallel structure by using MPI + OpenMP hybrid parallel algorithm. This is an effective solution to the traditional FDTD method in the calculation of the electromagnetic radiation on the problem of inadequate large data space and time. What is more, we by means of regional segmentation, sub-regional data communications, consolidation and other methods to improve procedures nested parallelism, and finally verify the correctness of the calculation results. According to compare the computational time with MPI Algorithm on the high performance computer of Shanghai University, We draw the conclusion that The MPI + OpenMP hybrid parallelized code is indicated to be practical in achieving high speedup and great runtime reductions.

Key words: finite difference time domain method; MPI + OpenMP; electromagnetic radiation calculations; parallel Computing

1 引言

FDTD^[1]是 Finite Difference Time Domain 的简写. 现在该方法已经成为求解麦克斯韦方程的常用方法. 它是一种全矢量法, 可以很自然地给出用户所需要的时域和频域信息. 这是该方法在电磁学和光子学应用中所拥有的独特优点.

FDTD 算法在时间和空间上都是离散的. 因此电磁场和所感兴趣结构材料必须在由所谓的 YEE

元胞构成的网格上予以描述. 麦克斯韦方程求解在时间上是离散的, 所用的时间步长通过光速与网格大小紧密相关. 在网格尺寸趋于零的极限情况下, 这项算法准确无误地描述麦克斯韦方程.

该算法在电磁辐射的多个领域获得了广泛应用, 比如辐射天线分析、散射计算、电子封装、雷达等, 而且随着高性能计算的不断发展, MPI 并行算法很好地解决了 FDTD 算法运算时间过长这个问题, 但是随着计算量不断增大, MPI 算法会使节点

收稿日期: 2013-11-03; 修回日期: 2013-12-28

基金项目: 国家自然科学基金项目(11002086); 国家自然科学基金重大研究计划项目(J1330116)

内单进程负担逐渐增大,从而我们运用 MPI + OpenMP 混合算法来解决这个问题,并利用分布共享存储的特点提高并行加速比和可扩展性。

2 FDTD 算法

2.1 FDTD 算法简介

FDTD 算法是基于麦克斯韦方程的数值计算方法,该算法通过 Yee 元胞,利用蛙跳式计算方法,使电场和磁场在空间上相互交替分布,并对半个步长内的电磁场进行抽样,逐渐完成整个电磁场的递推。

利用 FDTD 法计算电磁场时,归一化的电场和磁场差分方程为

$$\begin{aligned} & E_{n+1,x}\left[i+\frac{1}{2},j,k\right] \\ &= CA\left[i+\frac{1}{2},j,k\right]E_{n,x}\left[i+\frac{1}{2},j,k\right]+ \\ & CDCB\left[i+\frac{1}{2},j,k\right]H_{n+\frac{1}{2},z}\left[i+\frac{1}{2},j+\frac{1}{2},k\right]- \\ & H_{n+\frac{1}{2},z}\left[i+\frac{1}{2},j-\frac{1}{2},k\right]-H_{n+\frac{1}{2},z}\left[i+\frac{1}{2},j,k+\frac{1}{2}\right] \\ & H_{n+\frac{1}{2},z}\left[i+\frac{1}{2},j,k-\frac{1}{2}\right] \end{aligned} \quad (1)$$

$$\begin{aligned} & H_{n+\frac{1}{2},x}\left[i,j+\frac{1}{2},k+\frac{1}{2}\right] \\ &= H_{n+\frac{1}{2},x}\left[i,j+\frac{1}{2},k+\frac{1}{2}\right]+ \\ & CD\left[E_{n,y}\left[i,j+\frac{1}{2},k+1\right]-E_{n,y}\left[i,j+\frac{1}{2},k\right]-\right. \\ & \left.E_{n,z}\left[i,j+1,k+\frac{1}{2}\right]+E_{n,z}\left[i,j,k+\frac{1}{2}\right]\right], \end{aligned} \quad (2)$$

