基于OpenACC的GRAPES_GLOBAL模式长波辐射异构并行化研究

孙晨　王彬　顾文静　魏敏
（国家气象信息中心，北京 100081）

摘要：气象数值模式是天气预报的基本工具和方法，随着技术的发展，模式分辨率有了大幅的提高，分辨率的提升使计算量呈指数增长，然而气象预报的时效性对并行程序的设计与计算平台的性能都提出了更高的要求。以GRAPES_GLOBAL数值天气预报模式为研究案例，以“神威·太湖之光”新一代国产异构众核高性能计算系统为试验平台，分析其程序结构及计算原理，定位影响模式并行效率及扩展性的热点子程序。通过调整程序结构以及添加协处理器加速指示语句，并针对模式系统消息缓存过大等问题，为长波辐射过程的每个热点子程序分别设计了高效的通讯策略。实现了“粗粒度MPI并行+细粒度众核OPENACC并行”多级异构并行方案，使其普遍达到3～6倍的加速。本试验充分继承了原始代码的MPI 级并行，同时利用数量众多的协处理器为其中的热点函数提供加速，有效提升模式的并行效率，节约了开发成本。

关键词：神威·太湖之光，OpenACC，GRAPES模式，长波辐射过程

DOI：10.3969/j.issn.2095-1973.2018.01.027

A Research About Hybrid Programming and Parallelization of GRAPES_GLOBAL Based on OpenACC

Sun Chen, Wang Bin, Gu Wenjing, Wei Min
(National Meteorological Information Centre, Beijing 100081)

Abstract: Numerical weather model is a basic method and tool of weather forecasting. As the development of technology, the model resolution has been improved greatly, it, however, brings an exponentially-increasing computation cost. The timeliness for the weather forecasting puts forward more advanced requests to the program designing and the performance of computing platform. In this paper, we take GRAPES_GLOBAL as an example, to explore the feasibility of hybrid programming and optimization on the Sunway TaihuLight (new domestic high-performance computing system). By analyzing the program structure and calculation principle, we find hotspot subroutines which are influencing the parallel efficiency and extensibility, then we design an efficient communication strategy for each subroutine in the long-wave radiation by adjusting the structure of program and adding the coprocessor acceleration indicator statements. The implement of the hybrid programming of MPI parallel computation on CPU and OpenACC parallel computation on Many-Core shows that an acceleration ratio of hotspot subroutines is 3-10 times faster than before. This experimental results may confirm that the methods can inherit the most of the original MPI parallel computing codes and reduce the developing costs significantly.

Keywords: Sunway TaihuLight system, OpenACC, GRAPES model, long-wave radiation process

0　引言

GRAPES（Global/Regional Assimilation and Prediction System）是中国气象局自主研发的静力/非静力多尺度通用数值预报系统，该系统是气象与气候研究的基础和核心。

收稿日期：2017年7月31日；修回日期：2017年12月20日
第一作者：孙晨（1990—），Email：marksun1990@126.com
资助信息：国家重点研发计划项目（2016YFA0602102）；
公益性行业（气象）科研专项（GYHY201306062）

GRAPES系统的核心部分包括模式动力框架和物理过程，动力框架中计算时间最长的为半拉格朗日计算和求解Helmholtz大型线性代数方程组，物理过程中耗时较大的为辐射和微物理过程，其中辐射过程计算耗时占到GRAPES模式的40%，所以提升辐射过程的性能和计算效率，对整个GRAPES模式的性能提升具有重要科研价值和实际意义[1-2]。

近年来，并行计算技术日趋成熟，GPU巨大的计算能力已经吸引了越来越多的科研人员将其作为一个高效益低成本的高性能计算平台，然而，要在GPU

硬件上实现加速需要通过对底层的API进行编程来实现，程序编写复杂、难度大，且容易形成高度依赖特定设备的代码。因此，一种基于指令制导计算的编程模OpenACC应运而生[3]，OpenACC的编程机制是在源程序中添加少量编译标识，编译器根据作者的意图自动产生低级语言代码，无需学习新的编程语言和加速器硬件知识，变更迅速掌握。只添加少量编译标识不破坏原代码，开发速度快，即可并行执行又可恢复串行执行；硬件更新时，重新编译一次代码即可，不必手工修改代码。OpenACC标准制定时就考虑了目前及将来多种加速器产品，同一份代码可以在多种加速器设备上编译、运行、无成本切换硬件平台。

