

一种使用 GPU 加速地震叠前时间偏移的方法

张清¹, 谢海波¹, 赵开勇^{1,2}, 吴庆¹, 陈维³, 王狮虎³, 迟旭光³, 褚晓文²

(1.浪潮集团 高效能服务器和存储技术国家重点实验室, 北京 100085;

2.香港浸会大学 计算机系, 香港;

3.中国石油集团 东方地球物理勘探有限责任公司, 河北 涿州 072751)

摘要: 应用 GPU 通用高性能编程技术实现一种加速地震叠前时间偏移的新方法。该技术是地震勘探处理的常规流程, 其核心算法具有计算密集、数据独立性强、并行性高等特点。通过性能剖析获得其计算热点, 通过 CUDA 技术对其进行并行化改造, 并利用 CUDA 的流技术实现 CPU 到 GPU 的异步传输。通过集群环境下的性能测试, 应用 GPU 并行化的 PSTM 程序可明显缩短运行时间。

关键词: 叠前时间偏移; 地震勘探; 图像处理器; 计算统一设备架构; 流; 并行计算

中图分类号: TP303

文献标识码: A

文章编号: 1674-7720(2011)10-0087-04

A method of accelerating pre-stack time migration using GPU

Zhang Qing¹, Xie Haibo¹, Zhao Kaiyong^{1,2}, Wu Qing¹, Chen Wei³, Wang Shihu³, Chi Xuguang³, Chu Xiaowen²

(1.The State Key Laboratory of High Performance Server and Storage Technology, Inspur Group, Beijing 100085, China;

2.Department of Computer Science, Hong Kong Baptist University, Hongkong, China;

3.BGP, Zhuozhou 072751, China)

Abstract: A novel method of accelerating Pre-stack Time Migration (PSTM) by applying high-performance general-purpose GPU programming techniques is presented. PSTM is the key technology for seismic information processing. The algorithm is computing intensive, data independent with high parallelism. In this paper, we profile the algorithm, estimate the hotspot and parallelize the key function by adopting CUDA programming technology.

Key words: PSTM; seismic exploration; GPU; CUDA; stream; parallel computing

1 PSTM 技术

1.1 PSTM 简介

PSTM 是复杂构造成像最有效的方法之一, 能适应纵向速度变化较大的情况^[1], 适用于大倾角的偏移成像^[2]。PSTM 经过多年研究, 20 世纪 90 年代初期开始初步应用, 中后期在不少探区的地震勘探中发挥了重要作用, 进入 21 世纪后开始了较为广泛的应用, 目前部分处理公司和计算中心已将该技术作为常规软件加入到常规处理流程中, 成为获取保幅信息实现属性分析、AVO/AVA/AVP 反演和其他参数反演的重要步骤和依据。

PSTM 方法主要分为两类, 即用于准确构造成像的 PSTM 和振幅保持 PSTM, 每一类方法都有两种实现方式: Kirchhoff 型和波动方程型, 而目前工业界最成熟的是 Kirchhoff 型 PSTM^[3]。PSTM 每输出一个地震道, 就是

一次海量运算。以 1 ms 采样、6 s 数据为例, 一个地震道的输出需要至少 1 000 万道甚至更多(偏移孔径决定)的输入道, 每一个点要做两次均方根运算以及两次加法运算, 振幅补偿两次乘法运算。如此计算下来, 实现一道偏移需要 $1\,000\,000 \times 6\,000 \times 2 \times (\text{平方} + \text{加法} + \text{乘法})$ 次数学运算, 计算量和需要处理的数据量都极其巨大^[4,5]。目前, 人们往往使用大规模的服务器集群来进行叠前偏移处理, 其原理是将数据先分配到各个 CPU 核上, 然后由各个 CPU 核单独进行计算, 最后将结果汇总输出。这种做法消耗了大量的时间、电力和维护费用。而且, 随着人们对石油勘探地震资料处理的周期要求越来越短, 精度要求越来越高, 服务器集群的规模越做越大, 在系统构建成本、数据中心机房空间、内存和 I/O 带宽、功耗散热和电力限制、可管理性、编程简易性、扩展性、管理维护费用等方面都面临着巨大的挑战。

