

FDTD 算法的网络并行运算实现

薛正辉, 杨仕明, 高本庆, 张泽杰

(北京理工大学电子工程系, 北京 100081)

摘 要: 本文提出了 FDTD 方法在微机互连构成的机群(COW)并行计算系统上以消息传递方式执行的一种实现方案. 组成了验证性的机群并行计算系统, 以一维和三维典型算例验证了算法的可行性、正确性和高效性, 为运用 FDTD 方法进行电大尺寸复杂电磁问题数值模拟计算提供了一条有效的途径, 也为深入进行这方面研究奠定了基础.

关键词: 时域有限差分法; 机群系统; 并行虚拟机; 并行计算

中图分类号: O441. 4 TP338. 6 **文献标识码:** A **文章编号:** 0372-2112 (2003) 12-1839-05

A Parallel Implementation Strategy for the FDTD Algorithm

XUE Zheng-hui, YANG Shi-ming, GAO Ben-qing, ZHANG Ze-jie

(Beijing Institute of Technology, Beijing 100081, China)

Abstract: This paper introduced a strategy for parallel implementation of the FDTD algorithm used in COW(Cluster of Workstation)parallel computing system and PVM parallel environment. Because of the one-dimension space division, this method is efficient relatively highly efficient. Some computed examples were given to prove the feasibility and correctness of this method. This method provides a solve-scheme for the EM computation of electrically-large-size complex object and can be used in radiation, scattering and EMC analysis.

Key words: FDTD; COW; PVM; parallel computation

1 引言

时域有限差分法(Finite-Difference Time-Domain Method)是 1966 年由 K. S. Yee 首次提出的. 它的基本思想是基于直接根据时域麦克斯韦方程的场分量的微分式, 用差分替代微分式实施各场分量的迭代而建立的, 其应用领域已从早期的电磁散射, 推广到生物电磁学、微波电路分析设计、电磁兼容预测、电波传播与天线辐射问题等, 是一种较为先进的数值分析方法^[1, 2].

但是, 虽然 FDTD 方法是一种电磁学领域较为优秀和非常有实用意义的数值算法, 但它的实现和优越性的发挥也受到来自计算平台的极大限制, 其算法稳定性要求空间网格的最大尺寸应小于波长的十分之一, 计算时间又正比与网格总数, 这使得在当前的 PC 机计算能力下, FDTD 法仅在低频范围获得了较多应用. 例如, 在分析一个最大尺度为 15 个波长的电大尺寸物体的电磁散射特性时, FDTD 计算需要至少 $200 \times 200 \times 200 = 800$ 万个网格点, 总的变量个数可高达千万量级, 目前虽然单处理机的性能也在飞速发展, 计算速度和存储容量也在不断提高, 但由于工艺水平和其它因素的限制, 单处理机的性能还是不能满足前述的应用要求, 这一点是显而易见的, 而且依赖今后单机性能的大幅度提高几乎也是不可能的.

因此, 为了适应不断发展的现实应用问题, 构造并行计算机来满足计算需求成为必然. 目前国际上已经有研究者进行了 FDTD 并行计算的研究和实践^[7, 11, 14~17], 获得了宝贵的经验, 也提出了多种实现的思路, 目前还处于蓬勃发展之中.

FDTD 算法本身具有适合于开展并行运算的优势, 由于 FDTD 方法中每一网格点的电磁场值只与紧邻的网格点上电磁场值及本网格点上一时间步的电磁场值有关, 而与远端网格点的场值无直接关系, 因此适合于用空间分区、时间分段的方式来分配运算任务, 这一点与并行运算的思路是统一的. 本文提出的计算方案是在 PC 机通过以太网络连接组成的机群系统上, 以 Linux 操作系统和 PVM 软件为基本平台, 以 FDTD 为基本算法而构造的消息传递型并行计算系统, 在该系统上以一维和三维典型算例验证了算法的可行性、正确性和高效性, 为运用 FDTD 方法进行电大尺寸复杂电磁问题数值模拟计算提供了一条有效的途径, 也为深入进行这方面研究奠定了基础.

