

GRAPES 全球格点模式的并行计算负载平衡策略^{*†}

伍湘君^{1,2,3} 陈德辉^{2,3} 宋君强¹ 金之雁^{2,3} 杨学胜^{2,3} 张红亮^{2,3}

WU Xiangjun^{1,2,3} CHEN Dehui^{2,3} SONG Junqiang¹ JIN Zhiyan^{2,3} YANG Xuesheng^{2,3} ZHANG Hongliang^{2,3}

1. 国防科学技术大学计算机学院, 长沙, 410073

2. 中国气象局数值预报中心, 北京, 100081

3. 国家气象中心, 北京, 100081

1. *School of Computer Science, National University of Defense Technology, Changsha 410073, China*

2. *Center for Numerical Weather Prediction of CMA, Beijing 100081, China*

3. *National Meteorological Center of CMA, Beijing 100081, China*

2008-11-03 收稿, 2009-10-14 改回.

Wu Xiangjun, Chen Dehui, Song Junqiang, Jin Zhiyan, Yang Xuesheng, Zhang Hongliang. 2010. Parallelism of the GRAPSE global model and its loading balance strategy. *Acta Meteorologica Sinica*, 68(5):591-597

Abstract For the past 10 years, the peak floating point operation execution rate (floating point operations per second, or FLOPS) in parallel computers has increased by several orders of magnitude. Their large-scale applications in science rely on larger configuration of the supercomputers which comprises more than a hundred thousand of processors. Development of parallel software has been thought of as time and effort intensive. But the parallel algorithm design involves more than just using multiple processors, it also focuses on its efficiency and scalability. Setting load-balance tasks' assigning can commonly speedup the running. The GRAPES (Global and Regional Assimilation and PrEdiction System) global model is a semi-implicit semi-Lagrangian numerical prediction model formulated in spherical coordinates. For the parallelism of the GRAPES software system, there are two problems must to be solved, one is how to gather the data when computing the up-stream points near the south and north poles along Lagrangian trajectory, and another is how to set the values of periodic boundary latitudinally and of symmetry boundary longitudinally. Due to the convergence of the meridians, the longitudinal grid size decreases toward zero as the poles are approached. So the parallelism near the poles is a tough issue. How to get a better running time at the high performance computer system is a key-issue for the GRAPES software development. In this paper, a new method is proposed, which is based on group message passing of MPI. A more efficient load balance solution of assignment is also discussed. Experiments on the IBM-cluster 1600 of Chinese Meteorology Administration (CMA) showed that the parallel algorithm is scalable, efficient and portable. Its computing time cost can be acceptable for real-time weather forecast.

Key words GRAPES model, Parallel computing, Group message passing, Load balance, IBM-cluster1600

摘 要 随着高性能计算机技术的发展和应用, 并行计算已成为保证数值天气预报模式业务运行时效的关键技术之一。目前高性能计算机计算能力已达到每秒千万亿次浮点计算, 系统中处理器数量也早已达十万甚至更多, 如此巨大的计算资源对应用软件系统的设计也提出了挑战。数值天气预报软件系统要充分利用高性能计算机提供的计算资源, 必须依靠并行计算方法, 这包括适合计算问题的可扩展并行算法的设计、合适的数据分配方案以及良好的任务负载平衡方案。作为中国新一代数值天气预报格点模式, GRAPES(Global and Regional Assimilation and PrEdiction System)设计的最终目标是一个科研/业务通用, 区域/全球通用模式。作为一个格点模式, GRAPES 的并行计算具有与欧洲中期数值预报研究中心谱模式并行计算不同的特点, GRAPES 的并行计算采用了经典的水平网格数据划分。但对于全球的 GRAPES 模式, 由于采用拉格朗日差分

* 资助课题: 国家十一五科技支撑计划项目(2006BAC02B00)、国家自然科学基金(40575050, 40775073)项目。

作者简介: 伍湘君, 主要研究方向为数值天气预报软件系统并行计算。wuxj@cma.gov.cn

方案,模式极地及附近区域格点与格点之间距离的减小,使得模式并行计算在采用简单的经纬网格划分方式实现时,必须考虑极地地区并行计算跨越多个处理器时导致的频繁通讯解决途径。本研究提出了利用消息传递组通讯实现全球格点模式并行计算的一种方法,其核心思想是将极点附近一定区域内的处理器按纬向划归不同的处理器组。文中还给出了该实现方法的任务分配算法,提出了改进的任务分配负载均衡方案。在中国气象局高性能计算机 IBM-cluster1600 上的测试表明,算法具有较好的可扩展性,其负载均衡方案改善了计算的绝对墙钟时间,使并行计算效率提高 10% 以上。模式的准业务运行结果表明计算墙钟时间基本可以满足数值预报业务的实时性要求。