本文以GRAPES模式物理过程中的长波辐射为加速处理对象，依托国家超级计算无锡中心的“神威·太湖之光”高性能计算机系统，设计一种基于OpenACC的加速算法，最大程度继承原始代码的MPI 级并行，同时利用数量众多的协处理器加速热点函数。

1 “神威·太湖之光”简介

2016年6月20日，新一期全球超级计算机500强榜单在德国法兰克福举办的国际超算大会（ISC）上公布，“神威·太湖之光”超级计算机荣登榜首。

“神威·太湖之光”由国家并行计算机工程技术研究中心研制，是全球第一台运行速度超过10亿亿次/s的超级计算机，峰值性能高达12.54亿亿次/s，持续性能达到9.3亿亿次/s。该系统部署在国家超级计算无锡中心，系统由40个运算机柜和8个网络机柜组成。每个机柜有 4 个超级节点，每个超级节点包括32个节点板，每个节点板上有 4 个节点卡，每个节点卡有两个节点，每个节点上装有 1 个“申威26010”众核处理器和32GB的DDR3内存，一台机柜就有1024块处理器。节点之间通过基于PCI-E3.0（外设部件互联标准扩展3.0版）的神威高速网络进行互联。

“神威·太湖之光”系统全部采用“申威26010”众核处理器，架构如图1所示。

该芯片主频为145 GHz，由4个核组组成，每个核组包含一个主核和一个从核簇，每个从核簇由64个从核组成，共260个处理器核心，同一个核组内的主核和64个从核共享主存，每个从核有独立的、容量为64 KB的高速缓存（Local Data Memory，LDM），支持DMA（Direct Memory Access）的方式在主存和LDM之间传输数据。主核作为处理器核心，可以进行通信、IO、计算等操作，同一核组内的64个从核作为加速计算部件，用来加速主核代码中计算密集部分。

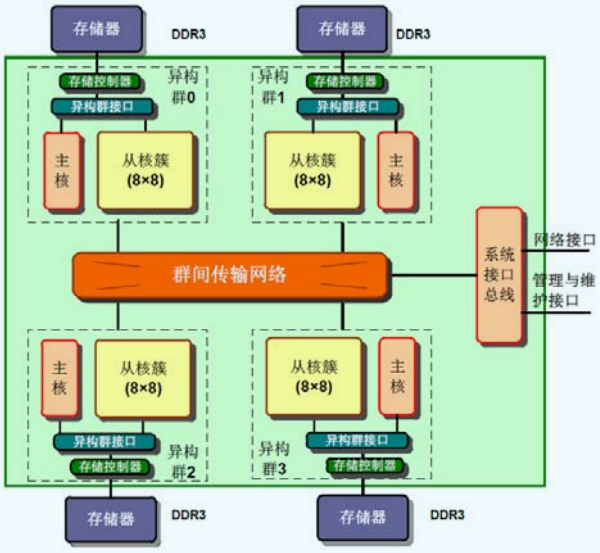

图1 “申威26010”处理器架构图

Fig. 1 Structure of the SW-26010 processor

2 OpenACC*总述

OpenACC是由OpenACC组织（www.openacc-standard.org）于2011年推出的众核加速编程语言，以编译指示的方式提供众核编程所需的语言功能，其主要目的是降低众核编程的难度。用于C/C++和Fortran的OpenACC API和指令负责把底层的GPU任务交给编译的同时，又提供跨系统、跨主机CPU和加速设备的支持[3]。