2 FDTD 并行计算环境

2.1 系统硬件环境

在我们实验室中组建的并行计算系统是一个典型的机群系统, 即 COW(Cluster of Workstation)系统. COW 在实用上具有

收稿日期: 2002-08-15; 修回日期: 2003-06-04

基金项目: 国家重点自然科学基金(No. 69931030); 高等学校博士点基金

一些明显的优点^[4],如投资风险小,编程方便,系统结构灵活,性能/价格比高,能充分利用分散的计算资源以及可扩充性好等.尤其对于绝大多数电磁场分析计算的场合而言,巨型机(如国产银河系列系统、曙光系列系统、神威系列系统等)一般是可望而不可及的,从基本工作条件出发充分利用已经具备的 PC 机和网络资源才是现实的.这正是选用 COW 平台开展本研究的基本出发点所在.

COW 系统由工作站和互连网络两部分组成^[4].工作站是个广义的称呼,它可以是高档微机,甚至也可以是个对称多处理机 SMP.互连网络可以是普通的 LAN(如以太网、令牌环和 FDDI 等),也可以是高速开关网络(如 ATM、交换式高速以太网等).从程序设计模式的角度看,它可采用面向消息传递的 SPMD(Single Program Multiple Data)编程方式,即各个工作站均运行同一个程序,但分别加载不同的数据,从而可支持粗粒度的并行应用程序.图 1 是我们组建的实验并行计算系统的简单示意图.工作站采用各种性能的 PC 机,互连网络采用以太网(Ethernet)技术,传输速率 10M/s;网络拓扑结构采用星型结构,以交换机(或集线器-Hub)为中心.最简化配置时,整个系统可仅由两台 PC 机和 HUB 组成.

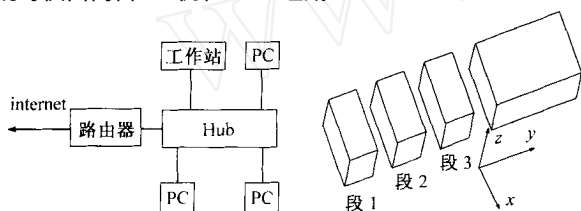


图 1 实验并行计算
系统示意图

图 2 计算空间
分割情况

2.2 系统软件环境

系统软件环境由两部分组成:操作系统及并行消息传递软件.操作系统采用基于 UNIX 的 Linux 操作系统,并行消息传递软件选用了虚拟并行机(PVM)软件.

2.2.1 Linux 操作系统 操作系统是软件系统的基础.目前大多数 PC 机安装的操作系统都是 WIN9X 系列操作系统,由于其对硬件要求较高,网络操作能力有限,对于大部分的并行计算平台不能提供有效的支持,所以目前还不能选择它作为机群系统的操作系统.

Linux 系统的内核最初由芬兰人 Linus Torvalds 自行开发而成,它是一个 32 位的操作系统,是 UNIX 在 IBM PC 兼容机上的完整实现.Linux 系统对硬件要求很低,配置 80386SX 以上档次 CPU 并且具有 2MB 以上内存的 PC 兼容机均可安装运行,它还是一个免费软件,它的内核完全自行开发而成,系统程序也全部由免费软件构成,因此便于推广.其网络功能强大,许多网络并行计算平台如 PVM 等均可在 Linux 系统上运行^[3].这些是我们选择 Linux 作为微机上的网络并行操作系统的主要原因.关于其细节可参看相关参考资料^[8-10].

2.2.2 虚拟并行机 PVM PVM(Parallel Virtual Machine)是美国国家基金会资助的公开软件系统,由美国 Oak Ridge 国家实验室研制.它具有通用性强及系统规模小的特点,适合于

TCP/IP 网络环境,当前所有的并行机公司都已宣布支持 PVM3^[3].

PVM 采用库函数法扩展 Fortran 和 C 语言来完成消息传递模型的并行编程.关于 PVM 的详细信息可参看参考文献 [3]和[6],这里不在赘述.

3 FDTD 并行算法设计

关于 FDTD 方法,已经有相当多的文献资料和专门书籍详细介绍其基本原理、编程过程及其它技术细节,这里不再赘述,可参看文献[1]、[2].进行 FDTD 并行算法设计的前提首先是对 FDTD 计算任务进行空间和时间上的合理分割,以适应不同计算节点加载不同计算任务.从 FDTD 计算本身模拟电磁波在时间上由前至后过程的特点及人们理解客观现象发生顺序的思维定势角度看,首要的还是计算任务的空间分割——即空间并行,其次才是不同计算任务之间时间的完全并行化.按照这一原则,我们首先开发了空间分割的 FDTD 算法,再进行了时空并行算法设计的尝试.

