# 基于 GPU 平台的联合迭代重构算法加速

**张广勇<sup>1</sup>** 卢晓伟<sup>1</sup> 沈 铂<sup>1</sup> 张 凯<sup>2</sup> 储 琪<sup>3</sup> 赵开勇<sup>1,4</sup> (浪潮集团高效能服务器和存储技术国家重点实验室 济南 250101)<sup>1</sup> (中国科学院生物物理研究所生物大分子国家重点实验室 北京 100101)<sup>2</sup> (中国科学院计算技术研究所前瞻研究实验室 北京 100190)<sup>3</sup> (香港浸会大学计算机科学系 香港)<sup>4</sup>

摘 要 电子断层三维重构技术(Electron Tomography, ET)是在纳米尺度下研究不具有全同性的细胞或大分子三维 结构的重要方法。迭代重构法是 ET 中重构效果最好的方法,但是其性能较差,重构大尺寸图像时需要数天的时间甚 至更长,使其应用受到限制。迭代重构法中经典的方法是代数重构法(Algebraic Reconstruction Technique, ART)和 联合迭代重构法(Simultaneous Iterative Reconstruction Technique, SIRT), SIRT 算法总是收敛的并且比 ART 重构的 效果更好。利用 CUDA 语言设计和实现了基于 Tesla C1060 GPU 平台上的并行 SIRT 重构算法,并利用存储器合并 访问、常量存储器、共享存储器等优化技术对并行算法进行优化,优化后的 SIRT 并行算法在 Tesla C1060 GPU 平台 上的最大加速比是 Intel i7 920 CPU 上的串行算法的 47 倍,并且重构的质量没有任何下降。 关键词 电子断层三维重构,联合迭代重构法,并行计算,GPU,CUDA,ATOM

中图法分类号 TP391 文献标识码 A

Design and Optimization of Simultaneous Iterative Reconstruction Technique Based on GPU Platform

ZHANG Guang-yong<sup>1</sup> LU Xiao-wei<sup>1</sup> SHEN Bo<sup>1</sup> ZHANG Kai<sup>2</sup> CHU Qi<sup>3</sup> ZHAO Kai-yong<sup>1,4</sup>
(National Key Laboratory for High-efficient Server and Storage Technology, Inspur, Jinan 250101, China)<sup>1</sup>
(National Laboratory of Biomacromolecules, Institute of Biophysics, Chinese Academy of Sciences, Beijing 100101, China)<sup>2</sup>
(Advanced Computer Research Center, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)<sup>3</sup>
(Department of Computer Science, Hong Kong Baptist University, Hong Kong, China)<sup>4</sup>

**Abstract** Electron tomography (ET) is widely used in reconstructing non-uniform cells or macromolecules in nano scale. One of the best methods of ET is iterative reconstruction due to its outstanding quality of reconstruction, but it is limited by its huge computational requirements. A parallel simultaneous iterative reconstruction technique(SIRT) was designed and implemented based on GPU platform with Tesla C1060 using CUDA programming languages. Experimental results demonstrate the performance of optimized parallel SIRT algorithm. The maximum speedup of the parall el SIRT is 47 times of sequential SIRT approach, and it is not any loss of accuracy.

Keywords Electron tomography, SIRT, Parallel computing, GPU, CUDA, ATOM

# 1 引言

电子断层三维重构技术是研究细胞器、亚细胞组装体甚 至整个细胞,以及病毒、蛋白质大分子等三维结构的重要方 法<sup>[1]</sup>。1968年,DeRosier在文献[2]中提出了电子显微三维 重构技术的概念,其基本原理基于中央截面定理:三维物体沿 电子束方向投影的傅立叶变换是该物体所对应的傅立叶空间 中通过中心且垂直于投影方向的一个截面。电子断层三维重 构技术中最常用到的重构算法有傅里叶重构、加权背投影重 构(WBP)和代数重构。迭代重构法中经典的方法是 R. Gorden<sup>[3]</sup>等提出的代数重构法(Algebraic Reconstruction Technique,ART)和 P. Gilbert<sup>[4]</sup>提出的联合迭代重构法(Simultaneous Iterative Reconstruction Technique,SIRT)。目前,应用 最广泛的是 WBP,但是,有限的角度数据造成的缺陷严重影 响了 WBP 的重构效果<sup>[5]</sup>。迭代重构法很好地解决了 WBP 的缺陷。然而,由于迭代重构法处理速度较慢,它没有得到更 广泛的应用。