关键词 GRAPES 模式, 并行计算, 消息传递组通讯, 负载均衡, IBM-cluster1600

中图法分类号 P409

1 引言

以 GRAPES 模式系统为基础架构,建立以全球数值天气预报为核心的中国气象局数值预报业务体系,是中国数值天气预报未来发展的一个总体目标。面向高性能计算机的并行计算高效算法是保证预报系统业务实时可用的关键技术之一。高性能并行计算机系统的发展为并行计算机应用提供了日益强大的计算平台和研究环境。随着高性能计算机规模的增大,数值天气预报可用计算机资源得到了极大提升。如何充分有效利用如此巨大的处理器资源,成为数值预报并行计算研究面对的一个极富挑战性的问题。

按照计算方法的不同,数值天气预报模式可分为谱模式和格点模式。通常谱模式的并行计算是将处理器拓扑视为矩阵形式,谱空间、傅氏空间和格点空间分别沿数据无关方向进行剖分,在不同计算空间通过数据的并行二维转置策略实现数据交换,并最终完成模式的并行计算。对于一般的区域格点模式,在计算区域一定时,其并行计算区域分解典型的做法是采用经纬网格分区,其分区可以是规则的,也可以是不规则的,视区域计算负载而设计。而对于全球格点模式的并行计算,随着极地区域格点之间距离的减小,其并行处理方案是一个必须考虑的问题,目前的方法中极地部分采用一个处理器计算以屏蔽通讯是一个常用的办法。

作为全球格点模式,GRAPES 在积分方案上使用了两个时间层的半隐式半拉格朗日方案。由于 GRAPES 全球模式极区对半拉格朗日上游点的寻找、东西向周期边界条件和南北极区对称边界条件设置的特殊性,其并行计算效率的高低将取决于并行方案设计、并行实现方法以及负载均衡方案等。因此,在实现 GRAPES 全球模式并行计算时,必须同时满足并行计算的正确、高效、可扩展以及任务的

负载均衡。

2 GRAPES 全球格点模式计算简介

GRAPES 全球模式动力框架使用一套含完全动力学过程的非静力、全可压的方程组,水平方向上采用经纬度球面坐标,垂直方向采用高度地形追随坐标,积分方案使用两时间层的半隐式半拉格朗日方案, v 放在极地的水平 Arakawa C 跳点格式,垂直方向为 Charney-Philips 变量跳层分布,使用广义共轭余差(GCR)方法求解关于气压扰动的三维 Helmholtz 方程等。模式预报变量包括扰动气压 Π 、扰动位温 θ 、水平风 u 和 v ,垂直速度 $\hat{w}$ 和水汽 q 。模式的变量预报方程组(详细方案及各变量的含义请见薛纪善等(2008))为

$$u = [\zeta_{u_1} \frac{1}{a \cos \varphi} \frac{\partial}{\partial \lambda} + \zeta_{u_2} \frac{1}{a} \frac{\partial}{\partial \varphi} + \zeta_{u_3} \frac{\partial}{\partial \varepsilon}] \Pi' + \zeta_{u_0} \quad (1)$$

$$v = [\zeta_{v_1} \frac{1}{a \cos \varphi} \frac{\partial}{\partial \lambda} + \zeta_{v_2} \frac{1}{a} \frac{\partial}{\partial \varphi} + \zeta_{v_3} \frac{\partial}{\partial \varepsilon}] \Pi' + \zeta_{v_0} \quad (2)$$

$$\hat{w} = [\zeta_{w_1} \frac{1}{a \cos \varphi} \frac{\partial}{\partial \lambda} + \zeta_{w_2} \frac{1}{a} \frac{\partial}{\partial \varphi} + \zeta_{w_3} \frac{\partial}{\partial \varepsilon}] \Pi' + \zeta_{w_0} \quad (3)$$

$$\theta' = \zeta_{\theta_3} \frac{\partial}{\partial \varepsilon} \Pi' + \zeta_{\theta_0} \quad (4)$$

$$\Pi' = \zeta_{\Pi_1} u + \zeta_{\Pi_2} v + \zeta_{\Pi_3} \hat{w} + \zeta_{\Pi_4} (D_3)_z + A_{\Pi} \quad (5)$$