3.1 空间分割的 FDTD 算法

关于 FDTD 算法中计算空间分割的可行性已经有文献加以论述并受到认可^[11,7].所谓‘空间分割的 FDTD 算法’的核心思想是利用单 PC 机的硬盘空间模拟内存空间,通过共用数组在应用程序级别实现内外存交换.以计算空间一维分割为例,在实施这种方法时复杂形体的三维 FDTD 计算空间沿坐标轴向被分割成许多小段,分别赋给标号 1, 2, 3, ..., n, 其中每一小段的空间大小小于现有微机内存容量的允许范围,如图 2 所示,每一小段按顺序分别加载内存,实现计算机内存的复用,以弥补 PC 机完成整个计算任务内存的不足.

其执行过程以及关于其验证算例及优缺点的讨论可见参考文献[12].

3.2 机群系统上的 FDTD 并行算法

空间分割 FDTD 算法的开发为在真正的并行机上实现 FDTD 的时空并行运算奠定了基础,也提供了思路:如果不同段能分别装载于并行机中的各个计算节点,按一定的同步设置在每一时间步中各段独立叠代运算,通过网络消息传递交换分界面的数据,而不是一台 PC 机分次装载不同段的模拟并行方式,则这一空间分割算法中占用硬盘空间大和内外存之间交换数据造成的计算时间过长问题即可解决.也就是说,不仅能算而且算的好.按这一思路,我们开发了 COW 系统上基于 PVM 的 FDTD 时空并行算法.

3.2.1 计算空间分割及场量传递分析 为检验并行算法设计的可行性和正确性,首先进行的是一维 FDTD 运算的并行化工作.仅以计算任务分割为两部分为例,计算问题是:在 y 向长度为 100 个网格的一维自由空间,第 90 网格处存在一无限大良导体反射平板,考察位于

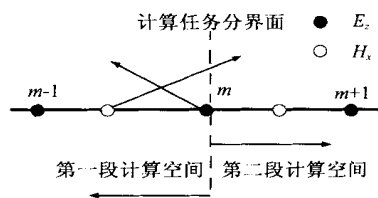


图 3 一维计算空间的分割及段间场量传递

$j = 12$ 相对幅度为 100

的高斯型平面波源激励的电磁波沿轴的传播情况。

设平面波场量为 H_x 和 E_z , 计算空间在 $j = m$ 处剖分为两段计算空间, 如图 3 所示。

一维情况下, H_x 和 E_z 的 FDID 差分格式分别为^[1,2]:

$$H_x^{n+\frac{1}{2}}\left(j+\frac{1}{2}\right) = C_x\left(j+\frac{1}{2}\right)\left\{-D_x\left(j+\frac{1}{2}\right)H_x^{n-\frac{1}{2}}\left(j+\frac{1}{2}\right) - \left[E_z^n(j+1) - E_z^n(j)\right] / y\right\} \quad (1)$$

$$E_z^{n+1}(j) = C_z(j)\left\{-D_z(j)E_z^n(j) - \left[H_x^{n+\frac{1}{2}}\left(j+\frac{1}{2}\right) - H_x^{n+\frac{1}{2}}\left(j-\frac{1}{2}\right)\right] / y\right\} \quad (2)$$

可以看到, 计算 $n + \frac{1}{2}$ 时间步 $j = m - \frac{1}{2}$ 处的磁场量 $H_x^{n+\frac{1}{2}}\left(m - \frac{1}{2}\right)$, 除需要本处上一个时间步的磁场量 $H_x^{n-\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 外, 还需要上半个时间步左右相邻的两个电场量 $E_z^n(m)$ 和 $E_z^n(m-1)$ 。如果 $H_x^{n+\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 在第一个计算节点内计算, 如图 3 所示, 则 $H_x^{n-\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 和 $E_z^n(m-1)$ 是存储在第一个计算节点的本地内存中的, 在差分计算中可直接引用; 但是 $E_z^n(m)$ 却并不存储在第一个计算节点的本地内存中, 它的计算是在第二个计算节点中完成的, 因而是存储在第二个计算节点的本地内存中的。这样, 在计算 $H_x^{n+\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 之前, 必须完成 $E_z^n(m)$ 从第二个计算节点到第一个计算节点间的传递, 每个时间步都是如此。同样道理, 在计算 $E_z^{n+1}(m)$ 之前, 必须完成 $H_x^{n+\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 从第一个计算节点到第二个计算节点间的传递。这两个步骤是循环互相依靠, 相辅相成的。这种段间场量数据传递也示意在图 3 中, 图中箭头所指表示数据传递方向。

