

文章编号:1673-9469(2015)02-0077-05

doi:10.3969/j.issn.1673-9469.2015.02.019

水质预报系统的 MPI + OpenMP 并行优化研究

李 春

(河北工程大学 信息与电气工程学院,河北 邯郸 056038)

摘要:OpenMP(Open Multi-Processing)能够在共享存储环境中获得理想的并行性能,MPI + OpenMP 混合同行编程模型更贴近于多核心集群的体系结构。在多核心集群上进行实验,混合模型相对串行程序可缩短95%的运算时间,加速比超过20。实验结果表明,该混合模型比纯 MPI 并行的系统具有更优的加速比和扩展性,是对纯 MPI 并行系统的一种改进。

关键词:多核心集群;水质预报系统;消息传递接口;开放式多处理;混合同行编程模型

中图分类号:TP302

文献标识码:A

Research on MPI + OpenMP parallel optimization of water quality forecast system

LI Chun

(School of Information and Electrical Engineering, Hebei University of Engineering, Hebei Handan 056038, China)

Abstract:OpenMP(Open Multi-Processing) can achieve an ideal parallel performance on shared storage system. MPI + OpenMP hybrid parallel programming model is closer to the architecture of multi-core clusters. The hybrid model on the multi-core cluster was executed. The test results showed that the hybrid parallel programming model could reduce 95% of the run time of the serial program. And its parallel speed-up ratio was more than 20. This paper shows the hybrid parallel model is an improvement of the pure MPI parallel system with better speed-up ratio and scalability.

Key words:multi-core cluster; water quality forecast system; MPI; OpenMP; hybrid parallel programming model

随着海洋各方面数据的完善及空间分辨率的提高,在使得预报更加准确的同时,预报系统的计算量也越加庞大。在高性能计算机上对预报系统进行并行开发是满足研究人员对计算效率需求的主要方法。目前国内外已有许多基于 MPI 或 OpenMP 等并行技术的应用和研究,但由于海洋生态动力学的复杂性,基于二者的混合同行技术应用用于海洋生态预报上的研究还非常少。

多核心集群系统的分布-共享体系结构使其既适合以 MPI 为代表的分布式存储并行计算技术,也适合以 OpenMP 为代表的共享存储并行计算技术,又可使用两种方法结合的混合同行计算^[1-3]。以胶州湾水质预报系统为例,胶州湾水质预报系统是对胶州湾水质进行监测的短期预报

系统,其纯 MPI 非阻塞并行程序比聚合通信方式并行版本具有更好的加速比和并行效率,但非阻塞通信对系统内存开销较大^[4]。根据系统的串行程序的特点,在其 MPI 并行的基础上,继续进行 OpenMP 并行,对其纯 MPI 并行模型进行优化,形成 MPI + OpenMP 混合同行编程模型,可以达到更好的并行效率和加速比。

1 MPI + OpenMP 混合同行实现原理

并行程序的实现需要高性能计算机和相应的并行计算软件开发运行环境。由于多核心处理器具有高性能低功耗的特点,装备了多核心处理器的集群系统也在保持了较低功耗的同时大大增加了集群的计算核心数量,其较高的性价比使得多

收稿日期:

基金项目:河北省自然科学基金项目(F2015402077)

特约专稿

作者简介:李春(1987-),女,河北威县人,硕士,从事并行计算研究。

核心集群在高性能领域占据的比重越来越大。多核心集群节点内的计算核心共享相同的全局内存,具有对称多处理系统的共享存储特点。同时,集群的不同节点间具有相互独立的内存,并通过网络连接起来,这样又具有分布式存储系统的特点,这就构成了多核心集群的分布-共享存储结构。MPI 应用于分布式存储并行系统,用于开发基于消息传递的并程序。MPI 函数库有消息发送和接收、同步、广播等在内的多种功能,可以在分布的节点间通过网络互传信息。OpenMP 是一种应用于共享存储并行计算机系统的并行编程规范,提供在共享存储计算机平台上进行编程所需要的并行、同步等功能,是基于线程的并行编程模型。在共享存储结构中,多个处理器通过总线等方式与物理内存相连接,各处理器共享相同的内存空间,使得多个 OpenMP 线程可以访问相同的一块内存空间,避免了线程之间的数据交换。虽然 MPI 也可移植到共享存储的体系结构中,但在共享存储结构中也需将内存划分成与所申请处理器个数等份的份数,再按照分布存储进行处理^[5]。因此在共享存储并行时一般采用 OpenMP,而不是 MPI,以减少时间和内存的开销。MPI 与 OpenMP 都支持多种程序设计语言,如 Fortran 的多个版本和 C、C++ 等,以及主流编译器如 Intel Compiler 等也支持这两种并行技术。以上为实现 MPI + OpenMP 混合并行提供了硬件和软件支持。