为了从程序软件框架上实现模式的并行计算方案,实现分布式并行计算和共享存储并行计算的统一,GRAPES 模式采用了 3 层软件框架结构(图 1),最上层是驱动层,中间界面过渡为中间层,最下层是模式层,也即核心计算层。驱动层的设计主要为处理与计算环境、计算数据有关的内容,如与计算机相关的数据长度的设置、控制变量的初始化、计算变量的空间分配、初始数据的输入处理等,有关并行计算的区域分解也在这一层完成。软件框架的模式层是主要的科学计算部分,完成一个积分循环的所有计算,如动力框架的方程线性和非线性项的计算、半拉格朗日差分方案中上游点的查找、大型矩阵的

Helmholts 方程求解等,以及当前步上选用的物理过程计算如辐射、陆面、对流层、微物理等,全球模式中还包括对极地的处理,如极地滤波和极地点变量的设置等。在驱动层和模式层之间是软件框架的中间层,中间层主要完成数据流的传输,控制积分循环计算以及一些历史资料的输出。

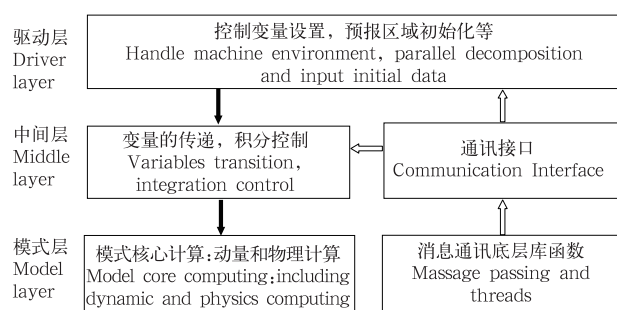


图1 GRAPES 模式的3层软件框架结构

(空心箭头表示支持, 实心箭头表示数据流)

Fig. 1 Three-level hierarchy superimposed over the GRAPES model (solid-arrow denotes “support”, virtual-arrow denotes data stream)

GRAPES 模式采用了不同于一般欧拉方法的拉格朗日方法进行空间离散计算。对于欧拉方法而言, 当一个数值模式的空间差分格式给定后, 参与模式变量空间离散计算的上下、左右、前后格点位置、格点数就确定了, 这使得并行计算方案及负载均衡比较容易实现; 而对于拉格朗日方法而言, 实现起来就要困难得多。因为拉格朗日的空间离散计算必须

确定质点的运动轨迹, 确定空气质点上一时刻的出发点(即拉格朗日上游点), 而“上游点”又取决于质点的运动速度, 速度越大, 上游点离当前点就越远, 而且通常情况下上游点并不落在整数格点上。这时参与拉格朗日空间离散计算的格点位置、格点数都是不确定的。因此, 在实现 GRAPES 全球模式的并行计算时, 有两个问题需要考虑, 即: 极点区域在格点距离急剧减小时上游点的计算以及全球模式数据边界的处理。其中极地并行计算效率的好坏决定了整个模式系统并行计算的效率。

3 GRAPES 全球格点模式并行计算区域分解方案及实现方法

数值模式的计算实际上就是一个积分循环的过程。计算由数据读入开始, 到结果输出为止, 并行处理主要采用数据并行方式, 其流程可简化如图2所示。

并行计算首先将输入的初始数据按照某种分配策略分发到各个处理器上, 各处理器分别针对各自得到的数据块进行动力积分和物理过程计算, 其中处理器之间在需要数据同步的地方进行必要的信息交换, 在需要结果输出和整个积分计算结束时完成对所有处理器上数据的收集。对于格点模式, 数据分配经典的做法是采用水平区域分解, 即经纬水平网格分解, 图3是16个处理器水平拓扑分布的一个示意图。