技术与方法 Technique and Method

1.2 PSTM 研究现状

自从 20 世纪 90 年代以来,叠前时间偏移在国外取得了很大发展。在理论研究方面,Bleistein、Bortfeld 和 Hubral 等进行了一系列有关真振幅叠前时间偏移理论的研究工作^[6],Schneider 给出了 Kirchhoff 型真振幅偏移权函数的一般公式。Graham A.Winbow(1999)推出了控制振幅的三维叠前时间偏移的权函数的显式公式,并且利用真振幅权函数估计进行了振幅补偿。另外,在权函数改进的基础上,提出了高精度的弯曲射线法绕射走时计算方法。在应用方面,最近几年,在常规叠前时间偏移基础上,研究开发了多种保幅型叠前时间偏移软件,尤其是 Kirchhoff 保幅型叠前时间偏移软件取得了巨大成功。随着 GPU 的出现,利用 GPU 的多核计算资源,实现对 PSTM 的并行处理,加速 PSTM 的执行时间,成为一个研究热门,国内外多家研究及工程机构皆针对此技术进行了研究。

1.3 GPU 通用高性能编程技术

多核技术已经代替单纯 CPU 频率的提升而成为计算机性能发展的主流。除了 CPU 技术之外,采用 GPU、FPGA 加速并与 CPU 异构并行的方式在这两年大放异彩。使用这种架构的技术如 ClearSpeed 的 FPGA 加速方案、AMD 的 Stream 方案、Mercury 的 Cell BE 加速方案和 NVIDIA 的 CUDA 方案。

GPU 已经体现了较 CPU 更高的晶体管密度和计算能力,相较 FPGA 等的实现,GPU 是一种非常成熟的商业芯片,有利于应用的快速部署,并表现出更好的性价比。从 2000 年左右开始,有部分研究者开始采用 GPU 强大的图像处理能力来进行通用计算^[7]。传统的 GPGPU (General-Purpose Computation on Graphics Processing Units)

的处理方法是把通用的计算问题转换为 GPU 中能处理的图形问题,然后通过图形编程语言来进行编程。这样的编程模式使得开发难度很高,开发者必须掌握强大的图形处理能力。

从 2006 年开始,NVIDIA 开始研究通用的 GPU 处理架构并推出计算统一设备架构 CUDA (Compute Unified Device Architecture),使得开发者不需要有很强的图形开发能力,而是采用扩展了的 C 语言对 GPU 进行直接编程。NVIDIA 把顶点处理和面处理整合到同一个处理芯片上,从而获得 GPU 硬件在架构上的通用性。经过几年的发展,CUDA 架构更加灵活,更适合通用计算。这使得 CUDA 方案成为异构并行模型中的佼佼者。该方案入门门槛低,程序移植效率高,加速比好。图 1 描述了 CUDA 技术体系结构。

2 PSTM 热点分析

通过分析 PSTM,不难发现整个程序分为两个部分,一是 FFT 计算部分(以下简称 FFT),二是 PSTM 计算部分(以下简称 PSTM Kernel)。FFT 为 PSTM Kernel 作数据准备,它主要是对输入地震道数据进行 FFT 计算,在这里只对 PSTM 的实际运行情况进行定量分析,找到 PSTM 中的热点计算部分。

2.1 测试环境及数据

测试环境包括硬件环境、软件环境、运行软件;测试数据为输入地震道数据集;对于成像空间而言,它是四维的,其中第一维为 X 轴方向的大小 NX ,第二维为 Y 轴方向的大小 NY ,第三维为炮点与接收点的偏移范围大小 $NOFF$,第四维为 Z 轴方向的大小 NZ 。为准确测试出热点,特选择一个线偏成像空间和一个体偏成像空间进行热点测试,具体各项参数如表 1 所示。

