

蒙特卡罗中子 - 光子输运程序 MCNP 的并行化^{*1)}

邓 力 黄正丰 许海燕 王瑞宏

(北京应用物理与计算数学研究所)

PARALLELIZATION OF MCNP MONTE CARLO NEUTRON AND PHOTON TRANSPORT CODE

Deng Li Huang Zhengfeng Xu Haiyan Wang Ruihong
(*Institute of Applied Physics and Computational Mathematics*)

Abstract

The coupled neutron and photon transport Monte Carlo code MCNP (version 3B) has been realized the parallelization in PVM and MPI by modifying serial code. The results of sample problems prove the correction of code. The speedup increases linearly and the efficiencies are up to 98.5% relate to parallel code and up to 95.7% relate to serial code for 12-processor, respectively.

Key words: PVM, MPI, Monte Carlo, Parallelization

§ 1. 引 言

随着并行计算机的问世, 并行算法和并行系统也不断发展, 如 PVM (Parallel Virtual Machine), SMP (Share Memory Processors), MPI (Message Passing Interface) 和 HPF (High Power Fortran) 等, 这些并行系统原理基本相同, 差异主要是并行指令和数据传递方式. 在这些并行系统中, PVM 和 MPI 系统具有通用性强、系统规模小、使用方便和可移植性强的优点, 且安装、测试、编程与实现相对要容易一些, 它是当前国际上公认的两种消息传递标准软件系统, 它比 SMP, HPF 系统成熟度高, 已能在国际上所有并行机上运行.

PVM, MPI 适合于并行区调用层次复杂, 并行横向处理器之间通讯少, 计算相对独立的问题, 如 Monte Carlo 问题. SMP 为共享存储式并行系统, 它在并行循环区外要对并行区内所涉及的变量进行变量类型说明. 适合并行区调用层次比较清晰、简单且易于描述的问题, 也适合于计算中各处理器之间有通讯关系、计算彼此有依赖关系的动态问题, 如迭代

* 1999 年 11 月 26 日收到.

1) 中国工程物理研究院重点基金资助.

问题, 矩阵运算等, 它同样适用于 Monte Carlo 问题. MPI 与 PVM 具有相似的特性. 一般来说, PVM 和 MPI 系统更适用于分布式并行计算, 特别适合微机并行计算.

在对中子 - γ 耦合输运 Monte Carlo 程序 MCNP(3B 版)^[1,2] 作 SMP 并行改造时发现, 由于原串行程序含有大量等价语句、覆盖结构和动态可调数组, 实现起来非常困难. 相比之下 PVM 和 MPI 方式对改造原有串行程序为并行程序要容易一些. 它只需对程序进行模块划分, 把动态可调数组变为公用数组, 将计数累加量定义为处理器的相关量, 除随机数产生改变了原有程序算法外, 程序的其它算法均不变, 它有利用程序版本的更新.

美国 Los Alamos 实验室 1997 年推出了 MCNP 4B 中子 - 光子 - 电子耦合输运 Monte Carlo 程序^[3], 4B 版程序为 PVM 并行版程序, 但在有关的文章中, 只给出了样本问题的并行加速比^[4], 具体做法未予介绍, 也未给出并行相对串行程序的加速比. 从双方取得的加速比可以看出, 我们得到的加速比是理想的, 超过了国外同类程序的水平.

§ 2. 并行随机数产生

Monte Carlo 程序并行的关键是随机数产生的并行化, 串行随机数序列 $\{x_n\}$. 满足递推公式

$$x_{n+1} \equiv \lambda x_n \pmod{M}, \quad n = 0, 1, 2, \dots, \quad (2.1)$$

$$r_{n+1} = x_{n+1}/M, \quad 0 < r_{n+1} \leq 1, \quad (2.2)$$

其中 λ 为乘子, x_0 为初始随机数, r_{n+1} 为 $(0, 1)$ 上均匀分布的随机数.

采用跳跃法, 并行随机数序列 $\{y_n\}$, 满足递推公式^[5]

$$y_{n+1}^{(k)} \equiv \lambda^l y_n^{(k)} \pmod{M}, \quad k = 1, \dots, l; \quad n = 1, 2, \dots, \quad (2.3)$$

其中 l 为处理器总数.

$\{y_n^{(k)}\}, \{x_n\}$ 满足关系

$$y_n^{(k)} = x_{l(n-1)+k}, \quad n = 1, 2, \dots; \quad k = 1, \dots, l. \quad (2.4)$$

2.1. 程序结构分析

通常满足如下三原则的程序适合并行计算: (1) 数据独立; (2) 大粒度循环; (3) 负载均衡. 并行加速比是检验并行效率的主要手段, 它是由问题中纯计算量所占百分比决定的.

MCNP 程序满足并行计算的三个条件, 原程序由五个覆盖节组成: 输入模块 IMCN; 输入绘图模块 PLOT; 截面数据模块 XACT; 输运模块 MCRUN 和计算结果绘图模块 MC-PLOT. 由于输运模块占计算时间的 98%–99%, 因此, 并行对象选择输运模块.

MCNP 程序诞生于八十年代初期, 由于受当时计算机内存限制, 从节约空间和内存考虑, 程序设计中使用了覆盖结构、可调数组、动态可调数组, 等价语句等. 这些经济做法对串行程序非常有效, 但给并行改造带来了诸多不便. 在做 SMP 并行改造中遇到了很多困难,

程序改动大,甚至影响算法.相比之下, PVM, MPI 系统对程序具有较好的宽容性,对程序编法、语句格式要求不多,更没有 SMP 那样的限制,非常适合早期串行程序的并行改造.