式中,

$$CA(i,j,k) = (1 - \frac{\sigma(i,j,k)t}{2\epsilon(i,j,k)}) / (1 + \frac{\sigma(i,j,k)t}{2\epsilon(i,j,k)}) \quad (3)$$

$$CB(i,j,k) = \epsilon_0 / (\epsilon(i,j,k) + (\frac{\sigma(i,j,k)t}{2})) \quad (4)$$

$$CD = t / \delta \sqrt{\epsilon_0 \mu_0} t \quad (5)$$

其他方向 E_y, E_z, H_x, H_y 方程也可以按照式 (1)(2) 类似推得,当收敛条件满足时,就会得到稳定的电场值和磁场值,该流程如图 1 所示。

2.2 计算参数初始化与边界条件

FDTD 算法的显示迭代需要初始的场值和边界条件,传统定义初始场值的方法是把空间内的所有场值都定义为 0,然后将两个连续时间步的场值储存作为下一步迭代的初始场值,至于边界条件^[2],则

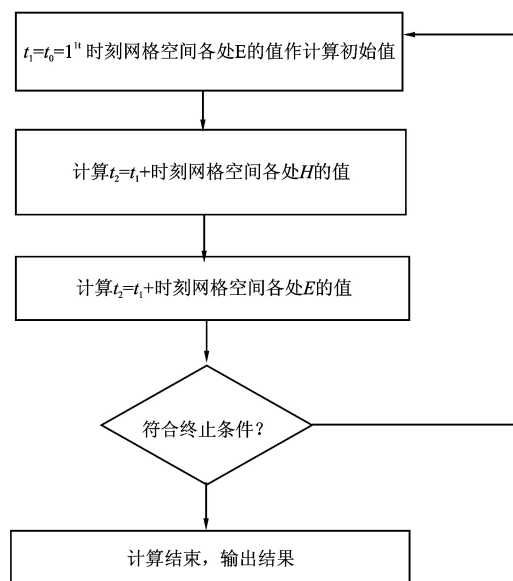


图 1 FDTD 算法流程图

有多种选择,常用的吸收边界条件有 Mur 吸收边界,完全匹配层(PML)和各向异性完全匹配层(UMPL),随着 FDTD 算法的完善,这 3 种匹配层都有着良好的吸收效果。

在我们的实验中,采用的是各向异性完全匹配层(UPML)作为吸收边界条件,UPML^[3]继承了 PML 吸收层良好的吸收特性,但是与基于一组修正方程的 PML 吸收层不同的是 UPML 是直接基于 Maxwell 方程的吸收层.这保证了 FDTD 算法的计算模型与 Maxwell 方程组具有良好的形式一致性,从而更容易被理解和编程实现。

3 FDTD 算法并行实现

3.1 MPI+OpenMP 模型介绍

MPI 编程模型是一个用于开发基于消息传递的并程序接口标准,作用于异构网络环境中,它提供了可靠地传输机制,并利用安全性通道实现了进程组内所有任务之间的通讯、数据交换和处理^[4].但是由于 MPI 是通过进程间通信的方式进行并行计算,这导致了并行效率较低、内存开销大、不直观、编程麻烦等缺点.而 OpenMP 编程模型^[5]是利用共享存储模型,采用 fork-join 执行模式,将一个进程任务分成若干个可并行执行的线程.在单节点中,当主线程运行到 OpenMP 并行域后,会产生不同线程队列,同时进行计算,从而达到并行的效果.但是,正因为 OpenMP 编程模型只能在单节点进行并行,计算能力受到了极大的限制。

MPI+OpenMP 编程模型则是这两种模型的结合,即多台机器间采用 MPI 分布式内存并行且每台机器上分配一个 MPI 进程,而各机器又利用 OpenMP 进行多线程共享内存并行^[6],这样可以减少 MPI 并行的进程数,从而减少消息传递的次数;每个节点上的 OpenMP 并行又可以节省内存的开销.所以这种混合并行方式比单一的并行方式具有一定优势.

3.2 MPI 并行通信设计

由式(1)、(2)可知,FDTD 算法中每一步电场值大小只依赖于上一步的场强大小和其周围临近网格的磁场值,具有显著的局域性.因此,MPI 只需要将计算区域划分为多个子区域,每个子区域分配给 MPI 节点,并在时间步内临近的网格进行电场值与磁场值的数据交换,最后在该时间步内各个节点进行数据同步^[7].为了获得高效率,高加速比的并行,我们按 X 方向进行网格划分,并将每个进程平均分配到各个节点中,在 X 方向占优的长方体模型中,这种划分可以减少节点之间的通讯时间.划分为多个子区域之后,数据通讯如图 2 所示,计算进程 I 中的 H_z 时需要周围的电场值 E_x, E_y ,而 E_x 的电场值在进程 II 中,此时,进程 II 与进程 I 进行通讯将 E_x 的值传输给进程 II,这样就完成了一次数据交换,其他各个方向的数据都可以此种方式进行数据通讯.