为了提高迭代重构法的性能,使其得到更广泛的应用,在 串行平台和传统并行平台上出现了对迭代重构法的众多改进 版本,如联合代数重建法<sup>[6]</sup>、改进的同步迭代图像重建技术<sup>[7]</sup> 和 AUTO3DEM<sup>[8]</sup>、UCSF TOMOgraphy<sup>[9]</sup>等。这些改进在 性能上有所提高,但幅度较小,仍不能根本上解决迭代重构法

#### 到稿日期:2011-07-21 返修日期:2011-09-10

张广勇(1985一),男,硕士,工程师,主要研究领域为高性能并行计算,E-mail;zhangguangy@inspur.com;卢晓伟(1984一),男,硕士,工程师,主要研究领域为高性能运算与数据流挖掘;沈 铂(1986一),男,工程师,主要研究领域为高性能运算;赵开勇(1982一),男,硕士,主要研究领域为高性能异构计算、生物信息学。

## 的重构过程漫长的问题。

近年来,以 NVIDIA Tesla<sup>10]</sup>为代表的异构芯片(协处理 器)逐渐被引入到通用计算领域中。最初的图形处理(Graphic Processing Unit,GPU)就是用来处理大规模的并行计算, 并且 GPU 的并行处理能力在不断地上升。本文基于 Tesla C1060 GPU 平台,采用 CUDA 编程语言设计并实现了 SIRT 并行算法。首先实现了最基本的满足合并访问的 SIRT 并行 算法,然后采用常量存储器对并行算法进行优化,以进一步提 高性能。本文实现的并行 SIRT 算法在 Tesla C1060 GPU 平 台上的最大加速比是 Intel i7 920 CPU 上的串行算法的 47 倍,并且重构的质量没有任何下降。

本文第2节简介迭代重构法;第3节给出基于 GPU 平台 的并行迭代重构法的设计和实现;第4节为实验环境和实验 结果分析;最后给出结论和进一步的工作。

# 2 迭代重构法

## 2.1 电子断层三维重构技术

电子断层三维重构技术(Electron Tomography,ET)是从 一个物体的二维投影图经过重构获取物体内部结构的技术, 其通过获取多个不同角度的二维投影图来反向重构出所研究 对象的三维结构。对于复杂生物样品的 ET 成像,是用单轴 倾斜来进行投影数据的获取,电子束从上到下透过样品打到 投影图上,然后通过旋转样品获取不同角度的投影图。最常 见的倾斜系列范围是在±60°~±70°之间,采用1°~2°间隔的 线性步长,从而获得在不同倾斜角下的序列投影图像。一般 ET 的投影图像的数量为 60~280 张。由于分辨率的要求,投 影图像大小为 1024×1024 到 2048×2048 像素。

## 2.2 迭代重构法

# 2.2.1 断层截面与投影的几何表示

在三维重构中,我们每次以切片为单位进行重构,断层截 面与投影的几何表示如式(1)所示,x<sub>i</sub>表示断层截面第<sub>j</sub>个 像素的值,P<sub>i</sub>是第 i 根光线测量的射线和(光线的线积分和 称为射线和),A<sub>i</sub>称为加权因子,代表第 j 个像素对第 i 根光 线积分的贡献,大多数 A<sub>i</sub> 均为 0。

$$\sum_{i=1}^{N} A_{ij} x_j = P_i, i = 1, 2, 3 \cdots, M$$
(1)

式中,N为像素总数,M为光线总数,在实际的应用中,N值 往往很大,如果要重构 1024 \* 400(截面像素的大小)的图像, N=409600,并且 M的值与 N的值在同一个数量级上,因此, 任何对线性方程组的直接求法都变得无法实现。对于 M和 N 很大的情况,目前很多采用迭代法求解。

2.2.2 代数重构法 ART

1970年, R. Gorden 等提出了代数重构法, 它采用式(2)的 迭代方式,其中, k 表示迭代次数,  $A_i$  为 A 的第 i 行, 即  $A_i =$ ( $A_{i1}$ ,  $A_{i2}$ , ...,  $A_{iN}$ ),  $A_i \cdot A_i$  为  $A_i$  与其自身的内积,  $\lambda$  为松弛 因子。ART 算法每次只用一条光线的射线投影迭代更新数 据, 当某条射线包含误差时, 迭代所得到的解也会相应地引入 误差。

$$X^{(k)} = X^{(k-1)} - \lambda_k \frac{A_i \cdot X^{(k-1)} - P_i}{A_i \cdot A_i} A_i$$
(2)