由于通信、延迟等问题,MPI 的并行往往需要大的代码粒度,通过消息传递并行在多核心集群的各节点之间,但无法有效地利用节点内部的多核心处理器的计算能力。此时,将 OpenMP 在节点内部进行细粒度并行,既能将多核心处理器的内存和计算核心充分利用起来,又可使程序并行更加充分,达到更好的并行粒度。因此,混合模型能够有效挖掘计算机的处理能力,充分利用多核心集群的硬件资源,从而获得较高的性能。

2MPI + OpenMP 混合模型实现

2.1 水质预报系统串行与纯 MPI 程序分析

为全面考虑外海、大气以及陆源排放对海洋环境的影响,胶州湾示范区的模式主要由嵌套海洋水动力、海洋生态模式、大气模式和陆域面源模式等耦合而成,其示意图几个部分作用如图 1 所示。其中的海洋生态模式就是胶州湾的海洋生态

模式,海洋生态模式包含胶州湾小区的水动力模块和水质模块。

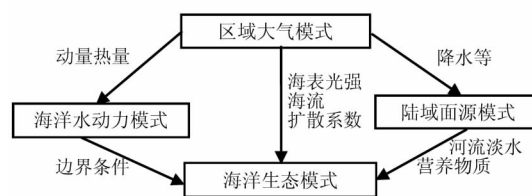


图1 胶州湾示范区水质预报模型

Fig.1 Demonstration area's water quality prediction model of Jiaozhou Bay

由图 1 可看出胶州湾水质预报系统的主要计算部分在其海洋生态模块,而整个系统最终输出的监测数值在海洋生态模块的水质模块部分,水质模块的计算量占整个计算系统的一半以上,水动力模块为水质模块的计算提供所需的边界条件。水质模块首先经过浮游植物光合作用、浮游动物捕食、浮游动植物呼吸死亡矿化等内部过程的四级生态物质变化过程,所有营养物质在四级过程中都参与计算,但后一级过程是建立在前一级过程的基础上进行计算的。考虑径流、流域面源输入、大气沉降、沉积物-水界面物质通量、与外海物质交换等外部过程,然后对包括氮(NH_4^+ 、 NO_3^- 、 NO_2^-)、磷(PO_4^-)、硅(Si)、溶解氧(DO)、化学需氧量(COD)、浮游植物(PHY)、浮游动物(ZOO)、水体碎屑(DET)、总氮(TN)、总磷(TP)等 12 个变量进行检测^[6],其中要进行水平扩散垂向扩散方向计算、海底通量计算、化学变化和开边界条件计算等,最后计算径流量。

胶州湾水质预报系统是基于空间差分网格中的 C 网格建立的,C 网格应用的是三维立体坐标系。系统中把监测区域划分为 $i * j * k$ 个网格点进行计算,根据预报目标海域的水平分辨率及嵌套水动力部分计算出,适合胶州湾的实际网格点数为 $159 * 185 * 5$,用 U 、 V 、 ω 分别表示水平 x 和 y 方向及垂向 z 方向的水平速度和垂向速度,其中每一个点的浓度值都受上下左右速度的影响,其示意图如图 2。

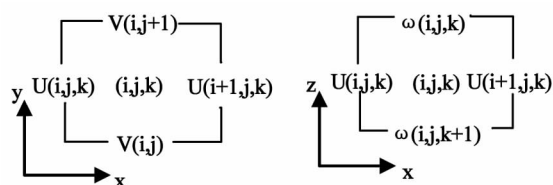


图2 三维C网格平面示意图

Fig.2 The plane schematic diagram of three dimensional C grid

胶州湾水质预报系统采用 Fortran 语言编写,其数据遵循列优先存储,因此,为减少数据传递的通信量从而减少数据传递所消耗的时间,在 MPI 并行设计中选择 i, j, k 三个方向的 j 方向划分任务区域。若向集群申请 n 个节点,那么计算任务将会被划分成 n 个子任务,每个子任务每一层包含 j/n 列行整数网格点,然后就会是 n 个进程分别对各子任务进行计算。