2.2 分析结果

为了保证数据的准确性,对线偏和体偏分别进行 10 次测试,计算 10 次 PSTM 运行的平均 CPU 使用时间和

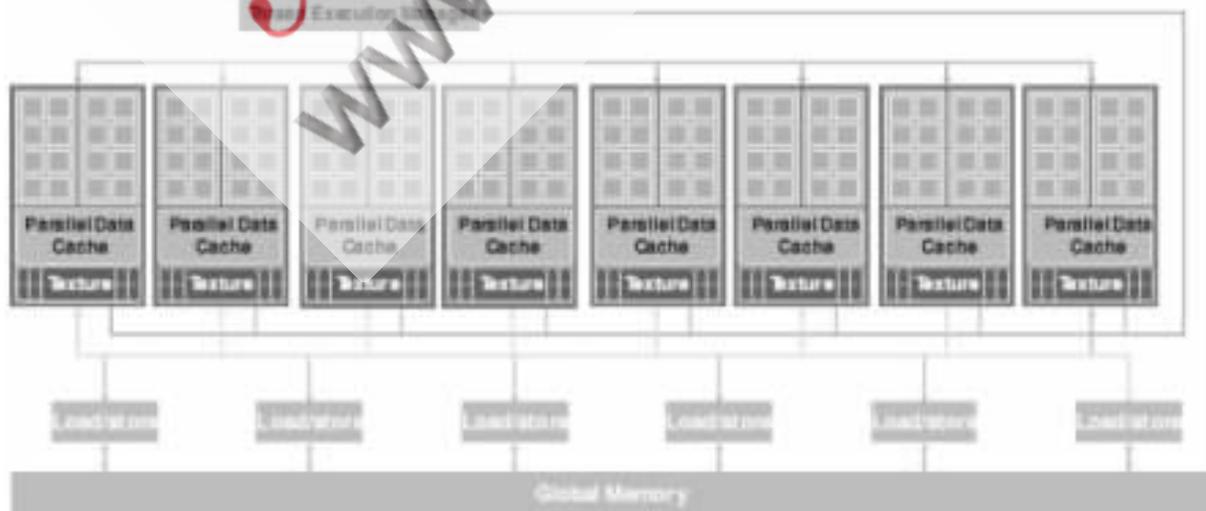


图 1 CUDA 技术体系结构图

技术与方法 Technique and Method

表 1 测试环境和测试数据

| | | | |
|------|---|-------|-------|
| 硬件环境 | 平台:浪潮“倚天”桌面超级计算机 | | |
| | CPU: Intel® Core(TM) i7 CPU 920@2.67GHz | | |
| | 内存: 2.67 GHz, 8 GB | | |
| 软件环境 | GPU: Tesla C1060 | | |
| | OS: 64 位 Linux RedHat 4.5 | | |
| | 编译器: ICPC | | |
| 运行软件 | 编译选项: O2 | | |
| | CPU 版 PSTM 软件 | | |
| | 维度 | 线偏 | 体偏 |
| 成像空间 | 第一维 NX 大小 | 1 | 11 |
| | 第二维 NY 大小 | 1 098 | 366 |
| | 第三维 $NOFF$ 大小 | 55 | 55 |
| | 第四维 NZ 大小 | 1 504 | 1 504 |
| 输入道集 | 输入道数: 661 443 | | |
| | 道长: 6 000 | | |

它的两个计算部分平均执行时间, 测试结果如表 2 所示。不难发现, PSTM_Kernel 是整个 PSTM 的核心计算部分, 占到整个运行时间的 90% 以上, 所以 PSTM_Kernel 为 PSTM 的热点计算部分。

既然 PSTM_Kernel 为 PSTM 的热点计算部分, 如果对 PSTM_Kernel 进行并行化改造, 它的运行时间将大大减少, 则整个 PSTM 的运行时间也将随之明显减少, 其性能将大大提高, 所以在本文的研究中, 将对 PSTM_Kernel 进行并行化的改造, 以加速 PSTM 的运行效率。

3 PSTM_Kernel 并行化改造

3.1 PSTM_Kernel 串行算法分析

PSTM_Kernel 串行算法的基本思想是: 对于每一道地震道数据而言, PSTM_Kernel 都循环计算成像空间中每一个点的走时, 即每一个成像点到此道地震道数据对应的炮点的走时, 和每一个成像点到此道地震道数据对应的接收点的走时^[7]。然后根据这两个走时之和取出对应的地震道数据, 更新此成像点的数据, 实现叠前时间偏移计算。

PSTM_Kernel 串行算法的核心是按照成像空间 X 轴方向、 Y 轴方向、 Z 轴方向三层循环进行实现的, 其具体过程如下:

