

基于 MPI 的海量数据线性拟合并行算法研究

冯小丹¹, 梁立¹, 袁凌云¹, 高丽金¹, 王文元²

(1 云南师范大学计算机科学与技术学院, 云南 昆明 650092)
(2 红云红河烟草 (集团) 技术中心生理生化研究室, 云南 昆明 650202)

[摘要] 给出了线性模型数据拟合的并行算法描述, 分析了该并行算法的任务分配、通信开销、负载平衡等问题; 通过多项式拟合在 PC 集群环境下的实现, 比较了串行多项式拟合与并行多项式拟合运算时耗的差别, 分析并行算法的性能, 从而探索建立集群环境对于解决海量数据多项式拟合次数范围大、运算速度过慢等问题的可行性。
[关键词] MPI 集群系统, 并行算法, 数据拟合, 线性
[中图分类号] TP301.6 [文献标识码] A [文章编号] 1001-4616(2010) 04-0157-04

Research on Parallel Algorithm of Mass Data Fitting Based on MPI

Feng Xiaodan¹, Liang Li¹, Yuan Lingyun¹, Gao Lijin¹, Wang Wenyuan²

(1 School of Computer Science and Information Technology, Yunnan Normal University, Kunming 650092, China)
(2 Technology Center Physiological and Biochemical Laboratory of HongyunHonghe Tobacco Group, Kunming 650202, China)

Abstract The linear model fitting of parallel algorithm is described, the distribution of tasks, communication costs, load balancing and other issues of the parallel algorithm are analyzed. Via the realization of polynomial parallel fitting in the PC cluster environment, the running time difference between the serial and parallel polynomial fitting is compared and the performance of the parallel algorithm is discussed. The experimental results confirm the feasibility of establishing a cluster environment for solving the problems exist in large data polynomial fitting, such as the wide range and the slow calculation speed.
Key words MPI, cluster system, parallel algorithm, data fitting, linear

随着计算机技术在各学科领域中的广泛应用, 在天体物理、地质、石油、气象、医学等领域产生了大量的科学数据, 为了解释这些数据或者根据这些数据做出预测、判断, 需要寻找一个反映数据变化规律的函数, 另外, 更多的领域对计算机处理精度和速度提出了更高的要求. 在计算机单机系统中, 相对存在的内存资源不足和计算速度过慢的问题, 使得现有的计算机处理能力不能满足实际的需要; 而并行计算为解决海量数据数值运算消耗时间过多问题的重要途径之一. 近年来, 传统数值方法的并行化研究主要有并行 Gauss 全主元消去法^[1]、矩阵乘法并行算法^[2]、矩阵特征值问题并行算法^[3]等; 章勤^[4]、梁立^[5]等对数学模型级上的并行化研究进行了一定的探讨. 结合本研究课题的特点, 本文进行了数学模型级上的数据拟合并行算法的研究, 希望能对海量数据生成函数模型提供一种新的选择.

1 数据拟合函数模型的确定

用初等方法来探讨解决实际问题问题是许多数学工作者和其他学科工作者近年来感兴趣的课题, 人们总是希望方法尽可能初等些, 而解决的问题相对的说并不初等^[6]. 由于本文算法不需要事先给出函数模型而由计算机盲目搜索, 所以给定由初等函数组成的一个函数集合, 再由该集合中的函数元素进行全组

收稿日期: 2010-06-10
基金项目: 云南省社会发展基础研究 (面上项目) (2008CD113)、云南省教育厅基金 (08Y0136).
通讯联系人: 冯小丹, 硕士, 研究实习员, 研究方向: 教务管理. E-mail: feda_0358@163.com

合,生成不同的函数模型.

2 线性模型数据拟合的并行算法

2.1 算法描述

为了从一批海量数据中快速得到最佳函数模型,算法采用分而治之的思想,把不同的函数拟合作为不同的子任务分摊到各个节点上,每个节点独立地进行拟合求解,再把拟合结果汇集,根据一定标准选择出最佳模型^[5]. 算法采用了并行程序设计中的主从模式,具体算法描述如下:

(1) 主进程的算法描述

①将要拟合的数据以文件形式读入,并广播给各子进程作为函数拟合的数据源;

②对给定的一组函数集合,定义模型矩阵,其列对应函数集合元素的位置,行则对应不同的组合情况,列数和行数分别为函数集合中元素的个数和全组合数,并将矩阵元素初始化为 0

③对函数集合中的元素进行组合,每种组合对应于模型矩阵的一行,每行又根据函数组合情况,对出现的函数,在对应的列位置标记 1;主进程对所有的行标记“未拟合”,然后发送给每个子进程一行;随后,主进程将重复以下过程:接收到某个子进程的返回结果,即模型的残差绝对值之和,就将相应的行标记为“已拟合”;若有标记为“未拟合”的行,从中选择一行发送给该子进程处理.该过程直到所有的模型行都标记为“已拟合”为止.