由于 MPI 并行系统采用非阻塞通信的方式,初始化完成后,划分每个进程所要计算的区域,然后对水质模块计算元素浓度所需的相邻区域的边界值进行传递,之后是水动力模块,这样就可以在后台程序进行数据传递的过程中,可在前台同时进行水动力方面的计算,为水质模块计算做好准备。等待所有数据传输完毕,进入水质部分的计算并输出结果。最后在程序完全结束后,释放并行环境。

但是,非阻塞通信模式在使通信和计算最大化同步的同时,存储消息的缓冲区也随之增大,这无形中也使得系统的开销加大。因此,对于计算量庞大的胶州湾水质预报系统的纯 MPI 非阻塞通信并行版本在多核心集群上运行时,没有最大化利用多核心集群的分布共享混合体系结构特点,不能充分发挥其计算性能,其较高的内存开销在一定程度上降低了系统的可扩展性。

2.2 混合并行模型系统设计实现

将 MPI + OpenMP 混合并行计算技术运用到胶州湾水质预报系统的水质模块,是对其纯 MPI 版本的一种优化。MPI + OpenMP 混合编程模型通常采用层次结构,即 MPI 并行在集群的上层 - 具有分布存储体系结构的各节点之间 - 用于粗粒度并行,OpenMP 并行于下层 - 节点内部具有共享内存体系结构的多核处理器的各核之间 - 提供轻量级线程,用于细粒度并行。MPI + OpenMP 混合编程模型易于实现,通过减少 MPI 进程数,增加 OpenMP 线程数,避免出现较大的负载不平衡和通信开销,比单纯用 MPI 编程具有更好的性能和更低的代价,实现更好的并行粒度。MPI + OpenMP 混合模型并行执行示意图如图 3。

对于 fortran 语言版本的胶州湾水质预报系统,其规模为 $159 \times 185 \times 5$,根据其列优先的数据存储特点,从 j 方向划分任务,各进程之间数据传递量会相对较少,因此,设计 MPI 并行在列,每个

进程计算其列的一个子区间,其行保有完整性,所以用 OpenMP 的“! \$OMP PARALLEL DO”结构在数据存储较为间断的 i 方向进行线程级并行,利用 OpenMP 线程共享内存的特点避免了数据的传递。而 k 方向只有 5 层,不对其进行划分。OpenMP 作为一种适用于共享存储平台的并行计算技术,能够有效利用节点计算机的多核心处理器进行线程级别的并行,使多个并行的线程共享访问分块边界处具有依赖关系的数据,从而降低消息传递带来的效率影响^[7]。

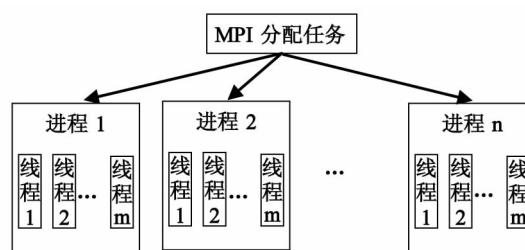


图3 MPI+OpenMP混合模型并行示意图

Fig.3 The schematic diagram of MPI+OpenMP hybrid parallel model

设申请 $size$ 个节点,每个节点使用 $threads_n$ 个处理器核,则系统将会被分成 $size$ 个进程,每个进程计算 $185/size$ 列行整面,每个进程中并行 $threads_n$ 个线程,又将行划分成 $threads_n$ 份,每个线程计算 $159/thread_n$ 行。声明 MPI 各进程并行区间的开始值和结束值分别为 $jbegin_col$ 和 $jend_col$,则

$$jbegin_col = myid * \text{int}(jm/size)$$

$$jend_col = (myid + 1) * \text{int}(jm/size) + 1$$

其中 $myid$ 为进程号, jm 为网格列数, $size$ 为进程数。

声明 OpenMP 并行的线程数量为全局变量 $thread_n$,以便在其他需要 OpenMP 并行的相应子函数中定义全局变量 X 后,通过 `omp_set_num_threads(X)` 语句进行设定线程数量,以后通过修改主函数中 $thread_n$ 的值来控制各函数 OpenMP 并行的线程数。

在下层并行的 OpenMP 线程数量可以根据运算需要及集群节点的核心数量进行一定的调整,其最大的线程数量不要超过节点计算机处理器核心的数量^[8],例如,集群节点的计算核心为 8 个,那么设置 OpenMP 线程数量的范围就在 1 ~ 8 之间。

以一级生态过程为例,其混合并行部分代码如下:

```

.....
do k = 1, kbm1
Loop_J: do j = jbegin_col, jend_col
! $ OMP PARALLEL DO
Loop_I; do i = 1, im
Calculation: if( wetmask(i,j) = 1. ) then
.....
End if calculation
Enddo Loop_I
! $ OMP END PARALLEL DO
Enddo Loop_J
Enddo
.....