- (1) 循环取出每个成像点在成像空间中对应的 X 、 Y 、 Z 坐标, 确定成像点位置;
- (2) 根据成像点位置计算出其到炮点和接收点的走时;
- (3) 根据这两个走时之和取出对应的地震道数据, 然后更新此成像点在成像空间中的数据, 实现叠前时间偏移计算。

从以上串行算法分析, 每一个成像点到炮点和接收点的走时计算是数据独立的, 并不存在依赖性, 因此存在较好的并行性, 适合 GPU 等细粒度并行体系结构的加速。在此采用 CUDA 技术进行实现。

3.2 CUDA 并行实现

3.2.1 线程计算粒度选择

应用 CUDA 模型加速通用计算, 涉及线程模型及对应的计算粒度问题。如果线程计算的粒度太小, 例如成像空间的每一个点对应一个线程进行计算, 则线程总数为成像空间四个维度数值的乘积, 即 $NX \times NY \times NOFF \times NZ$ 。由于每道数据并非对成像空间的所有点都有贡献, 因此细粒度的并行会产生较多无效计算线程。这种情况下, 线程计算粒度太小反而会导致线程的并发度不高。如果线程计算粒度太大, 一个线程进行多点的循环计算, 则线程数过少, 不能使流处理器满负荷运行, 性能同样不能达到最优。通过性能测试发现, 最佳计算粒度为: 32 个线程负责计算 NZ 个点, 即 32 个线程负责计算成像空间 Z 方向上一条线上的所有点, 此时计算性能最为理想。

3.2.2 线程模型选择

线程模型是用来明确 CUDA 程序内核的执行配置。根据 GPU 硬件资源, 定义网格和线程块, 好的线程模型能大大提升程序的性能。

(1) 网格 (grid) 的定义: 即把所有线程如何进行分块, 定义线程块数和线程块的组织方式, 这里根据成像空间的 X - Y 平面来划分线程块, 其具体定义为 $\text{dimGrid}(NX, NY/10)$;

(2) 线程块 (block) 的定义: 即定义一个线程块有多少个线程和线程的组织方式, 根据内核所需的寄存器数和共享内存数量, 定义一个线程块为 320 个线程, 其具体定义为 $\text{dimBlock}(32, 10)$;

(3) 线程模型描述: grid 和 block 都是定义为二维的, 线程总数为 $NX \times NY \times 32$, 线程块数为 $(NX \times NY)/10$ 。每个 block 的 320 个线程处理 10 个 (x, y) 坐标所对应的 Z 轴的所有点, 即处理 Z 轴的 10 条线。如果把每个 block 的 320 个线程按照 32 个线程进行分组, 每一组为一个 warp, 则整个 block 有 10 个 warp, 第 0 个 block 的第 0 个 warp, 即 threadIdx.y 等于 0 的线程处理第 0 个 (x, y) 坐标对应的 Z 轴的第一条线; 第 0 个 block 的第 1 个 warp, 即 threadIdx.y 等于 1 的线程处理第 1 个 (x, y) 坐标对应的 Z 轴的另一条线, 依此类推, 第 0 个 block 的第 9 个 warp, 处理第 9 个 (x, y) 坐标对应的 Z 轴的一条线。同理, 其他 block 处理另外 10 个 (x, y) 坐标对应的 Z 轴的 10 条线。10 条线并行进行走时计算, 可以使计算更均衡, 性能更高。

表 2 PSTM 软件各部分运行时间表

| | PSTM 运行时间/s | FFT 运行时间/s | PSTM_Kernel 运行时间/s | PSTM_Kernel 时间占用比例/% |
|----|-------------|------------|--------------------|----------------------|
| 线偏 | 2 039 | 210 | 1 829 | 90 |
| 体偏 | 6 626 | 210 | 6 416 | 97 |