3.2.2 计算时空并行化处理 依据上面描述的空间域分解, 可以实现各个计算节点 FDID 运算的完全并行化, 但是基于 FDID 计算程序‘空间扫描’计算的原理, 这种并行是节点间存在‘时间错位’的并行化。以图 3 中计算空间分成两部分为例, 设空间扫描从第一段计算空间 $j = 0$ 开始, 在计算的第 $n + 1$ 个时间步, 第一计算节点开始计算, 第二计算节点必须等待第一计算节点计算空间电磁场扫描所得交界面左侧数据 $H_x^{n+\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 到来(至于计算第一段 $H_x^{n+\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 需要的第二段 $E_z^n(m)$, 是上一时间步从第二计算节点传过来的, 下面将会看到), 待接收数据后方能开始这部分空间第 $n + 1$ 时间步内的计算, 首先计算交界面上第一计算节点所需数据 $E_z^{n+1}(m)$ 并即刻传递给第一计算节点, 随后顺序进行第二计算节点其余空间 $j = m + 1, j = m + 2, \dots$ 的扫描计算; 与此同时, 第一计算节点已经可以开始第 $n + 2$ 个时间步的扫描, 扫描到达空间右端时需接收第二段刚刚算完的 $E_z^{n+1}(m)$ 以完成 $H_x^{n+\frac{1}{2}}\left(m - \frac{1}{2}\right)$ 的计算并传给第二段, 于是第二计算节点开始计算 $n + 2$ 时间步同时第一计算节点开始计算 $n + 3$ 时间

步, …… 这样的过程一直重复到预定时间步完成为止, 每当第一计算节点开始 $n + 1$ 时间步的计算时, 第二计算节点开始第 n 时间步的计算, 每一时间步内计算节点之间交换一次数据。这样, 两个计算节点在空间扫描上是‘接力’的, 在时间上是首尾相接并错开的, 把原来串行算法整个空间扫描的一个‘波浪’拆分成了各个计算节点都存在空间扫描的多个‘波浪’, 显然在所需计算内存和计算时间方面大大减少。当子程序划分大于 2 时, 其过程完全是类似的, 其流程可以用图 4 表示。

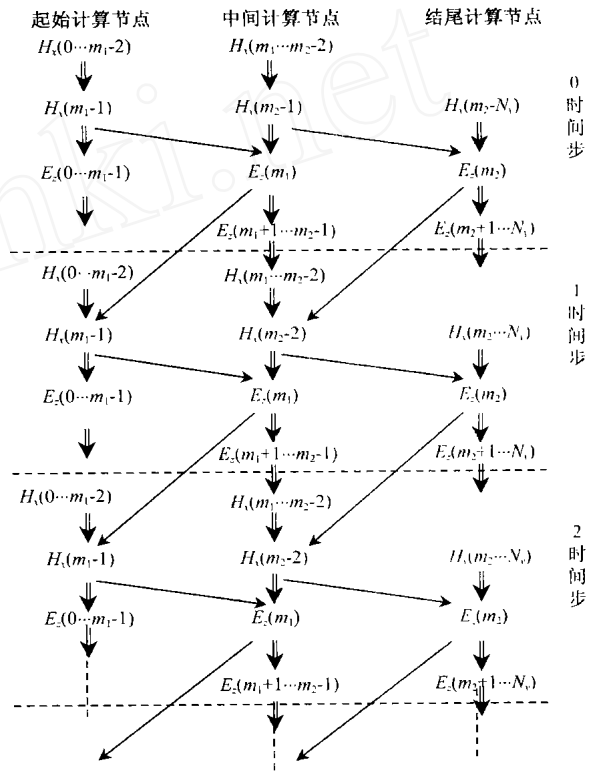


图 4 时空完全并行 FDID 算法示意图

该图表示了计算空间分为三部分时计算过程, 图中 m_1 和 m_2 分别为第一、二及第二、三段分割点网格标号, N_y 为一维计算空间总长度, 符号 \Rightarrow 表示正常的 FDID 计算过程, 而符号 \rightarrow 表示计算节点间的数据传递。