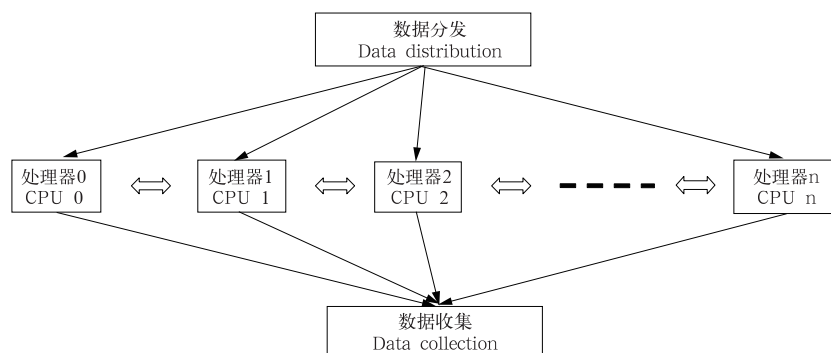


图2 数值模式并行计算的一般流程

Fig. 2 A typical flowchart for numerical weather prediction parallel computing

对于 GRPAES 全球模式并行计算而言, 需要考虑的第一个问题是极地及附近区域求上游点的问题。GRPAES 全球模式采用的是 Arakawa-C 格

点, 从其变量水平分布(图4)可见, 随着向极地逼近, 格点与格点之间的距离越来越短, 直至到达极点时距离为零。

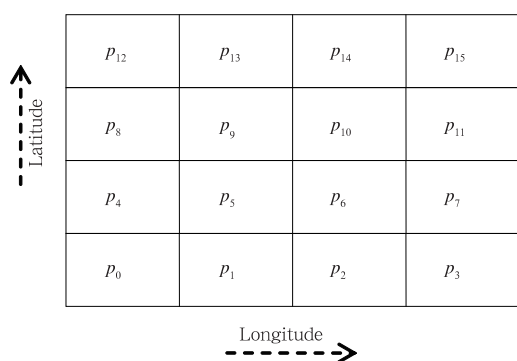


图3 格点模式并行计算的经纬水平分解示意(p_i 指处理器)

Fig. 3 The schematic structure of 2D latitude-longitude partition (p_i denotes the i th process)

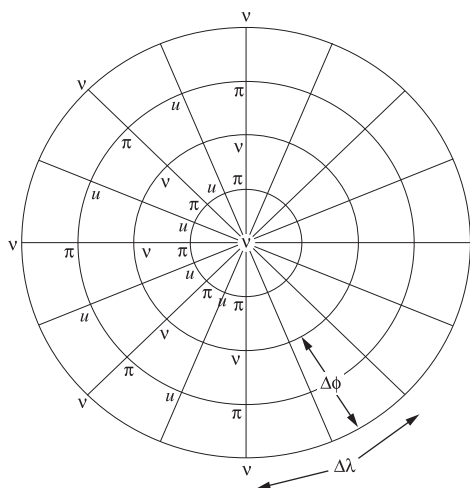


图4 Arakawa-C 格式的模式变量水平分布
Fig. 4 Variable layout on Arakawa-C for the GRAPES model

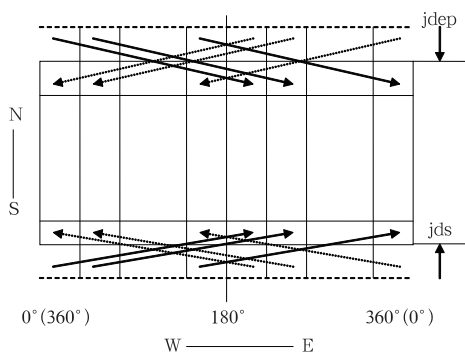


图5 GRAPES 全球模式南北对称边界设置
(实线表示模式计算格点, 上下两条虚线分别表示南北边界点, 带箭头线示意边界点与模式点的对应关系。字母“N, S, W, E”分别表示“北, 南, 西, 东”)
Fig. 5 Symmetric boundary setting for the GRAPES-Global model (solid lines depict the grid-points and the upper and lower dashed lines denote the north and south boundaries, respectively)