技术与方法 Technique and Method

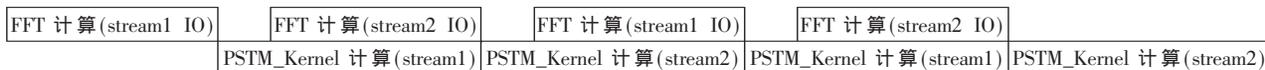


图 2 异步 IO 逻辑结构图

3.3 CPU 与 GPU 的异步 IO 实现

基于上述的 CUDA 并行化改造,显著提升了计算性能。而随之带来的问题是 GPU 加速所必然造成的 IO 传输问题。PSTM_Kernel 执行之前,需要 FFT 计算模块对输入道进行预处理,之后这些数据需要从系统内存传递至 GPU 显存。这是 GPU 计算的引入所带来的额外 IO 开销。如果采用同步方式进行数据传输,即:对一道地震道数据而言,完成一道 FFT 的计算,就将数据传输给 GPU 再进行叠前时间偏移计算,那么 IO 开销将会很大,对性能造成很大影响。

本文采用 CPU 与 GPU 的异步传输方式来试图减少上述 PSTM 执行的总时间。采用 CUDA 的流(stream)技术,利用双流双缓冲,实现 CPU 与 GPU 的异步 IO 传输,其逻辑结构图如图 2 所示。

4 地震资料测试

4.1 测试环境及数据

通过一套小规模集群系统测试 GPU 加速 PSTM 的效果。具体各项参数如表 3 所示。

表 3 测试环境及测试数据

| | |
|--------|--|
| 节点硬件环境 | CPU:2×Intel Xeon 四核 E5620(Westmere,2.4 GHz) 内存:6×4 GB DDR3 ECC REG 1333 内存 GPU:2×Tesla C1060 |
| 软件环境 | OS:64 位 Linux RedHat 4.5 编译器:ICPC 编译选项:O2 |
| 运行软件 | CPU 多线程版 PSTM(16 计算线程/节点) GPU 版 PSTM |
| 成像空间 | 第一维 NX 大小:177 第二维 NY 大小:1097 第三维 NOFF 大小:1 第四维 NZ 大小:1 504 |
| 输入道集 | 输入道数:1 300 万 数据大小:72 GB 道长:6 000 ms |

4.2 性能结果

为了保证测试性能结果的稳定性,对上述体偏作业进行了 3 次测试,CPU 多线程版 PSTM 在集群上运行 3 次的平均时间是 110 908 s,而 GPU 版 PSTM 在集群上运行上述同样体偏作业 3 次的平均时间是 21 454 s,GPU 版 PSTM 运行的性能是 CPU 多线程版 PSTM 的 110 908/21 454=5 倍。

4.3 成像效果图比对

通过上述作业,从获得的效果图来看,图像不存在明显差异,证明 GPU 加速 PSTM 运行结果的正确性。

通过对 PSTM 算法的性能剖析,分析算法的热点函数,并对该热点函数进行 GPU 并行化改造。初步改造移植结果说明,使用 GPU 进行加速可显著提高叠前时间偏移计算速度,实验数据证明了加速效果。测试数据表明,对于一个完整的 PSTM 体偏作业,一个 GPU 节点的运行时间是一个采用最新双路 CPU 节点,并运行商用多线程 CPU 版 PSTM 时间的 1/5。由此可见,PSTM 高并行度、无数据依赖性等特征使得其较适合 GPU 类设备的加速。

参考文献

- [1] 张占杰.叠前时间偏移技术及其在东海复杂地震资料处理中的应用[J].海洋石油,2007,27(3):22-26.
- [2] 王余庆.叠前偏移技术探讨及应用[J].西北油气勘探,2006,18(2):31-39.
- [3] 王棣.叠前时间偏移方法综述[J].勘探地球物理进展,2004,27(5):313-320.
- [4] 罗银河.Kirchhoff 弯曲射线叠前时间偏移及应用[J].天然气工业,2005,25(8):35-37.
- [5] 洪钊峰.GPU 计算:在石油勘探领域的革命[EB/OL]. [2009-5-13].http://server.it168.com/a2009/0513/276/000000276186.shtml.
- [6] 张丽艳.相对振幅保持的转换波叠前时间偏移方法研究[J].石油地球物理勘探,2008,43(2):30-36.
- [7] 李博.地震叠前时间偏移的一种图形处理器提速实现方法[J].地球物理学报,2009,52(1):245-252.

(收稿日期:2010-12-20)

作者简介:

张清,男,1981年生,硕士,主要研究方向:高性能计算。