```

MPI + OpenMP 混合并行胶州湾水质预报系统水质模块模型中,上层并行的每一个 MPI 进程内部包含多个 OpenMP 线程用于下层的线程级并行处理,多个并行的线程分别对不同的数据分块进行运算求解并可以直接访问相邻分块的边界数据,多线程共享访问需要对线程实施同步操作以保证边界数据访问的正确性,线程同步操作会对系统效率产生负面影响^[9]。

对于系统的最后输出部分,考虑到各进程并行的负载均衡问题,往往把水质部分的输出和水动力部分的输出分开,在根进程中输出质地部分的数据,在其他任一进程中输出水动力部分的数据。

3 实验结果及分析

为了验证胶州湾水质预报系统 MPI + OpenMP 混合并行模型的正确性,本实验将混合模型在多核心集群上进行了测试,与串行和原有非阻塞 MPI 并行模型的结果一致,实验证明 MPI + OpenMP 的混合并行模型是可靠的。实验所用多核心集群有 20 个节点,由高速千兆网互联,每个节点的处理有 8 个核。

一般情况下,对于一定规模的问题在采用纯 MPI 进行并行时,程序的并行性能并不随着进程数量的增加而提高,它会有一个最优值,这个时候使用混合编程会达到一个较好的加速比^[10]。因为系统是短期预报且只预测 1~2 d 的水质状况,给出的测试数据为预测 2d 的实验数据。其水质模块纯 MPI 非阻塞通信并行性能测试结果如表 1。将表 1 中执行时间、加速比分别与进程个数的关系转换为折线图的形式,如图 4。

表 1 中的性能结果及图 4 中折线走势表明,在纯 MPI 非阻塞通信模式下,当进程个数不超过 8 时,系统并行效率基本在 80% 以上,当进程数继续增加,程序依然保持加速,但趋势已明显缓慢。

表 1 非阻塞通信性能测试结果

Tab. 1 Performance test results of non-blocking communications

进程个数	运行时间/s	加速比	效率
1	2014	1.00	1.00
2	1178	1.71	0.86
4	547	3.68	0.92
6	404	4.99	0.83
8	318	6.33	0.79
10	283	7.06	0.71

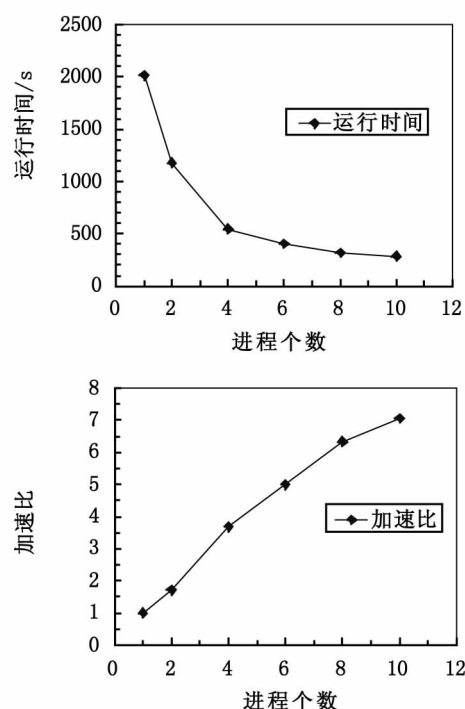


图4 纯MPI并行系统运行结果,左右分别为其运行时间、并行加速比与进程个数的关系

Fig. 4 The running result of pure MPI parallelization, the left and right are the relationship of computation time and speedup

混合并行模型不同进程数不同线程数的运行时间及加速比折线图分别如图 5、图 6 所示。综合图 4、5、6 可得出,对水质模块的行和列都并行的混合并行模型比只在列并行的纯 MPI 并行系统具有很大优势,当进程数等于 2 时,随线程数在 1~8 范围内增加,其加速比接近线性加速比,当线程数为 8 时,其加速比已超过 11,比纯 MPI 并行进程数为 8 时还高出 4.46。当进程数为 4 线程数为 8 时,其加速比已超过 16,计算时间比串行程序缩短了约 94%。当进程数为 8 线程数也为 8 时,能够

