

# 一个层次式面向对象并行计算框架的设计

李英军, 吕建, 于大川, 马晓星

(南京大学计算机软件新技术国家重点实验室, 南京 210093)

**摘 要:** 本文讨论了一个层次式并行应用软件开发框架的设计思想, 将并行计算底层细节进行逐级抽象, 采用面向对象封装与继承机制, 为科学计算领域提供了快速原型实验和应用开发环境. 该框架的层次结构使之成为独立于体系结构的并行开发环境, 它的高层抽象界面允许以顺序方式书写的代码不经任何改变即可在不同体系结构的并行环境上运行.

**关键词:** 应用框架; 并行计算; 面向对象

**中图分类号:** TP3 **文献标识码:** A **文章编号:** 0372-2112 (2000) 08-0113-02

## The Design of a Hierarchical Object-oriented Parallel Framework

LI Ying-jun, LU Jian, YU Da-chuan, MA Xiao-xing

(State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China)

**Abstract:** Parallel computation is always a main focus in scientific application domain, and the high performance and easy developing parallel environment is the fundamental for prototyping parallel applications. In this paper, a hierarchical object-oriented parallel application framework is proposed. It is an innovative parallel class library for structured grid applications, and allows for parallel software development in the preferred serial environment and such software to run invariably and efficiently on parallel architectures.

**Key words:** framework; parallel; object-orientation

## 1 引言

并行计算一直是科学计算应用领域倍受关注的课题, 而并行软件设计与实现的复杂性却限制了并行算法及其应用软件的发展. 因此, 高性能易使用的并行应用原型实验和开发环境无疑是研制并行软件的基础. 本文将讨论一个层次式并行应用软件开发框架 HOOPE (Hierarchical Object-Oriented Parallel Environment) 的主要设计思想: 以油气勘探领域叠前深度偏移为应用目标, 运用虚拟共享网格 VSG (Virtual Shared Grids) 和 Overlay 网格数据覆盖技术, 采用 SPMD (Single Program Multiple Data) 计算模型, 内含多种网格分割机制, 并将并行计算底层细节进行逐级抽象, 采用面向对象封装与继承机制, 为并行计算领域提供了快速原型实验和开发环境. HOOPE 的层次结构使之成为独立于体系结构的并行开发环境, 它的高层抽象界面允许以顺序方式书写的代码不经任何改变即可在不同体系结构的并行环境上运行. 目前国际上已对大规模科学并行计算做过较多研究, 例如美国 Los Alamos 国家实验室在离子物理领域研制的并行应用开发环境 POOMA<sup>[1]</sup>等. 另外许多研究机构用在程序设计语言中嵌入并行语句的方式简化并行开发过程, 例如 IBM 等研究机构为现有语言所作的并行扩充 ABC

++ , pC++ , Fortran90D<sup>[2]</sup>等. 然而用对语言的并行成分扩充方式实现并行开发环境存在着诸多问题, 如它依赖对现有编译器的修改, 而编译程序与具体并行机体系结构相关等. 因此, 对语言的并行扩展本身就存在移植性问题.

## 2 系统层次式体系结构

HOOPE 并行应用框架主要包括四个层次: 专业应用框架层, 应用组件层, 抽象数据层和并行支撑层. 框架中的每个层次均承担相对独立的任务, 每一层次均可由不同需要的用户使用. 上面层次可以使用下面层次的对象来完成其特定的功能, 而下面层次的对象主要用作上面层次的支撑环境. 如应用组件层中的并行组件主要是利用抽象数据层中所提供的领域数据类型及其操作, 根据领域的需要加以并行组合而成. 抽象数据层中的高级领域数据类型及其相关操作的并行实现则基于并行支撑层中所提供的各种机制. 而并行支撑层的实现则基于 MPI 环境.

层次式体系结构的主要特点如下: 首先, 其分层结构将并行应用开发的责任分解. 软件和领域专家主要集中于框架的低级层次, 而应用开发者则可直接使用上面层次的功能来构造并行应用软件. 从而可简化并行程序开发过程, 缩短并行应

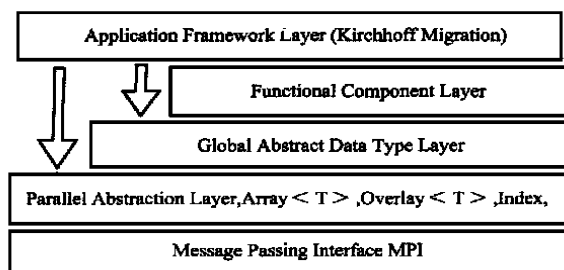


图1 HOOPE并行应用框架层次体系结构

用软件开发周期。其次,由于并行应用软件通常基于应用组件层和抽象数据层,因此,只要将并行支撑层稍作修改即可在顺序、分布和并行平台上运行而不必修改源代码。从而便于应用程序在不同的并行体系结构间的移植。最后,由于并行应用软件常用的关键组件已集成在应用框架之中,因此,可集中采用现有的各种测试、验证和优化技术来保证其质量和运行功效,从而可提高并行应用软件的质量和功效。