“神威·太湖之光”计算机系统中的OpenACC*语言是在OpenACC2.0文本的基础上，针对申威26010众核处理器结构特点进行适当的精简和扩充而来的。

2.1 执行模型

OpenACC*程序的执行模型是在host（主处理器）的指导下，host和device（加速设备）协作的加速执行模型，如图2所示。

程序首先在host上启动运行，以一个主线程串行执行，或者通过使用OpenMP或MPI等编程接口以多个

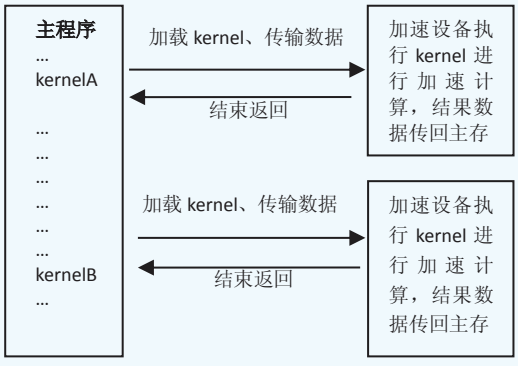

图2 OpenACC*执行模型

Fig. 2 Execution model of the OpenACC*

主线程并行执行，计算密集区域则在主线程的控制下作为kernel或parallel（加速构件）被加载到加速设备上执行。加速构件的执行过程包括：在设备内存上分配所需数据空间；加载构件代码（包含相关参数）至device；构件将所需数据从主存传输至设备内存；等待数据传输完成；device进行计算并将结果传送回主存；释放设备上的数据空间等。大多数情况下，host可以加载一系列构件，并在加速设备上逐个执行[4]。

OpenACC*支持三级并行机制：gang、worker、vector。gang是粗粒度并行，在加速设备上可以启动一定数量的gang。worker 是细粒度并行，每个gang内包含有一定数量的worker。vector并行是在worker内通过SIMD或向量操作的指令级并行。

在“申威26010”中，一个运算控制核心（简称主核）仅控制一个运算核心阵列(加速设备，简称从核阵列)的运行，每个运算核心阵列内有64个运算核心（简称从核），每个运算核心可以运行一个加速线程。默认情况下，64个加速线程被组织成64个gang、每个gang内一个worker、worker内可以vector并行的逻辑视图。通常情况下，尽量用满64个加速线程可以获得较好的运行性能。

2.2 存储模型

在CUDA和OpenCL等低层次加速器编程语言中，主机和加速器存储器分离的概念非常明确，内存间移动数据的语句占据用户大部分代码。

“申威26010”中从核和主核共享内存，从核可直接访问主存空间，并在从核内提供加速线程私有的LDM，加速计算需要存放到LDM的数据由从核控制传输。存储模型如图3所示。

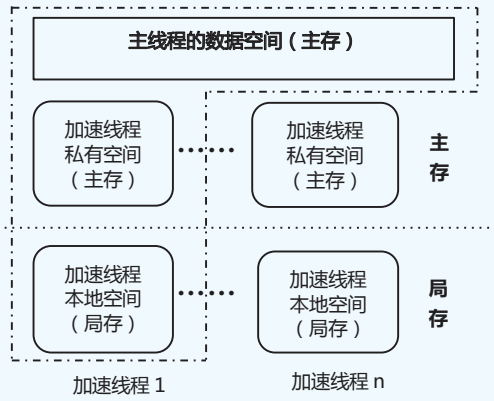

图3 OpenACC*存储模型

Fig. 3 Memory model of the OpenACC*

OpenACC标准中的数据管理功能将不产生实际的作用，但直接访问主存往往带来性能损失，因此需要充分利用从核中的LDM，提升数据访问效率。在“神威·太湖之光”中，OpenACC*对OpenACC标准所做的主要功能延伸和语法扩展就是为了解决共享内存架构下片内高速存储空间的使用问题。