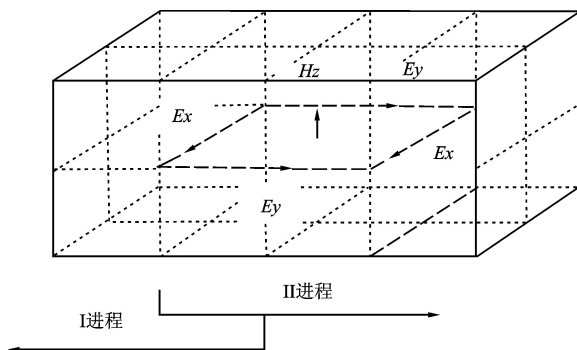


图2 进程间数据通信

3.3 OpenMP 线程设计及优化

在电磁场计算中,更新电场强度和磁场强度只依赖于上一步的电磁场数据,具有天然的并行性,因此,在多核计算中,可以采用 OpenMP 增加程序的并行性.

对于内存利用较好的情况,负载平衡可能并不是很好;同样,较优的负载平衡的策略也可能对内存利用不利.因此,在考虑性能优化时,必须在优化内存利用和负载平衡之间进行折中,通过对性能的测

量找到能够得到最佳结果的方法.

在 OpenMP for 结构中,使用 schedule 子句可以将循环调度传达给编译器和运行时库^[8]: #pragma omp for schedule (type, [size]). 如果指定 size(块大小),则 size 参数必须是随循环变化的正整数常量或者整数表达式.它可能对性能带来不利影响.随着块大小的减小,线程用于从任务队列中获取任务的时间增加,从而使访问任务队列的开销增加,降低性能,并可能抵消负载平衡带来的性能提升. Type 调度有 4 种,static、dynamic、guided、runtime,由于实验环境中各个 CPU 运算能力差别不大,所以采用了 static 调度方式,使每个线程平均的执行计算任务.

```
#pragma omp for schedule(static)
for(i = 1; i < ie_tot; i++)
for(j = 0; j < je_tot; j++)
for(k = 0; k < ke_tot; k++)
{.....}
```

为了增加 OpenMP 程序的负载平衡和可扩展性,我们通过循环融合调整了代码结构,即通过合并嵌套循环将 3 层循环合并成 2 层.

```
j_end = je_tot;
ij_end = (ie_tot) * (je_tot);
ij_start = j_end + 1;
#pragma omp for schedule(static)
for(ij = ij_start; ij < ij_end; ij++)
{
i = ij / j_end;
j = ij % j_end;
if (! i) continue;
for(k = 0; k < ke_tot; k++)
{.....}
}
```

很明显, x, y, z 方向的计算都可以用类似的方式进行迭代计算来提高程序并行性.

3.4 MPI+OpenMP 混合编程实现

随着运算规模的不断增大,共享存储处理器的进程开销会限制并行性能.通过 OpenMP 和 MPI 混合编程,可以很好地解决这个问题,通过在每个 MPI 节点进行多线程计算,并选择最优的 MPI 节点数和每个 MPI 节点最优线程数,使 FDTD 算法具有更好的并行性.由于每个 MPI 节点的计算量被多个线程分担,从而降低了节点通信开销,缓解了进程压力.算法伪代码如下:

```
//MPI 并行环境初始化
MPI_Init (&.argc, &.argv);
MPI_Comm_rank (MPI_COMM_WORLD, &.rank);
```

```

MPI_Comm_size ( MPI_COMM_WORLD , &size );
MPI_Get_processor_name ( processor_name ,
&namelen );
printf( " Node %3d, Name:%s...\n",rank,proces-
sor_name );
MPI_Barrier(MPI_COMM_WORLD);
node.init( x_size /size, y_size , z_size );//网格数据初
始化
UPMLcompute() ;//设置吸收边界
//按时间步进行迭代
for( int i = 0; i < max_step ; i++)
{
//利用 OpenMP 进行并行计算
#pragma omp for schedule(static)
for( i = 1;i < ie_tot;i++)
{..... Compute;//根据公式计算电场值和磁场值
}
//向相邻节点发送数据
MPI_Send ( datasend_right , data_size , MPI_DOUB-
LE , this->NodeRank+1 , 1 , MPI_COMM_WORLD );
//接受相邻节点数据
MPI_Recv( datarecv_right , data_size , MPI_DOUBLE
, this->NodeRank+1 , 1 , MPI_COMM_WORLD ,
&status );
MPI_Barrier( MPI_COMM_WORLD );
}
MPI_Finalize();
Out( EHdate );//输出计算结果