三维问题的处理过程与一维情况是完全类似的。最简单的情况是按照一维划分计算空间的办法分割二维或三维计算空间, 与一维情况的区别仅在于一维计算空间分界是‘点’, 二维是‘线’, 而三维是‘面’, 需要在节点间传递的是整个分界‘线’或分界‘面’上的所有数据。

3.2.3 与并行化相关的其它问题 除了以上介绍的并行 FDID 算法的核心思想外, 算法设计还包含其它外围问题处理的设计和流程, 这些 FDID 外围相关问题包括: 计算模型的分割, 吸收边界的分割, 连接边界的分割, 近远场变换积分面的分割及各节点计算任务平衡问题等。

(1) 计算模型的分割

适应于各个计算节点分别承担一部分计算任务, 原来串行 FDID 计算中待分析目标形体的整个模型必将要分割成与

各计算节点对应的各部分. 这一点一般来说并不困难, 因为 FDID 计算模型本身就是离散化的数字模型, 按照要求的划分方式‘肢解’计算模型即可. 图 5 表示了待分析的一实心圆球的模型分割, 分割数是 3 段.

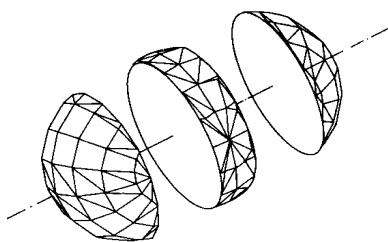


图 5 FDID 计算模型的并行化分割

(2) 吸收边界的分割

在一维问题中, 由于两维无限大, 吸收边界仅存在于计算空间的两头, 按照图 3 所示的分割, 仅需计算 $E_z(j=0)$ 和 $E_z(j=100)$. 这两个场量的计算并不受到计算空间分段的影响, 只要在第一计算节点和第二计算节点分别计算即可.

当问题扩展到二维, 边界条件需施加于计算面的四边, 若仍按一维分割, 则垂直于分割线的两条边上边界场值计算受到影响. 当问题扩展到三维, 边界条件需施加于计算体积的 6 个面和 12 个棱边, 若仍按一维分割, 则与分割面正交的 4 个面及 4 个棱边上边界场值计算受到影响. 按照前面介绍的并行原则, 要完成分割面左右的边界场量计算, 必须进行相应的段间数据传递, 其考虑思路与计算空间内部的数据传递类似, 但须从边界条件的具体表达式出发进行细致的分析, 这里不再赘述了.

(3) 连接边界的分割

在应用并行 FDID 算法完成散射问题计算时, 需涉及到连接边界的分割. 由于连接边界的实质是根据连接边界面上场源的已知数值和总场值求出外向场的数值, 不涉及差分计算, 故不需要段间的数据传递, 只需根据分段情况实现求出各段内连接边界面上的场源数值即可, 还须注意各段内连接边界面上场源的时延关系不要被分段割裂.

(4) 近远场变换积分面的分割

由于近远场变换需计算积分面表面电流密度 $\vec{j}_s = \hat{n} \times \vec{H}_s$ 与磁流密度 $\vec{M}_s = -\hat{n} \times \vec{E}_s$ 的面积分, 因此在计算分界面邻近处的积分时需进行段间数据传递, 其原理与处理吸收边界类似, 需根据积分式等效的离散求和公式去分析需传递的场量. 当各段的积分分别完成后, 再把各段的积分结果求和即可求出总的散射远场.

(5) 计算任务平衡问题

各个计算节点承担的计算任务规模是相同的, 也可以是不同的. 一般情况下, 根据 COW 各节点的性能不同应该承担不同规模的工作, 计算速度快及内存大者应承担较大任务量, 反之则承担较小任务量, 这样可以避免‘快’节点等待‘慢’节点同步而造成的计算时间浪费. 这种平衡可以通过改变分割面完成, 需根据组成 COW 的各节点机具体情况而定. 由于是一次性的工作, 这种平衡可以手工完成也可以自动完成.