PVM 程序由 Master 和 Slave 组成, Master 为总控程序(串行),负责任务的分发、数据传递和接受来自 Slave 的信息; Slave 为 Master 的分支,从事并行计算^[6].为节约计算机资源,提高计算效率,并行编程往往把 I/O 部分尽量放在 Master 做,对 Slave 中的 I/O 通道不能冲突.

§ 3. 计算结果及加速比

选用 MCNP 3B 基准问题中有代表性的 4 个问题作为算例,作了串行程序与并行程序的对比计算,基准问题 1, 2, 4, 11 分别使用样本数 360 万, 180 万, 120 万和 600 万.表 1、表 2 和表 3 分别给出了两者的计算结果和计算时间对比;表 4 给出了并行程序分别用 1、2、4、8 和 12 个处理器得到的加速比以及平均加速比.加速比估计使用下面公式:

$$S(P) = T_s / T_m(P), \quad (3.1)$$

其中 T_s 为并行程序 1 个处理器下完成计算任务所花 CPU 时间, $T_m(P)$ 为并行 Master 用 P 个处理器完成计算任务所花 CPU 时间.

另外表 5 给出了串行程序相对并行计算时间给出的加速比,图 1 给出了两种情况下加速比与处理器的关系.

表 1 串行与并行 12 个处理器结果比较

中子流		中子点通量		例题 1
平均结果	相对误差	平均结果	相对误差	
8.21540E-1	0.0007	1.39001E-3	0.0039	串行
8.21500E-1	0.0007	1.38183E-3	0.0033	并行
γ 体通量		γ 沉积能		例题 4
2.23634E-3	0.0002	1.75911E-4	0.0001	串行
2.23815E-3	0.0002	1.76012E-4	0.0001	并行
中子体通量		次级 γ 流		例题 11
1.74442	0.0044	1.09364E-1	0.00656	串行
1.74285	0.0045	1.08465E-1	0.00660	并行

表 2 串行与并行 12 个处理器结果比较

中子面通量		中子体通量		中子点通量		例题 2
2.86037E-3	0.0012	2.84218E-3	0.0055	2.85537E-3	0.0016	串行
2.85231E-3	0.0012	2.85359E-3	0.0055	2.85857E-3	0.0016	并行

表 3 MCNP 串行与并行程序不同处理器时间比较

处理器数	MCNP 样本问题计算时间 (分)			
	问题 1	问题 2	问题 4	问题 11
串行	11.43	16.69	14.28	21.16
1	12.09	16.76	15.01	21.31
2	6.06	8.43	7.51	10.64
4	3.02	4.22	3.77	5.32
8	1.52	2.10	1.89	2.67
12	1.02	1.42	1.29	1.78

表 4 并行 MCNP 程序 12 个处理器计算加速比

处理器数	问题 1	问题 2	问题 4	问题 11	平均
2	2.00	1.99	2.00	2.00	2.00
4	4.00	3.97	3.98	4.00	3.99
8	7.95	7.98	7.94	7.98	7.96
12	11.85	11.80	11.64	11.97	11.82

表 5 并行相对串行的加速比

处理器数	问题 1	问题 2	问题 4	问题 11	平均
2	1.89	1.98	1.90	1.99	1.94
4	3.78	3.95	3.79	3.98	3.88
8	7.52	7.95	7.56	7.93	7.74
12	11.21	11.75	11.07	11.89	11.48

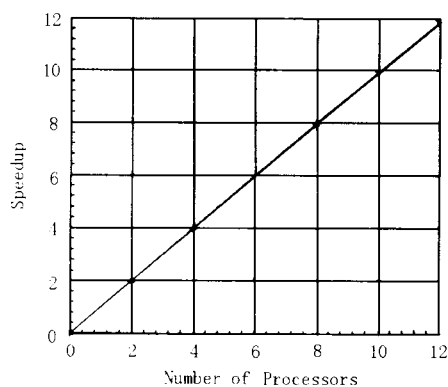


图 1 MCNP 在 12 个处理器上的加速比

§ 3. 结 论

计算结果显示, 并行 MCNP 程序在 12 个处理器的结果与原串行程序结果相符, 两种方式给出的加速比曲线基本重合且均呈线性增长, 两种方式的并行加速效率均在 96% 以上.

参 考 文 献

- [1] J.F. Briesmeister, Editor, MCMP-A General Monte Carlo Code for Neutron and Photon Transport, Version 3A, LA-7396-M, Rev. 2, Los Alamos National Lab. (1986).
- [2] J.F. Briesmeister, Editor, MCNP 3B Newsletter, Los Alamos National Laboratory (July 18, 1988).
- [3] J.F. Briesmeister, Editor, MCNP-A General Monte Carlo Code for N-particle Transport Code, LA-12625-M (March, 1997).
- [4] G.W. MCKINNEY et al., Multiprocessing MCNP on an IBM RS/6000 Cluster, Trans. Am. Nucl. Soc., 68, 212(1993).
- [5] 王嘉谟, 沈毅主编, 并行计算方法, 下册, 科学出版社.
- [6] 孙家昶等编著, 网络并行计算与分布式编程环境, 科学出版社, 1996.
- [7] IEEE. Trans. on Nuclear Science, **39** :4, 1035-1040.