“申威26010”中，从一个加速线程的角度来看，其可见的数据空间有三种[4]：

1）主线程数据空间：位于主存，主线程的内存空间对其创建的加速线程直接可见，加速线程可以直接访问相应的数据；对于主线程创建的多个加速线程而言，这部分空间是共享的；程序中在加速区外定义的变量，均位于该空间内。

2）加速线程私有空间：位于主存，每个加速线程有独立的私有空间，程序中使用private、firstprivate子句修饰的变量将存放于该空间内。

3）加速线程本地空间：位于设备内存，每个加速线程有独立的本地空间LDM，本地空间的访问性能是三种空间中最高的。程序中使用local、copy等数据子句修饰的变量将由编译系统控制全部或局部存放于LDM内。主存与本地空间的数据交互由加速线程控制。

3 长波辐射过程OpenACC优化方案

GRAPES模式的辐射模型采用欧洲中期天气预报中心（ECMWF）的长短波辐射方案，该方案将整个大气层划分成水平方向和垂直方向的三维网格，水平方向为横跨地球表面的二维网格，垂直方向则表示大气的层数[5]。数值天气预报模式是一种非线性化离散计算模式，其计算量巨大[6]，最初的都是通过CPU计算实现，GPU出现之后，虽然可以把该算法移植到GPU上去并行执行，但编程难度大且易出错。因此，文中设计一种基于OPENACC加速方法，以实现通过简化的编程算法和简洁的Fortran语言代码完成对长波辐射过程并行处理过程。

3.1 辐射过程的构成

辐射过程是GRAPES系统中一个重要的物理过程，包含云结构的描述和辐射算法（RRTM模块）两部分。其中云对调解辐射平衡起着至关重要的作用[7]，本系统在云产生器的基础上利用McICA进行辐射计算。辐射过程首先调用mcica_subcol_lw函数实现云结构的描述，之后调用RRTM模块进行辐射计算。RRTM模块初始化函数rrtmg_lw_init读取数据文件中的输入数据；然后调用rrtmg_lw_rad函数计算辐射传输的过程；最后调用辐射模型子函数rrtm_lw；rrtm_lw子函数一次计算一个垂直列的辐射传输值[6]。它主要包含以下几个基本步骤：inatm，准备大气剖面；cldprmc，计算云层光学深度；setcoef，计算这种大气

剖面下辐射传输算法所需的各种量；taumol，计算气体光学深度和普朗克分数；rtrnmc，计算晴空和云层的辐射传输值。其中，taumol中的16个子函数taugb，计算了全部16个长波波域谱带的气态光学厚度和普朗克（Planck）函数。辐射过程的函数包结构如图4所示。