2.2.3 联合迭代重构法 SIRT

SIRT 算法中每个像素的迭代求解是通过该像素的所有 射线和的差值的累加实现的,如式(3)所示,因此,重构的图像 对测量误差和噪声不敏感,在测量数据有误差时能够显示出 它在重构质量上的优越性。SIRT 算法比 ART 算法计算量 更大,因此,SIRT 算法以减慢收敛速度来换取比 ART 较好 的图像重构质量。大量实验数据表明,迭代重构法能够在数 据不完整的情况下得到较精确的重构图像,具有其他重构法 无法替代的优势。

$$X^{(k)} = X^{(k-1)} + \lambda_k \sum_{i=1}^{M} \frac{P_i - A_i \cdot X^{(k-1)}}{A_i \cdot A_i} A_i$$
(3)

# 3 基于 GPU 平台的并行 SIRT 算法

由于 SIRT 算法重构的图像对测量误差和噪声不敏感, 在测量数据有误差时能够显示出它在重构质量上的优越性, 并且 SIRT 算法总是收敛的,因此本文中主要分析 SIRT 算法 移植到 GPU 平台上的过程。

# 3.1 SIRT 串行算法

在本文中我们假设一个样品的长、宽、高分别为 SX,SY, 和 SZ,切片垂直于 SY 方向,即有 SY 个切片,每个切片的大 小为 SX \* SZ;假设旋转角度的总数为 ANGLE\_NUM,即有 ANGLE\_NUM 个二维投影图像,每个投影图像的大小为 SX \* SY。

在三维图像重构中,以切片为单位,每次重构一个切片, 把所有的切片都二维重构完毕之后即可得到三维重构图像。 重投影(ReProject)和背投影(BackProject)是 SIRT 算法中的 重点步骤,并且占用了重构的绝大部分时间。重投影是指根 据赋初值得到的  $X^{(0)}$ 或上次迭代得到的  $X^{(k-1)}$ 进行重新投影 得到一个投影图像值  $P_{\rm H}$ ;背投影是指根据赋初值得到的  $X^{(0)}$ 或上次迭代得到的  $X^{(k-1)}$ 以及测量得到的投影图像值  $P_{\rm M}$ 和重投影计算得到的投影图像值  $P_{\rm H}$ 进行反向投影得到 重构图像 X(k)。背投影主要是进行 X 值的更新操作,通过 多次的重投影和背投影操作可以得到满足精度要求的重构图 像。

## 3.2 P\_SIRT 算法

SIRT 算法主要的时间用于重投影和背投影的计算中。 在这两部分的计算中,数据完全满足 SIMD 的特性,因此,我 们并行的重点就是重投影和背投影算法。本文中重投影算法 的并行度为 SX \* ANGLE\_NUM, block 结构为(BLOCK-DIM,1,1), grid 的结构为 dim3((SX + BLOCKDIM-1)/ BLOCKDIM,ANGLE\_NUM,1), BLOCKDIM 大小为 64-256。内核如算法 1 所示,线程(n,i)负责  $P_{at}(n,i)$ 的计算。 背投影的并行度为 SX \* SZ, block 的结构为 dim3(BLOCK-DIM,1,1), grid 的结构为 dim3((SX + BLOCKDIM-1)/ BLOCKDIM,SZ,1)。内核如算法 2 所示,在内核中,每个线 程负责 X 中一个元素的计算。算法 1 和算法 2 可以很好地 满足全局存储器的合并访问。我们把整个 SIRT 并行算法记 为 P\_SIRT。

**算法1** P\_SIRT\_ReProject n← blockIdx, y //n 表示旋转角度序号 i ← blockIdx. x \* blockDim. x + threadIdx. x //i 为第 n 个旋转角度 投影图像中的光线序号