### 3 框架的可扩充性和可使用性

#### 3.1 框架的可扩充性

在并行框架的三个层次上设计了稳定点和可变点。稳定点抽象了并行计算中相对稳定的结构部分,形成了框架的控制流。同时可变点定义了框架中的“热点”,允许用户插入不同方法的组件。可变点处提供的插槽和插入技术主要基于面向对象技术中的类继承、函数重载和多态机制。

框架的第一层,即专业应用框架层抽象了 Kirchhoff 积分偏移算法的控制流,形成了一个稳定的框架。它为偏移成像计算定义了参数分析、计算映射表格、偏移成像和后继处理等几个主要步骤,且对每一个步骤均定义了通用的用户界面,专业用户可以使用该通用界面重载不同的顺序实现方法,也可以使用高级抽象数据层中定义的并行 VField 等类,重载它们的方法,达到并行计算的目的。

框架的第二层次,即抽象数据类型层上的稳定点抽象了具体计算方法中大块数据分割方法,以及在数据分割策略下的计算函数接口。如射线旅行时间表并行数据类型 VField 使用并行数组 Array < float > 管理速度场和旅行时间表数据。旅行时间表计算函数定义为虚函数,即该层的可变点,用户可以通过继承 VField 类并重载计算函数,将不同计算方法函数插入框架“热点”。另外,层次结构框架的开放性和可扩充性允许用户插入新的并行抽象数据类型。

第三层为并行数组层,该层基本上属于稳定层,它抽象了数组分布和通讯策略。它的数组分布算法和通讯算法理论上允许用户重载,但是并不提倡。因为该层的数据分布和通讯算法均通过了优化和形式化正确性证明。重载该层的函数将对框架的运行效率产生影响。

#### 3.2 框架的可使用性

该框架为用户提供了三种应用方式:(1)直接使用

HOOPE 提供的并行数组层、功能组件层以及应用框架构筑并行应用程序,例如我们可以使用 VField 类和 Image 类以及 Kirchhoff 偏移框架直接构造叠前深度偏移程序。(2)专业用户使用框架提供的可变点插入新算法,例如:用户要将 3D-DMO 算子加到框架的抽象数据类型层中,他仅需要用 DMO 算子重载 Mapping-Amp 方法即可,最终用户使用界面保持不变;如果专业用户欲使用新的射线旅行时算法,他仅需重载 VField 类中的 Compute 方法即可。(3)深层并行应用用户可以在并行数组层增加新的数据分布和通信方法,而保持高层用户界面不变。

### 4 结论

本文描述了一个层次式面向对象并行应用框架 HOOPE 的体系结构、技术特点和实现策略。目前,针对油气勘探领域叠前深度偏移应用,正在努力研究并部分实现了该并行环境的基础数组以及其分割策略部分,应用领域深度偏移高级抽象数据类型如 Image、VField 等将在基础环境上建造。HOOPE 提供的三种数据分割策略可以基本满足目前应用领域的需求,随着应用需求的不断增加,我们也将考虑新的并行数组分割策略。考虑到由于对 C++ 运算符的重载使得复杂并行表达式的运行效率降低,我们已对此提出相应对策。该框架目前集中采用 SPMD 并行模式,将来可能嵌入任务并行模式,以提高其运行效率。

#### 参考文献:

- [1] Reynders J V W, Hinker P J, Cummings J C. POOMA: A framework for scientific simulation on parallel architectures [R]. Paper from Los Alamos National Laboratory, USA, 1995.
- [2] Gregory V. Wilson and Paul Lu. The editor: parallel programming using C++ [M]. Cambridge, Massachusetts: The MIT Press, 1996.
- [3] William Croup, Ewing Lusk, Anthony Skjellum: Using MPI, Portable Parallel Programming with the Message-Passing Interface [M]. Cambridge, Massachusetts: The MIT Press, 1994.

#### 作者简介:



李英军 1962 年 2 月生, 1999 年毕业于南京大学计算机系, 获博士学位, 石油地球物理研究院高级工程师, 南京大学计算机软件新技术国家重点实验室访问学者, 主要研究方向为面向对象程序设计和应用框架, 油气勘探应用软件集成平台与分布式计算等, 电话: 13601064127, Email 地址: yjlee@mx.cei.gov.cn.

吕建 1960 年 3 月生, 南京大学计算机系教授, 博士生导师, 研究领域为软件自动化, 并行程序形式化方法和面向对象语言及环境等。