```

4 实验与分析

4.1 实验环境

至强 4000 系统计算部分刀片计算节点 Normal 集群由 140 台 IBM X240 刀片式服务器构成, 每台刀片式服务器包含 2 颗 intel E5-2690 CPU, 64 GB 共享内存, 每个节点由 16 个核组成. 系统软件环境包括 Red Hat 4.1.2-48 操作系统、支持 OpenMP 制导 C/C++ 编译器和 MPICH2 并行环境.

4.2 模型验证与分析

为了检验算法的正确性, 对点源辐射问题进行了计算. 激励源频率为 900 Hz, 采用高斯脉冲, 网格大小为 0.02 mm, 规模为 $50 \times 50 \times 50$, UPML 吸收层设置为 10 个网格大小, 图 3、4 为点源辐射 XOY 面电场强度计算结果, 从第 34 层到 37 层成扩散状, 电磁波向外逐渐辐射.

4.3 并行结果分析

程序并行化后, 计算规模为 $1\,000 \times 400 \times 400$, 采用

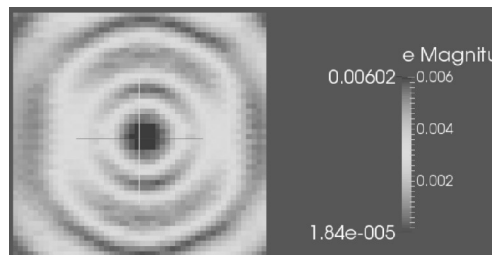


图 3 第 34 层 XOY 截面电场强度

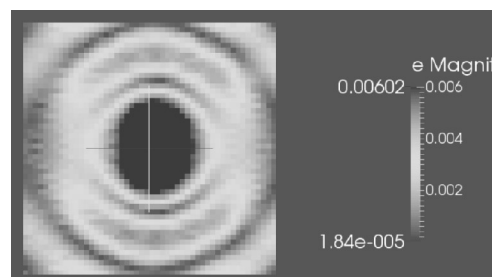


图 4 第 37 层 XOY 截面电场强度

MPI 编程模式在计算 100 步的计算结果见表 1 所示.

表 1 MPI 编程模式 FDTD 算法计算结果

核数	整体计算时间/s	计算时间/s	通信时间/s
1	7 222	7 222	0
8	926	915	11
16	475	461	14
32	245	221	24
48	161	139	22
64	132	105	27

表 1 中给出了 MPI 方法在点源辐射的总时间和计算通讯时间. 对 MPI 程序, 我们每个节点执行 16 个进程, 从表中可以看出, 随着进程数增加, 计算的加速比成线性增长, 可以看出该电磁辐射程序在分布式并行环境中有良好的可扩展性, 但随着通讯开销的增加, 加速比增加速度开始缓慢下降.

本文的有限时域差分程序主要由参数初始化、时间步计算、吸收边界计算、电场磁场值更新、数据处理组成, 在电磁场更新阶段利用 OpenMP 编程模式提高程序并行性. 设置线程数为 8, 每个节点运行 2 个进程, 即 1 个进程对应 8 个核, 计算结果见表 2.

表 2 MPI+OpenMP 编程模式 FDTD 算法计算结果

核数	整体计算时间/s		加速比	
	MPI	MPI+OpenMP	MPI	MPI+OpenMP
1	7 222	7 222	1	1
8	926	1 003	7.8	7.2
16	475	505	15.2	14.3
32	245	251	29.4	28.7
48	161	155	44.8	46.4
64	132	127	54.7	56.8

从表 2 可以看出, 当计算核数较少时, 纯 MPI 程序比 MPI+OpenMP 程序有更好的并行性, 这是

因为进程数较少时,MPI 通讯量比较小,而 OpenMP 程序会相应增加线程开销,但随着核数的增多,MPI 的增长幅度会下降,这是由于随着进程的增加,MPI 进程通讯和同步开销逐渐增大,而 MPI+OpenMP 程序却因为通讯量较少,所节省的时间抵消了线程的开销,从而取得了更好的并行性.通过对比两种并行方法,当计算核数小于 32 时 MPI 程序性能要优于 MPI+OpenMP 程序,而随着计算核数的增加,MPI+OpenMP 程序的优势会逐渐显露出来,获得更好的加速比与扩展性.