## 如果 x<SX //判断边界

- 根据旋转角度和样品的几何参数确定光线(n,i)经过的像素点(0 ~SZ,x)及 d\_A(0~SZ,x);
- 根据这些像素点 d\_X<sup>(k-1)</sup>(0~SZ, x)和 d\_A(0~SZ, x)计算 d\_P<sub>tt</sub>(n,i)
- 算法2 P\_SIRT\_BackProject

z ← blockIdx. y //z 为像素点在重构图像中的行号

- x←blockIdx. x \* blockDim. x+ threadIdx. x//x 为像素点在重构图像 中的列号
- 如果 x<SX//判断边界
  - for n←0:ANGLE\_NUM-1 //n 代表旋转角度序号
    - 根据旋转角度和样品的几何参数确定经过像素点(z,x)的光 线(n,i)以及 d\_A(z,x);
    - 根据 d\_A(z,x)、d\_P<sub>微</sub>(n,i) 与 d\_P<sub>H</sub>(n,i)的差值以及 d\_  $X^{(k-1)}(x,z)$ 更新 d\_X<sup>(k)</sup>(x,z);

## 3.3 P\_C\_SIRT 算法

ET 的投影图像的数量一般为 60~280 张,即 60~280 个 旋转角度。在重投影和背投影并行算法中,每个线程都要访 问其中的一个角度,同时这些旋转角度的值在重构过程中不 会有任何的改变。因此,我们可以把这些旋转角度存放到常 量存储器中。由于旋转角度的个数不能确定,我们在设备端 分配常量存储器空间时给一个较大的值(如 360),用于满足 各种情况,用 ANG\_MAX 表示,我们把该方法记为 P\_C\_ SIRT。代码如下:

\_\_constant\_\_ float c\_ang[ANG\_MAX]={0.0}; //定义全局变量 c\_ ang,在常量存储器上分配空间

cudaMemcpyToSymbol(c\_ang,ang,sizeof(float) \* ANG\_MAX); // 在主机端函数中进行数组的传递,ang 是主机端旋转角度数组

#### 4 实验方法和实验结果

#### 4.1 实验方法

在本文中,我们采用了浪潮"倚天"桌面超级计算机的硬 件平台,硬件环境的详细参数如表1所列。

表1 实验中使用的 CPU 和 GPU 硬件配置

CPU	Intel i7 Quad core 920 CPU 2, 66GHz, DDR3 1066Mhz 8GB
GPU	Tesla C1060 GDDR3 4GB, 240SPs 30SMs, 浮点计算能力: 933Gflops/s;存储器带宽:102GB/s 每个 SM 包含 16384 个寄存器,每个 SM 拥有 16kB 的共享存储器 和 64kB 的常量存储器 每个 SM 最多设置 1024 个 threads,计算能力:1.3

实验中,我们选取了 http://www.nic.st 网站上的两个 ST 格式的图像:cav.st 和 tset4a.st,两个 ST 图像的详细参数 如表 2 所列。

表 2 测试用例

ST 图像	大小(像素)	旋转角度	样品厚度(像素)	角度文件
cav	512 * 512	56	100	一60 度开始每隔 两度取一个角度
test4a	2048 * 2048	112	400	从一60 度开始每隔 一度取一个角度

## 4.2 P\_SIRT 算法与 SIRT 算法的性能比较

串行 SIRT 和并行 P\_SIRT 算法的测试结果如表 3 所列。

对于 cav 测试用例,迭代 1 次时的加速比为 20.22 倍,迭代 30 次时达到最好的加速比为 21.73 倍;对于 test4a 测试用例,迭代 1 次时的加速比为 12.57 倍,迭代 50 次时达到了 35.72 倍的加速比。

在本文中,我们实现的并行算法相对于原来的串行算法 没有任何质量上的下降。图1给出了串行算法 SIRT 和并行 算法 P\_SIRT 对 cav 测试用例进行测试得到的二维截图,我 们选取了具有代表性的最中间一个切片的截图。从该图可以 看出,GPU 得到的重构图像相对于串行算法没有任何质量的 下降。

表 3 SIRT 和 P\_SIRT 测试结果

数据	迭代次数	SIRT (s) (17 CPU)	P_SIRT(s) (Tesla C1060)	加速比
	1	75.23	3.72	20. 22
	10	500.43	23.55	21. 25
cav	20	970.12	45.84	21.16
	30	1563.85	71.96	21.73
	50	2345.36	109.91	21.34
	1	10520.73	478.14	22, 00
	10	69887.10	2026.71	34.48
test4a	20	135850.33	3936.91	34.5
	30	207159.71	5896.21	35.13
	50	333740.33	9344, 13	35, 72