4 算例检验

依据上面描述的空间域分解, 我们选择了‘消息传递’编

程模型和‘主-从’式的并行编程风范. 编制的主-从式并行程序包括三个阶段: 第一阶段, 由主任务产生子任务群, 并且分发任务信息和问题的一些参量; 第二阶段, 执行计算, 在计算中子任务之间保持同步和通信; 第三阶段, 主任务从子任务收集结果并显示结果, 然后终止所有的子任务. 在程序编制中, 两个计算节点的子程序实际上是完全一致的, 只是在子程序中设置变量, 根据主程序任务分配号去识别自己承担的是那部分计算任务. 程序中, 数据传递由 PVM 库函数承担, 关于实现并行 FDID 计算所需的 PVM 库函数定义及各变量含义可参看参考文献[3]等, 这里不再详述了.

作者用于检验算法的实验并行计算系统取最简化配置, 整个系统由两台 PC 机和 Hub 组成, 两台 PC 机分别为 P 266/64M 和 P 200/64M, 网卡为 PCI 10/100M 快速以太网适配器, Hub 选用 TP-LINK 10BASE-T.

关于选择何种算例(或计算对象)来检验 FDID 并行算法及相应程序的问题, 按照检验‘并行算法’的一般原则, 仅需确保计算模型与串行运算的一致性, 检验的内容应该是: 在一致的计算模型上, 并行算法是否取得与串行计算完全相同的计算结果, 相对于串行算法的加速比和并行效率如何, 并不需要检验计算模型和计算结果的正确性和合理性, 也不要求它们能反映真实情况等, 甚至不需要针对实际计算任务, 仅仅虚拟、假想模型即可. 按照这样的思路进行了大量的检验计算, 这里仅给出两个实例: 一维的 TEM 波传播计算以及作为三维问题检验的某飞行器远场时域散射计算.

4.1 一维问题

以 100 个网格的 TEM 波传播问题为例, 分别在第 15、30、60 和 80 网格处记录通过的波形. 在第 12 网格处设置高斯脉冲激励源, 在第 90 个网格设置金属反射板, 使得从第一计算节点到第二计算节点正向和反向波的传播都得到了检验, 其中在 30 网格处记录的波形如图 6 所示, 图中同时绘出了串行和空间分割 FDID 计算同一算例所得结果.

由图中可见计算是成功的, 这种并行计算思路完全能够得到与串行计算同样的准确度, 表明整个并行计算系统是可行的, 并行算法设计是科学的. 在计算中观察到, 当计算空间较

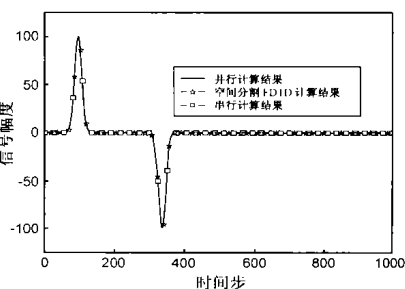


图 6 第 30 个网格处记录波形

小(以 100 个网格为例), 由于计算节点之间通信的时间开销要大于由于计算节点减小了计算空间而节省的计算时间, 在 100 个网格的算例中, 并行计算所需时间大于串行计算所需时间, 本算例仅是检验了这一并行计算思路的可行性, 还看不到其节省内存和计算时间的优势. 当计算空间逐步扩大, 这一优势才被逐步显示出来, 计算证明: 当计算空间为 100000 个网格时, 并行加速比 S 约为 1.6, 并行效率 E 约为 0.8, 可以看出这种并行思路的潜力还是巨大的.

4.2 三维问题

三维散射问题计算需要考虑上节所介绍的各种因素,因此并行运算程序更加相对复杂化.散射并行运算的算例选择某飞行器的平面波散射问题,计算空间为 $76 \times 76 \times 76$,受到机头轴线方向 V-V 极化平面波照射,记录经近远场变换后后向散射的 E 和 E 时域数据, E 结果示于图 7,图中同时绘出串行计算同一问题所得结果.这里不讨论建模及波源施加问题,只考虑‘并’、‘串’目标一致.

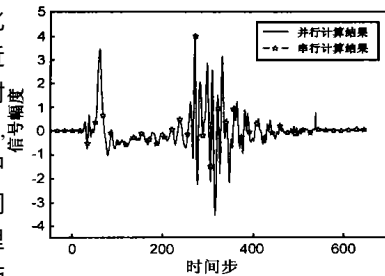


图 7 远场散射 E 分量

由图中可见散射计算并行化也是成功的,连接边界分割与组合是正确的.计算统计显示加速比约为 1.3~1.4,效率约为 0.65~0.7.与一维问题相比加速比有所下降,主要原因在于需在段间传递的数据大大增加而导致了通信开销加大.