由于模式采用半隐式半拉格朗日差分方案, 在计算某一时次格点值时需要知道其上游点的情况。以一维质点平流为例, 从拉格朗日的角观看质点运动, 假设气团以速度 U 作位移运动, 且满足

$$\frac{d\phi}{dt} = 0 \quad (6)$$

在 $t + \Delta t$ 时刻、某一物理量 $\phi(x, t + \Delta t)$ 在某格点 (x) 的值等于“ t ”时刻该物理量在上游点的值 $\phi(x - \delta x', t)$, 即

$$\phi(x, t + \Delta t) = \phi(x - \delta x', t) \quad (7)$$

那么, 只要知道 $\delta x'$ (起始点位置与当前点的跨度), 就可以通过“ t ”时刻的格点值求得“ $t + \Delta t$ ”时刻格点值。通常, $\delta x'$ 的长度可能是若干个格点位置, 这取决于气团的速度 U 。如果 U 较大, 或 Δt 较大, 都可能导致 $\delta x'$ 较大。而如果计算又在极地附近, 则在普通水平分区并行计算时, 随着极地格点间距离越来越小, 上游点极可能跨出若干处理器, 甚至翻越极点。将在极点处的径向风分量视做带方向的风矢量, 将图4沿经线展开(图5), 则极点处某个模式格点的 v 风矢量指向该格点对应的半球上以极点为中心对称的格点。过极地的三维矢量场上游点跨越极地时同样看作是对称半球上某个格点。

GRAPES 全球格点模式并行计算需要解决的第2个问题是边界数据的设置, 这包括南北对称边界(图5)和东西周期边界(图6)的设置。特别是对称边界的设置, 必须得到对称半球处理器模式格点的值。

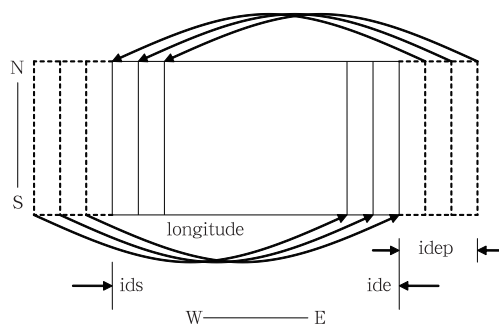


图6 GRAPES 全球模式东西周期边界设置
(实线表示模式计算格点, 东西两侧虚线分别表示东西边界点, 带箭头线示意边界点与模式点的对应关系。字母“W, E”分别表示“西, 东”)
Fig. 6 Periodic boundary setting for the GRAPES-Global model (solid lines depict the grid-points and the eastern and western side dashed lines denote the east and west boundaries, respectively)

在 GRAPES 全球模式中,当模式时间步长较长时,在高纬度易产生计算不稳定现象。在缩短时间步长的前提下,GRAPES 采用极地滤波方案去掉不稳定的波分量。由此亦可见,在 GRAPES 全球模式并行计算中,极地处理显得尤为重要。

对于全球模式的数据分发策略,一个简单的方案是采用纬向分区。纬向分区可以屏蔽极点计算纬向风上游点过远的问题,但此方案可用的处理器数目受限于计算的纬圈数,因此其并行计算的可扩展性将受到影响。另一种并行区域划分方案是在经纬网格分区基础上将南北两个极点附近一定范围区域分在同一个处理器上,形成极点通区。这种方案虽然可以屏蔽极点计算纬向风的问题,但由于极点区域始终只有一个处理器参与计算,因而可能导致计算负载的不平衡。

为了解决极点计算特殊性导致的并行数据分配问题,GRAPES 全球模式并行计算在标准经纬网格划分基础上,将南北极地及附近一定区域内处理器按处理器矩阵行设置为组,提供处理器行组内的通讯函数,其目的就是为半拉格朗日上游点的查找提供更宽的“halo”区域(即处理器计算区之外相互间内存重叠部分。如图 7 中阴影部分分别表示南北极地区域的处理器组)。在每个通讯组内部,提供由单处理器空间到整个行空间的数据收集函数,并允许进行通讯组内部处理器之间的相关操作。理论上说,所有的处理器组都可以进行组通讯,但这将会大大增加通讯量。因此,根据经验可以给定一个临界值 $polar_line$,覆盖这个值至极点区域的处理器设组。