图 1 SIRT 和 P\_SIRT 对 cav 进行测试的结果截图

#### 4.3 使用常量存储后的性能

常量存储器是 CUDA 内核加速中的一个重要因素,使用 常量存储器后的 P\_C\_SIRT 并行运行时间和加速比如表 4 所 列。对于 cav 和 test4a 测试用例,P\_C\_SIRT\_F 的最高加速 比为 26.95(迭代 30 次时)和 47.86(迭代 50 次时)。为了进 一步说明常量存储器对性能的影响,我们对 P\_C\_SIRT 和 P\_ SIRT 的加速比做了比较,使用常量存储器后的性能平均提高 了 24%~34%。

	表 4	SIRT	和PC	SIRT	测试结界
--	-----	------	-----	------	------

-	数据	迭代次数	SIRT (s) (17 CPU)	P_SIRT(s) (Tesla C1060)	加速比	
-		1	75.23	3, 05	24.67	
		10	500.43	19.15	26.13	
	cav	20	970, 12	36.38	26.67	
		30	1563.85	58.03	26.95	
		50	2345, 36	87.23	26.89	
		1	10520,73	362.23	29.04	
		10	69887.10	1512.47	46.21	
	test4a	20	135850, 33	2916.23	46.58	
		30	207159.71	4367.56	47.43	
		50	333740.33	6973.23	47.86	

# 4.4 ATOM1.0

目前,电子断层三维重构软件有很多,但速度较慢和硬件 费用高昂等原因制约了这些软件的广泛应用。针对现有重构 软件的不足,由中科院生物物理所、计算技术研究所和浪潮集 团高效能服务器和存储技术国家重点实验室联合研发了 ATOM软件——基于 GPU 平台的电子断层三维重构软件, 并免费发布了 ATOM 1.0 版本<sup>[11]</sup>。本文所述基于 GPU 的

• 312 •

SIRT 算法即包含在 ATOM 软件中。同时, ATOM 软件还包 含串行 WBP×ART 和并行 ART×SIRT 重构算法。

**结束语**本文主要分析了 SIRT 算法移植到 GPU 平台上的过程,并通过 CUDA 中的合并访问、常量存储器、共享存储器等优化技术对并行算法进行了优化。并行迭代重构算法在 Tesla C1060 GPU 平台上相对于运行在 Intel i7 920 CPU 平台上的串行算法性能上最高提高了 47.86 倍,极大地提高了重构的性能,且没有任何重构质量上的下降。在串行平台上重构 2048 \* 2048 像素的图像时需要数天的时间,而在我们开发的并行算法平台上仅需要不到 3 个小时的时间。同时,本文简单介绍了国内首款免费的电子断层三维重构软件-ATOM 1.0,其使用简单、性能高、成本低、移植性强。

本文所完成的工作还有进一步优化的空间,还有待做更 深入的研究和进一步的完善:SIRT 算法是按切片进行重构 的,片与片之间的数据也没有任何依赖性,因此可以在 GPU 集群上采用 MPI 和 CUDA 相结合的技术,由 MPI 完成片与 片之间的并行,由每个节点上的 GPU 完成片内的并行,通过 GPU 集群可以使重构速度更快,争取在数分钟内完成大尺寸 图像的重构。ATOM 1.0 软件的功能还需要进一步的完善, 如添加更多的并行重构算法(如并行加权背投影重构算法) 等。

## 参考文献

- [1] 张凯,张艳,胡仲军,等.电子显微镜三维重构技术发展与前沿[J].生物物理学报,2010,26(7);533-559
- [2] Derosizer D J, Kakg A, Reconstruction of Three Dimensional Structures form Electron Micrographs[J]. Nature, 1968, 217:

(上接第 303 页)

两者吻合很好。通过对图 4、图 5 超声速绕流状态( $Kn_{\infty}$  = 0.03, $M_{\infty}$  = 3.83)并行计算结果、有关实验数据的理论分析与 比较确认,证实了本文并行计算结果正确可靠。