5 结束语

随着高性能计算的发展,为 FDTD 算法提供了更好的应用空间,而各种并行方式的出现,使混合编程模式逐渐成为高性能计算的主流.本文分析了 MPI+OpenMP 混合编程模型的性能,并基于 FDTD 算法对三维电磁辐射进行数值模拟,通过计算结果可以发现,MPI+OpenMP 混合编程模型可以更好的利用分布式共享存储的优势,提高并行效率.

参考文献:

- [1] 葛德彪,闫玉波. 电磁波时域有限差分方法[M]. 西安:西安电子科技大学出版社,2002.
- [2] M F Su, I E Kady, D A. Bader, et al. A novel FDTD application featuring openMP—MPI hybrid parallelization [C]//Proceeding of the 2004 International Conference on Parallel processing Montreal, Canada: IEEE, 2004:373-379.

(上接第 126 页)

物线,从仿真图中可以看出高速飞行目标的运动特征符合理论分析,表明该算法能够比较精确的对高速飞行目标进行跟踪.

参考文献:

- [1] 张婷,许可. 一种提高海洋雷达高度计重跟踪估值精度的处理方法[J]. 遥感技术与应用,2007,22(3):422-427.
- [2] 侯静,景占荣,羊彦. 远距离干扰环境下目标跟踪的扩展卡尔曼粒子滤波算法[J]. 电子与信息学报,2013,35(7):1587-1592.
- [3] 严世华,何永强,周玉龙. 基于红外鱼眼探测系统的运动目标模型[J]. 光学学报,2012,32(9):82-87.
- [4] 王学斌,徐建宏,张章. 卡尔曼滤波器参数分析与应用方法研究[J]. 计算机应用与软件,2012,29(6):

- [3] Gedney S D. An anisotropic perfectly matched layer absorbing media for the truncation of FDTD lattices [J]IEEE Trans Antennas and Propagation, 1996(44):1630-1639
- [4] 郑汉垣,刘智翔,封卫兵,等. MPI+OpenMP 混合编程模型在大规模三对角线性方程组求解中的应用[J]. 微电子学与计算机,2011,28(8):158-161.
- [5] 罗秋明,明仲,刘刚,等. OpenMP 编译原理及实现技术[M]. 北京:清华大学出版社,2012.
- [6] 赵永华,迟学斌. 基于 SMP 集群的 MPI+OpenMP 混合编程模型及其有效实现[J]. 微电子学与计算机,2005,10(22):7-11.
- [7] Cai D, Li Y T, Nishikawa K, et al. Three-dimensional electro-magnetic particle-in-cell code using high performance fortran on PC cluster[J]. High Performance Computing, Lecture Notes in Computer Science, 2002(2327):515-525.
- [8] Li Jianjiang, Shu Jiwu, Chen Yongjian, et al. Analysis of factors affecting execution performance of openMP programs [J]. Tsinghua Science and Technology, 2005, 10(3): 304-308.

作者简介:

唐 龙 男,(1989-),硕士研究生.研究方向为电磁辐射计算.

何 冰 男,(1974-),博士,讲师,硕士研究生导师.研究方向为高性能计算.

张 武 男,(1958-),博士,教授,博士生导师.研究方向为并行计算方法与应用.

212-215.

- [5] 尹俊,郭迎清. 神经网络和卡尔曼滤波融合的航空发动机传感器故障诊断[J]. 计算机测量与控制,2011,19(8):1936-1938.
- [6] 商临峰,杨小军,邢科义. 局域跟踪的测速雷达网量测融合弹道解算[J]. 电子学报,2013,41(3):615-618.
- [7] 高建国,崔业勤. 基于信息熵理论的连续属性离散化方法[J]. 微电子学与计算机,2011,28(7):187-189.
- [8] 罗倩,宋鸿梅. 基于卡尔曼滤波和 AR 参数的非平稳杂波仿真[J]. 计算机工程与设计,2013,34(3):1092-1096.

作者简介:

王雪瑞 女,(1977-),硕士,讲师.研究方向为计算机应用与网络安全.

何永强 男,(1977-),硕士,副教授.研究方向为人工智能与计算机技术.