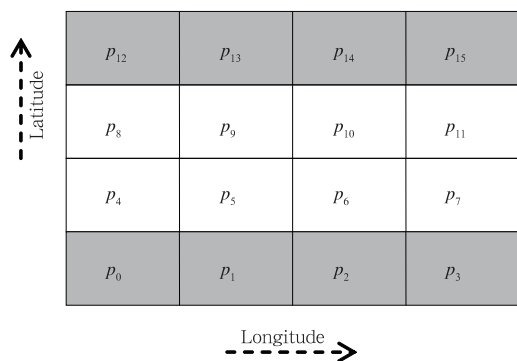


图 7 GRAPES 全球模式并行计算极地附近区域分解方案(p_i 指处理器)

Fig. 7 Cap-decomposition scheme near the poles for the GRAPES-global model (p_i denotes the i_{th} processor)

针对设有处理器组的区域和其他计算区域分别提供插值函数,其区别就在于设有处理器组的函数输入变量为整个纬圈范围,亦即计算式的右边变量可扩展到整个纬向域,但计算式左边变量的区域仍然是网格划分区域,以保证计算负载的平衡。

在水平经纬网格划分基础上,将处理器矩阵拓扑按行设组(图 7),其中以阴影区域表示极地处理器组,其他表示普通处理器。极地处理器组允许处理器间数据进行组内通讯。

4 GRAPES 全球并行模式的任务分配负载平衡策略

目前中国气象局数值模式业务上使用的高性能计算机为 IBM-cluster1600,这是一个分布式内存并行计算机,整个机器系统由两个集群组合而成,其中的研发集群包括 256 个节点,每个节点含 8 个 Power4++ 共享存储处理器,通过 HPS Switch 连接,每个处理器含 2G 内存。

将 GRAPES 全球模式的并程序在 IBM-cluster1600 上进行试算,测试选定个例格点规模为 $720 \times 361 \times 31$ 、时间积分步长 600 s、全物理过程、使用 16 个节点 128 个处理器(每个节点上 8 个处理器)。将这 128 个处理器的拓扑结构设置为 2×64 ,即径向分 2 组、每组纬向分 64 个处理器,在 $polar_line$ 取 $+(-)70^\circ$ 时,处理器组 $(0,1) \cdots (12,13)$ 为南极地组, $(112,113) \cdots (126,127)$ 为北极地组。

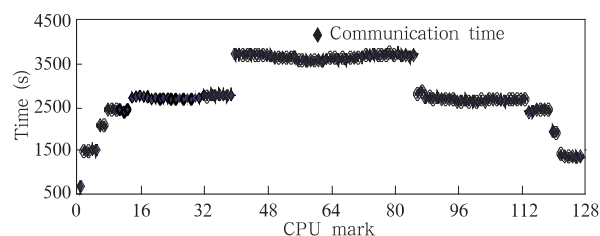


图 8 不做负载平衡处理的通信时间分布
Fig. 8 Communication time of each process when the original load scheme used

一次测试中各个处理器的通讯时间分布(图 8)中,通讯时间指数据通讯开始至结束的整个时间,其中包括处理器等待时间(下同)。由于南北方向上任务分配两头多中间少,导致极地处理器计算较慢,从而使得中间处理器等待时间较长、通讯时间增加。

从图 8 中可见,处理器通讯时间呈现两头小中间大的特点,这与我们的极地处理方案是不协调的。如上节所述,在极地处理器组中由于行处理器间需要通讯,因而极地组应占有更多的实际通讯时间。但事实上事与愿违。分析各处理器上不同 MPI 函数所消耗的时间,可以发现中间处理器耗费了大量的通讯等待时间,这说明极地处理器组计算任务较多而中间处理器组任务相对较少。由此可见原来的任务分配方案应该有不合理的地方。

GRAPES 全球模式的任务分配采用简单均分方案,其分别在径向、纬向采用一维分配。假设维长为 L ,本维处理器个数 n ,则原始任务分配(图 9a)中“o”表示 L/n ,”+”表示不能除尽时 $L/n + 1$ 。这种分配导致纬向处理器在极地及附近区域分配的任务可能偏多,显然这不利于全球极地计算的负载平衡。调整任务分配方案(图 9b)中“o”、“+”含义同前,“-”表示在 L/n 的基础上减少一定的任务量,如在纬向上减少 1 个纬圈、2 个纬圈甚至 3 个纬圈。