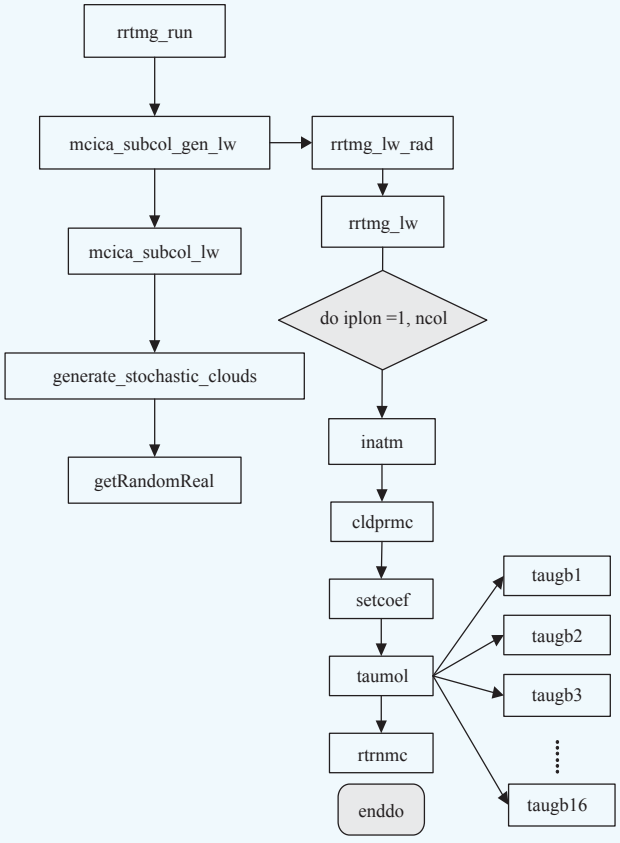

图4 长波辐射过程的函数结构

Fig. 4 Structure diagram of the long-wave radiation function

3.2 并行优化方案

OPENACC并行化的唯一目的是利用从核充足的硬件资源来提高程序运行速度，缩短运行时间。整个长波辐射过程中最耗时间的是RRTMG_LW主过程，计算并行化的目标是将循环迭代步分散到多个不同的线程上执行，这些线程运行在多个加速器核心上，从而将计算任务由CPU转移到加速器上，减轻CPU的负担。

结合程序代码的分析，确定需要众核加速的代码段，并根据源代码数据结构进行预处理，对相关数组和循环进行调整，以适合OpenACC加速。优化的基本思想是确定将程序中的标量和关键数组尽可能多的放到局存LDM中，并设法提高程序中数据的传输效率，将需要使用众核加速的循环的前后使用OpenACC加速编译指示进行标注，其中PARALLEL表明该部分代码是需要加速执行的并行代码；LOOP表示下面紧跟着的循环需要在加速线程间进行并行划分；DATA表示主从核之间的数据拷贝。

rrtm_lw函数包含一层循环，涵盖inatm、cldprmc、setcoef、taumol和rtrnmc五个子函数调用，每个子函数内部有多个do循环分块，最外层的iplon循环阈值为[1，90]，根据OPENACC以及申威“26010”处理器硬件特点，可以采取的并行化方案有如下几种：

3.2.1 从核并行化方案 1：最外层直接并行

结合辐射过程的计算模型为单柱计算模型，核心迭代针对每个单柱串行进行多个物理过程的计算，每个单柱之间不存在数据依赖关系。可以在该层循环外层添加OpenACC加速指示语句，单个从核将独立运行循环内部的所有函数，完成全部计算后将结果传回主核，代码结构示意如图5所示，此时从核使用率为70.3%。

```
subroutine rrtmg_lw（）
!$acc parallel [子句列表]
do iplon = 1, ncol
call inatm（）
call cldprmc（）
call setcoef（）
call taumol（）
call rtrnmc（）
enddo
!$acc end parallel
```

图5 iplon层直接并行代码结构示意图

Fig. 5 Structure of direct parallel methods on the iplon layer

本段核心代码中，5个子函数中包含众多变量，多为二维数组，少数为三维数组（标量及一维数组产生局存压力较小）。其中输入数组22个，包含温度，压力，痕量气体浓度等多个物理量，总大小约为22×140×60(or 61)×8=1478400 byte；计算过程中的中间变量超过40个，总大小约为2688000 byte；输出数组多为一维数组或标量。即使通过调整代码使计算过程中的大多数中间变量空间能及时释放，栈容量需求也远超局存大小（65536 byte），以0.5°分辨率计算过程为例，无论是添加routine函数将部分从核函数的栈数组放到主存，还是使用分块语句指导数据依次拷入拷出，都会产生极大的通讯开销，影响运行速度。

另外，子函数cldprmc与setcoef只有简单的赋值与计算过程，在原本的运行过程中耗时很少，但与前后子函数存在数据依赖，所以此方案下，同样需要拷贝程序与数据在从核运行，运行时间大量增加。