5 结论

从以上三个方面的计算检验可见,FDTD 运算的并行化是可行的,并行计算程序编制是较成功的,所得加速比与效率等并行评价参量与资料显示基本相当^[14],这证明这种 FDTD 并行思路已经基本可以投入实际运用去解决工程问题.

最后还必须指出的是在以上并行算法设计中,尚未进行算法和程序的最优化工作,因此计算效率方面应该还有进一步提高的潜力.同时,这里开发的这一 FDTD 并行思路仅是众多并行化方案的一种,而且也不见得是最优的一种,同样依据‘消息传递’模型,还可能有许多编程风范和格式也是完全可行的,甚至不遵从‘消息传递’模型也可以导出全新的 FDTD 并行化方案,这一点从并行算法的介绍中已经可以联想到:因为决定并行算法的诸因素实在是太多了.作者的工作并没有也不可能穷尽这方面的新思路,发展的空间还是相当大的,这也正是作者下一步工作的方向之一.

参考文献:

- [1] 高本庆.时域有限差分法 FDTD Method[M].北京:国防工业出版社,1995.
- [2] 王长青,祝西里.电磁场计算中的有限差分法[M].北京:北京大学出版社,1994.
- [3] 孙家昶,张林波,迟学斌,汪道柳.网络并行计算与分布式编程环境[M].北京:科学出版社,1996.
- [4] 陈国良.并行计算—结构 算法 编程[M].北京:高等教育出版社,1999.
- [5] 张宝琳,谷同祥,莫则尧.数值并行计算原理与方法[M].北京:国防工业出版社,1999.

- [6] A Geist, A Beguelin, J Dongarra, W C Jiang, R Manchek, V Sunderam. PVM: Parallel Virtual Machine[M]. USA: The MIT Press, 1994.
- [7] Amir Fijany, Michael A Jensen, Yahya Rahmat-Samii, Jacob Barhen. A massively Parallel Computation Strategy for FDTD: Time and Space Parallelism Applied to Electromagnetic Problems[J]. 1995, IEEE Trans on AP, 1441 - 1449.
- [8] 陈力锋. Red Hat Linux 6.0 网络与通信专辑[M].北京:人民邮电出版社,2000.
- [9] 陈军,龙浩. Red Hat Linux 7 自学指南[M].北京:国防工业出版社,2001.
- [10] 周巍松,等. Linux 系统分析与高级编程技术[M].北京:机械工业出版社,1999.
- [11] 马积福.基于 FDTD 方法的谐振结构三维电磁场的同意分析及应用[D].北京:中国科学院电子学研究所,1998.
- [12] 薛正辉,高本庆.空间分割的 FDTD 方法[A].微波电磁兼容第四届全国学术会议论文集[C].中国:微波电磁兼容学会,1998.
- [13] 薛正辉.电磁场的时域技术及应用[D].北京:北京理工大学,2002.
- [14] Gillan C J, Fusco V. Enhanced distributed computing message passing strategies for FDTD[J]. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 1999, 12(6): 483 - 488.
- [15] Rodohan D P, Saunders S R, Gover R J. Distributed implementation of the finite difference time-domain (FDTD) method [J]. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 1995, 8(3 - 4): 283 - 291.
- [16] Davidson David B, Zolkowski, Richard W. Connection machine (CM-2) implementation of a three-dimensional parallel finite difference time-domain code for electromagnetic field simulation[J]. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 1995, 8(3 - 4): 221 - 232.
- [17] Gillan C J, Fusco V F. Optimizing FDTD electromagnetic field computation on distributed networks[J]. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 1998, 11(6): 277 - 287.
- [18] Varadarajan V, Mitra Raj. Finite-difference time-domain (FDTD) analysis using distributed computing [J]. IEEE Microwave and Guided Wave Letters, 1994, 4(5): 144 - 145.

作者简介:



薛正辉 男,1970 年生于山西,现为北京理工大学电子工程系副教授,1992 年毕业于北京理工大学电子工程系,1995 年在同校电磁场与微波技术专业获工学硕士学位并留校任教至今,2002 年获得北京理工大学电磁场与微波技术专业博士学位,主要从事电磁场与微波技术方面的教学和计算电磁学、电磁兼容及微波毫米波技术方面的有关研究,发表论文 20 余篇.