节省约 95% 的运算时间,且加速比已超过 20。这些都说明 OpenMP 线程的增加对并行系统所带来的负面影响比 MPI 小的多,所以混合并行编程模型应用到胶州湾水质预报系统是很合适的。

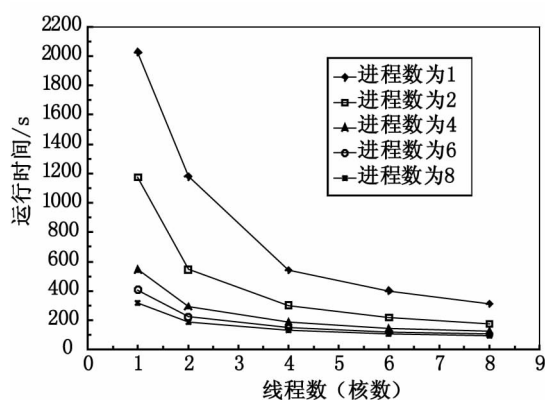


图5 不同进程的运行时间随线程数(核数)变化的折线图

Fig.5 The line chart of the run time of different processes number changing with threads number or cores number

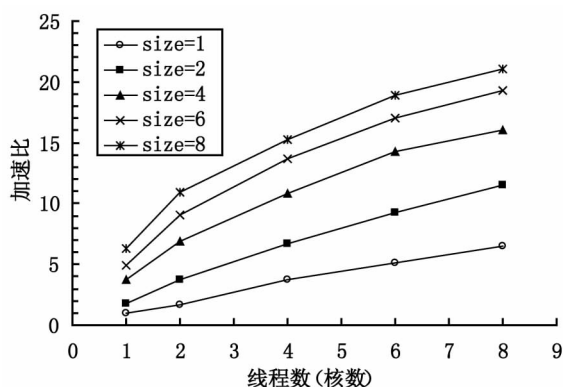


图6 混合模型不同进程数随线程数增加的加速比

Fig.6 The hybrid model speedup of different processes number changing with threads number

4 结论

对于纯 MPI 非阻塞并行模型达到的加速比和并行效率,在 MPI + OpenMP 混合模型中通过减少 MPI 进程数,增加 OpenMP 线程级并行,在不增加

通信开销的前提下,也可以很好的实现,既使系统保持了良好的可扩展性,又节约了集群的内存资源。

参考文献:

- [1] RABENSEIFNER R, HAGER G, JOST G. Hybrid MPI/OpenMP parallel programming on clusters of multi-core SMP nodes [C]. Parallel, Distributed and Network-based Processing, 2009 17th Euromicro International Conference on. IEEE, 2009: 427-436.
- [2] ADHINTO L, CHAPMAN B. Performance modeling of communication and computation in hybrid MPI and OpenMP applications [J]. Simulation Modelling Practice and Theory, 2007, 15(4): 481-491.
- [3] CHORLEY M J, WALKER D W, GUEST M F. Hybrid message-passing and shared-memory programming in a molecular dynamics application on multicore clusters [J]. International Journal of High Performance Computing Applications, 2009, 23(3): 196-211.
- [4] 牛沛. 并行计算在胶州湾水质预报系统中的应用研究 [D]. 邯郸: 河北工程大学, 2012.
- [5] 冯云, 周淑秋. MPI + OpenMP 混合并行编程模型应用研究 [J]. 计算机应用系统, 2006, (2): 86-89.
- [6] 阎菊, 鲍献文, 王海, 等. 胶州湾污染物 COD 的三维扩散与输运研究 [J]. 环境科学研究, 2001, 14(2): 14-17.
- [7] KRAWEZIK G, CAPPELLO F. Performance comparison of MPI and OpenMP on shared memory multiprocessors [J]. Concurrency and Computation: Practice and Experience, 2006, 18(1): 29-61.
- [8] DAGUM L, MENON R. OpenMP: an industry standard API for shared-memory programming [J]. Computational Science & Engineering, IEEE, 1998, 5(1): 46-55.
- [9] 周伟. 并行计算在海洋水龄谱模拟中的应用研究 [D]. 邯郸: 河北工程大学, 2013.
- [10] 王惠春. 基于 SMP 集群的 MPI - OpenMP 混合并行编程模型研究与应用 [D]. 湘潭: 湘潭大学, 2008.

(责任编辑 刘存英)