3.2.2 从核并行化方案 2：循环内移

对于气象模式，运算过程中的中间变量数组通常个数很多并且数据量极大，由于局存压力需要多次多

步拷贝，数据通信会造成较大的时间开销影响加速效果，例如长波辐射过程的rrtmg_lw函数。

此时可以通过循环内移，对大函数进行分块，内移工作从rrtmg_lw函数调用的函数inatm开始，依次对cldprmc、setcoef、taumol和rtrnmc子函数进行循环内移达到子函数分块的目的，之后，对于适合从核并行的inatm、taumol和rtrnmc子函数按OPENACC规则添加导语及子句使其在从核运行，不适合从核运行的cldprmc、setcoef子函数仍在主核运行，代码结构示意如图6所示。此时，数据拷贝通讯量缩短至方案1的20%左右，与此同时，可根据每个子函数的最外层循环数，合并外层循环，增加从核运行指示语句，最大化扩大子函数运行的并行度。

```
subroutine rrtmg_lw（）
!$acc parallel [子句列表]
DO loop = begin_chunk, end_chunk, 1
call inatm（）
Enddo
!$acc end parallel
do iplon = 1, ncol
call cldprmc（）
endo
do iplon = 1, ncol
call setcoef（）
endo
!$acc parallel [子句列表]
DO loop = begin_chunk, end_chunk, 1
call taumol（）
Enddo
!$acc end parallel
!$acc parallel [子句列表]
DO loop = begin_chunk, end_chunk, 1
call rtrnmc（）
Enddo
!$acc end parallel
end subroutine rrtmg_lw
```

图6 循环内移代码结构示意图

Fig. 6 Diagram of the loop inside displacement

循环内移使程序原本的计算顺序发生了改变，由从核单线程串行计算子函数1-5变为先计算所有子函数1，保存计算结果后计算子函数2，以此类推。所以，代码中大部分中间变量以及输出结果需要升维用于解决数据依赖，保证正确性。

循环内移后，局存压力显著降低，特别是rtrnmc函数，只有6个二维输入变量和2个二维中间变量，经过计算和试验，只需最低12个循环分块即可正确运行，通讯次数降低，运算速度显著提高。同时，inatm函数作为大气剖面准备模块，输入数据较多，中间变量较少，循环内移后，从核运行时所需的循环分块个数以及数据拷贝次数也明显降低，运行速度有所提高。

3.2.3 从核并行化方案 3：循环分列

为最大程度利用异构众核处理器的结构优势，同时进一步降低局存压力，减少通讯时间，可以对循环内移后的子函数进行循环改写，扩大并行度使每个从核承担更少的计算任务，从而降低计算所需通讯量。通过分析子函数循环结构，对于迭代层数较多，耗时较长的计算热点进行进一步从核并行化改写。同时，通过collapse循环合并语句，可以使代码运行时的并行度显著增加。以inatm函数为例，改写循环后k由子函数内层转至函数外，为保证数据正确性，需改写原函数，部分常量数据需要升维以符合最外层k循环，部分变量数据需要转置以适合最高效的访存方式，新函数名称以inatm_trans表示，循环分列前与分列后的代码结构示意如图7所示，此时从核使用率为99.3%。

```
subroutine rrtmg_lw（）
do iplon = 1, ncol
call inatm（variable(1:nlayers)）
call cldprmc（）
call setcoef（）
call taumol（）
call rtrnmc（）
enddo
end subroutine rrtmg_lw

subroutine  inatm（variable(1:nlayers)）
do k = 1, nlayers
f(x)=f(variable(:,k))
enddo
end subroutine inatm
……
```

```
subroutine rrtmg_lw（）
!$acc parallel [子句列表] collapse(2)
do iplon = 1, ncol
do k= 1, nlayers
call inatm_trans（variable(k)）
call cldprmc_trans（）
call setcoef_trans（）
call taumol_trans（）
call rtrnmc_trans（）
enddo
enddo
!$acc end parallel
end subroutine rrtmg_lw

subroutine inatm_trans（）
f(x)=f(variable(k))
end subroutine inatm_trans
……
```

图7 inatm函数循环分列前后结构示意图

Fig. 7 Diagrams before and behind the loop splitting

循环分列的目的是通过减少单个从核承担计算任务量的方式降低数据通信量，同时增大并行度且简

化程序结构，需要在一定程度上改写代码结构使其重组，合并，扩大最外层循环大小。整个rrtmg_lw函数改写合并循环后，同时并行的线程增大至5400个，每个从核的输入数据量为22×140×8=24640 byte，中间变量大小为7×140×8=7840 byte，总数据量已经低于局存大小，不再需要数据分块拷贝，重新调整其通讯策略后，运行速度得到进一步提升。