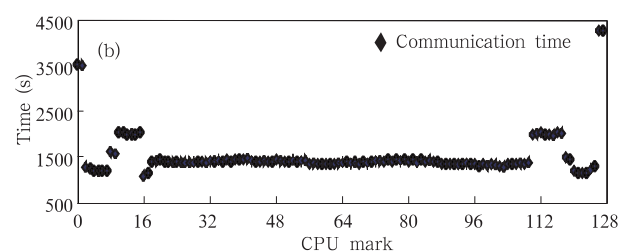
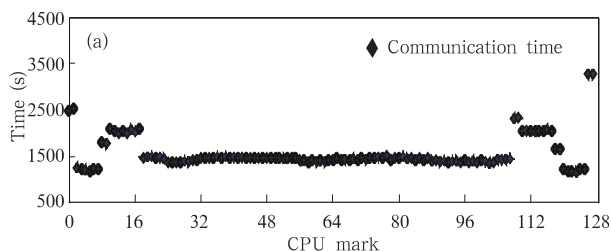


图 10 采用极地任务减少 2 个纬圈(a)、3 个纬圈(b)的通信时间分布

Fig. 10 Communication time of each process in the case of the polar tasks reduced by (a) two latitude-tasks and (b) three latitude-tasks

表 1 3 种计算方案的墙钟和通讯时间统计表(单位:s)

Table 1 Cost times for the three schemes (unit:s)

	计算墙钟时间	平均通讯时间	比率(%)
原方案	9371.0	2851.0	30.4
减少 2 纬圈方案	7956.0	1444.0	18.1
减少 3 纬圈方案	7929.0	1412.0	17.8

采用优化的任务分配负载平衡方案,在 IBM-cluster1600 上使用 128 个处理器、极地任务减少 3 圈、不同处理器矩阵拓扑结构得到的 10 d 预报模式计算墙钟时间(表 2)中, $n \times m$ 表示径向处理器数为 n 、纬向处理器数为 m 。计算针对如上面一样的测试个例。由于内存不足,单向处理器分布、即处理器拓扑

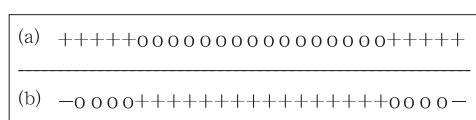


图 9 一维情况下任务分配方案的比较

(a. 为原始分配方案,b. 为改进分配方案)

Fig. 9 Comparison of task-distribution for processors in the one dimensional case between the (a)original scheme of task-distribution and (b) the new scheme of task-distribution

通过减少极地区域处理器任务量平衡各处理器计算任务,使得中间更多处理器通讯等待时间降低,从而使计算总时间得到减少。

从极地处理器任务减少 2 个纬圈(图 10a)和减少 3 个纬圈(图 10b)、使用 128 个处理器、仍然采用 2×64 个处理器矩阵时的通讯时间分布(图 10)可见,减少极地处理器组任务使得中间处理器通讯时间大大降低,而南北两个极点处理器通讯时间相对增加,但通讯的平均值有所降低,而且由于极点处理器任务数减少较多,计算总的时间大大降低(表 1)。

为 1×128 时计算不能正常结束。从表中可以发现,在相同数目处理器下,径向分配处理器数越少使得参与极地组通讯的处理器也越少,因而计算速度越好。

表 2 128 个处理器在不同矩阵结构下的计算时间统计(单位:s)

Table 2 Elapsed times of the GRAPES under the different processor topologic structures for 128 CPUs (unit:s)

矩阵拓扑	2×64	4×32	8×16	16×8	32×4
计算墙钟时间	7695.0	8160.0	9296.0	11032.0	15445.0

比较在 IBM-cluster1600 上取不同数目处理器、在最优矩阵拓扑结构下、10 d 预报原始任务分配方案与优化分配方案的计算墙钟时间以及各自的相对加速比(表 3)中,新方案选用极地任务减少 3

圈。由于在使用 2×16 时内存不足,32 个 CPU 只能采用 4×8 的矩阵拓扑。同样,由于纬向每个处理器上分配的任务数不能小于 3,256 和 512 采用的矩阵拓扑分别取 4×64 和 8×64 。从表 3 中可以看