结束语 本文通过研究区域分解并行计算模式,引入输 入/输出、通信与 CACHE 优化策略,对求解 Boltzmann 模型 方程的三维绕流气体运动论数值模拟程序进行 MPI 并行化 移植与高性能计算调试,以高空稀薄过渡流区飞行器绕流状 态为算例,进行了 MPI 大规模并行计算测试与并行计算研 究,证实了本文所发展的 MPI 并行化区域分解策略及程序优 化途径的正确性。研究表明本文开展的并行化实现能明显地 缩短模式计算时间,取得了较好的效果。同时看出,每个进程 上的负载取决于区域分解时分配的网格大小。当结点数较少 时,由于每个进程所分配的分割区域较大,这时负载不平衡及 通信带来的影响相对而言不明显;而当结点数较多时,每个进 程所分配的分割区域越来越小,这时负载不平衡及通信带来 的影响是较明显的。由于选取的模式网格空间的大小一定, 随着结点数的增加,每个进程所负担的计算量不断减少,通信 所占用的时间就显得较为重要。

本文所发展的 MPI 并行调试方法与程序优化工具在求 解 Boltzmann 模型方程的三维绕流气体运动论数值模拟程序 130-134

- [3] Gordon R, Bender R, Herman G T. Algebraic Reconstruction Techniques (ART) for three Dimensional Electron Micoscopy and X-ray Photography [J]. Journal of Theoretical Biology, 1970,29(3):471-481
- [4] Gilbert P. Iterative Methods for the Three-Dimensional Reconstruction of an Object from Projection[J], Journal of Theoretical Biology, 1972, 36(1):105-107
- [5] Herman G T. Fundamentals of Computerized Tomography: Image Reconstruction from Projection(2nd edition)[M]. Springer, 2009
- [6] Andersen A H, Kak A C. Simultaneous Algebraic Reconstruction Technique (SART): A Superior Implementation of the ART Algorithm[J]. Ultrason Imaging, 1984, 6(1):81-94
- [7] 向良忠,邢达,谷怀民,等.改进的同步迭代算法在光声血管成像 中的应用[J].物理学报,2007,56(7);3911-3916
- [8] Yan X, Sinkorits R S, Baker TS, AUTO3DEM—an Automated and High Throughput Program for Image Reconstruction of Icosahodral Particles[J]. Journal of Structural Biology, 2007, 157 (1):73-82
- [9] Zheng S Q, Keszthelyi B, Branlund E, et al. UCSF Tomography: an Integrated Software Suite for Real-Time Electron Microscopic Tomographic Data Collection, Alignment, and Reconstruction[J]. Journal of Structural Biology, 2007, 157(1): 138-147
- [10] Lindholm E, Nickolls J, Oberman S, et al. NVIDIA Tesla: A Unified Graphics and Computing Architecture [J]. IEEE Micro, 2008,28(2):39-55
- [11] 储琪,张凯,万晓华,等. ATOM 1.0:基于 GPU 的电子断层重构 软件[J]. 生物物理学报,2011,27(3):231-241

的大规模并行计算的应用中,已取得了一定的成果,以后将进 一步努力,不断优化并行算法程序和 CACHE、内存使用,提 高计算速度,以满足不断发展的模式研究和实际计算需求。

## 参考文献

- [1] Wang F, et al. Completely Debugging Indeterminate MPI/PVM Programs [J]. Journal of Software, 2001, 12(3)
- [2] 王巍,方滨兴,张宏莉.并行调试中的若干关键问题[J].计算机 科学,2003,30(3):132
- [3] 李志辉,张涵信.稀薄流到连续流的气体运动论统一数值算法初步研究[J].空气动力学学,2000,18(3):255-263
- [4] 李志辉. 从稀薄流到连续流的气体运动论统一数值算法研究 [D]. 绵阳:中国空气动力研究与发展中心,2001
- [6] 李志辉,张涵信.基于模型方程的气体运动论统一算法研究[J]. 力学进展,2005,35(4):560-572
- [5] 李志辉,张涵信. 稀薄流到连续流的气体运动论统一算法研究[J]. 空气动力学学,2003,21(3),256-264
- [7] 李志辉,张涵信,符松. 基于 Boltzmann 模型方程的气体运动论 HPF 并行算法[J]. 计算物理,2003,20(1):1-8
- [8] 陆林生,董超群,李志辉.多相空间数值模拟并行化研究[J].计 算机科学,2003,30(3):129-131,137