对于某些原本不适合进行从核并行的计算核心段需要循环改写后再进行从核并行，否则只能通过循环内移分块跳过，使其在主核完成计算过程，影响模式运行效率。例如计算气态光学厚度和普朗克量的taumol函数包含16个子函数对应lay变量的16种分割方式，不具备从核并行条件，通过合并成为一个最外层以lay为变量的大循环后，顺利在从核运行且使用率超过90%。

在实际情况下，方案1，2，3实现数据并行计算的不同思路通常需要结合使用。针对不同核心函数不同的计算结构与数据规模，分析设计出各自最高效的通讯策略。本次工作中主要选取了长波辐射部分的2个主函数和7个子函数作为核心段，最终实现 “粗粒度MPI并行+细粒度众核并行”多级异构并行方案。

3.2.4其他并行优化措施

使用ldm数学函数库：在OpenACC加速指导语句的基础上，使用ldm数学函数库，经测试，ldm数学函数精度与默认库保持一致，计算性能提升30%～80%不等，进一步提升了GRAPES系统整体运算效率。

3.2.5 结果与分析

试验采用分辨率为0.5°×0.5°的GRAPES全球模式长波辐射方案，积分步长为36，优化选项使用-O2。由于cldprmc、setcoef耗时较少，没有进行优化。加速后，对比纯主核运行输出的全球长波辐射通量与模式输出值，经验证误差，相对误差的量级在10^{-5}至10^{-5}之间，在允许范围内。优化结果对比如表1所示。

表1 众核加速结果

Table 1 The result of many-core acceleration

函数名称	主核运行时间/s	主从并行运行时间/s	主从并行加速比
inatm	30.48	5.3	5.8
taumol	20.97	5.99	3.5
rtrnmc	28.65	8.31	3.45

4 展望与不足

随着编译器的进一步优化和硬件技术的发展，OpenACC加速与CUDA等在底层技术实现上的差距将会越来越小，而且支持OpenACC加速指令转换将不仅仅针对CUDA设备，还包括其他更多厂商的硬件加速设备，从而能极大地提高OpenACC加速指令的普适性，增大程序可移植性。

文中以GRAPES全球模式RRTMG辐射模块长波过程为加速对象，目前只应用到为数不多的热点函数，就能得到较好的加速效果，优化还有很大的提升空间。

下一步将继续深入研究源代码数据结构，加强数据预处理，减少从核与主存之间的通讯频率，充分利用从核中高速缓存LDM，提升数据访问效率；同时尝试多种OpenACC加速指令组和使用，进一步提升加速效果。

参考文献

[1] 伍湘君, 金之雁, 黄丽萍, 等. GRAPES模式软件框架与实现. 应用气象学报, 2005, 16(4): 539-546.
[2] 陈德辉, 沈学顺. 新一代数值预报系统GRAPES研究进展. 应用气象学报, 2006, 17(6): 773-777.
[3] 曾文权, 胡玉贵, 何拥军, 等. 一种基于OPENACC的GPU加速实现高斯模糊算法. 计算机技术与发展, 2013, 23(7): 147-150.
[4] 何沧平. OpenACC并行编程实战. 北京: 机械工业出版社, 2016.
[5] 郭妙, 金之雁, 周斌. 基于通用图形处理器的GRAPES长波辐射并行方案. 应用气象学报, 2012, 23(3): 348-354.
[6] 郑芳, 许先斌, 向冬冬, 等. 基于GPU的GRAPES数值预报系统中RRTM模块的并行化研究. 计算机科学, 2012, 39(6A): 370-374.
[7] 荆现文, 张华. McICA 云一辐射方案在国家气候中心全球气候模式中的应用与评估. 大气科学, 2012, 36(5): 945-958.