到,优化的任务分配方案与原始方案相比计算墙钟时间得到降低,而计算加速比和计算效率均有所提高,如当使用 512 个处理器时,计算效率由原来的 40% 提高到 50% 以上。

表 3 使用不同个数处理器的并行计算时间统计(单位:s)

Table 3 Elapsed times with different number of processors and different load-balance scheme used (time unit:s)

矩阵拓扑	原任务方案			新任务方案		
	墙钟时间	加速	效率	墙钟时间	加速	效率
32(4×8)	27373.0	1.0	1.0	26918.0	1.0	1.0
64(2×32)	16093.0	1.70	0.85	14352.0	1.88	0.94
128(2×64)	9146.0	2.99	0.748	7695.0	3.50	0.875
256(4×64)	5499.0	4.98	0.63	4451.0	6.05	0.756
512(8×64)	4080.0	6.71	0.419	3143.0	8.56	0.535

5 结论与讨论

GRAPES 全球格点模式由于极点及附近区域计算的特殊性,其并行计算数据分发策略必须要求考虑到跨多个格点插值的情况,本文提出了利用消息传递组通讯方法实现格点模式并行计算的一种方法,巧妙地解决了格点模式并行计算的极地处理问题,与其他并行计算实现方案相比,在并行可扩展性和并行计算负载均衡性上具有较大优势。同时通过调整处理器之间的任务分配方案改善计算负载均衡,使并行计算的效率得到提高。

并行计算使 GRAPES 模式的计算墙钟时间有了极大改善,保证了 GRAPES 系统业务的实时运行,同时也为模式向更高分辨率和更精细过程计算提供了可能,为 GRAPES 系统的升级和下一步工作打下良好的基础。随着模式分辨率和计算机性能的提高,并行计算 CPU 将成百上千地增加,因此进一步的工作将围绕模式核心计算方法等进行优化,以满足精细预报下业务实时预报要求。

目前 GRAPES 全球并行模式版本已进入准业务运行应用阶段,水平分辨率为 0.5° 的全球准业务并行版本计算稳定、有效、可靠,基本能够满足业务预报实时性要求,为其业务运行奠定了良好的基础。

致谢:感谢沈学顺博士对模式上游点计算的详细说明,同时感谢李兴良博士提供的模式变量分布图绘制。

参考文献

陈德辉,沈学顺. 2006. 新一代数值预报系统 GRAPES 研究进展. 应

用气象学报,17(6): 773-777
陈德辉,薛纪善,杨学胜等. 2008. GRAPES 新一代全球/区域多尺度统一数值预报模式总体设计研究. 科学通报, 53(20):2396-2407
沈桐立等. 2003. 数值天气预报(修订版). 北京:气象出版社,471pp
伍湘君,金之雁等. 2007. 新一代数值预报模式 GRAPES 的并行计算方案设计与实现. 计算机研究与发展,44(3):510-515
薛纪善. 2006. 新世纪初我国数值天气预报的科技创新研究. 应用气象学报,17(5):602-610
薛纪善,陈德辉. 2008. 数值预报系统 GRAPES 的科学设计与应用. 北京:科学出版社,383pp
Ananth Grama, Anshul Gupta, George Karypis, et al. 2003. Introduction to parallel computing Second Edition. 北京:机械工业出版社,(影印版), 636pp
Deborah Salmond, Sami Saarinen. 2004. Early Experiences with the IBM p690 + at ECMWF//Proceedings of the Eleventh ECMWF Workshop on the Use of Parallel Processors in Meteorology. World Scientific, 1-12
Dent D, Mozdzyński G. 1996. ECMWF Operational Forecasting on a Distributed Memory Platform; Forecast Model//Proceedings of the Seventh ECMWF Workshop on the Use of Parallel Processors in Meteorology. World Scientific,36-51
John Hague. 2002. Optimisation of ecmwf's Integrated forecast system(IFS) for the IBM p690//Proceedings of the Tenth ECMWF Workshop on the Use of Parallel Processors in Meteorology. World Scientific, 294-303
Michalakes J, Dudhia J, Gill D, et al. 2004. The Weather Research and Forecast Model; Software Architecture and Performance//Proceedings of the Eleventh ECMWF Workshop on the Use of Parallel Processors in Meteorology. World Scientific,156-168