④选取残差绝对值之和最小者作为最优函数模型,并用最优函数模型进行最终的数据拟合.

(2) 从进程的算法描述

①接收主进程传来的要进行函数拟合的数据;

②接收主进程传来的模型矩阵的一行,并结合函数集合,确定使用函数集合中的哪些函数线性生成函数模型;

③用生成的函数模型拟合数据;

④计算拟合函数和数据点残差的绝对值之和,并发送给主进程.

2.2 任务划分

算法按行来进行任务划分,假设处理器数目 P ,模型矩阵的行数为 N ,当处理器数目 P 小于矩阵的行数 N 时,主进程按行动态分配给从进程,通信时间为 $O(N)$.

2.3 通信模式及通信开销

本文在并行数据拟合计算过程中,从进程专门负责某个数学模型的数据拟合,从进程之间基本上不需要进行通信,只有当主进程向从进程发送要拟合的数据和函数模型,以及最后从各个从进程收集残差时,主从进程之间才有通信.可见,通信占用的时间非常少.

2.4 动态负载平衡

运行时间是衡量并行算法性能的重要指标,而运行时间由运行时间最长的进程决定.动态负载平衡是一种自调度算法,包括发送者启动算法、接收者启动算法和对称启动算法等策略^[7].本算法采用接收者启动算法策略,由接收者来激发模型的分配,即当一个子进程完成一个子任务后,它主动向主进程发出负载请求.

2.5 算法性能分析

(1) 通信步

假设模型矩阵的行数为 N ,列数为 M ,处理器数目为 P .由于要把拟合的数据传送给 $P-1$ 台子处理器,而且有 N 个模型拟合任务需要返回结果,还有主进程根据结果响应动态分配子任务进行的 N 次通信,则通信步为: $2N + P - 1$

(2) 通信复杂度

假设要拟合的数据量为 T ,加上返回的 N 个残差的绝对值之和,以及主进程发送给从进程的 N 个子任务的数据量为 NM ,则通信量为 $T + N + NM$,算法的通信复杂度为 $O(NM + T)$.

(3) 并行度

在从计算开始直到整个计算结束的时间范围内,能够并行执行程序的处理器 P ,所有处理器均处于繁

忙状态, 所以并行度为 P .

3 多项式拟合并行算法实现及算法性能分析

最小二乘法数据拟合可以用任何一组基本函数来实现, 在各类函数中, 用多项式是最简单最实用的. 通过对多项式函数的研究和探讨, 可以进一步研究一般函数的同类问题. 因此, 本节在 2.1 算法的基础上, 将不同次数的多项式作为函数集合, 把多项式模型按次数的不同分配到不同的节点上, 每个节点进行一种多项式数据拟合.

本文实验平台由 6 台同构的 PC 机用 10M 交换机互联组成局域网, 集群中 PC 机的配置如下: CPU: Intel(R) Pentium(R) D CPU 2.80 GHz; 内存: 512MB; 硬盘: 80 G; 网卡: 10M /100M 以太网卡. 另外, 采用的操作系统是 Red Hat Linux 3.2.2-5 并行编程环境采用 MPICH-1.2.7p1.

在测试过程中, 由于并行计算网络环境实验室局域网的不稳定性, 所以对每一种情况下并行计算时间的确定, 采用多次运行取平均值的方法. 以下是用随机生成的数据在单机和在多个节点处理器进行 3 次多项式并行拟合计算的运行记录, 其中 1 个节点的记录表示相应的串程序序.

并行计算性能的评价在并行处理中非常重要, 并行性能主要受并行环境的通信性能、各个结点机的计算性能以及所采用的并行算法等因素影响^[8].

表 1 是不同规模数据的 3 次多项式并行拟合的测试结果 (时间单位为秒; P 为处理器节点数; T_s 为串行平均时间; T_p 为并行平均时间; S_p 为加速比; E_p 为效率).

表 1 一般多项式 3 次并行拟合的测试结果

Table 1 The test results of polynomial parallel fitting with rank 3

数据规模	串行 ($P=1$)			并行 ($P=3$)			并行 ($P=4$)			并行 ($P=5$)			并行 ($P=6$)		
	T_p	S_p	E_p	T_p	S_p	E_p	T_p	S_p	E_p	T_p	S_p	E_p	T_p	S_p	E_p
5×10^3	0.007	1	1	0.003 9	1.79	0.60	0.003 1	2.26	0.57	0.002 6	2.69	0.54	0.002 3	3.04	0.51
5×10^4	0.03	1	1	0.015 4	1.95	0.65	0.011 7	2.56	0.64	0.009 6	3.13	0.63	0.008 5	3.53	0.59
5×10^5	0.35	1	1	0.171 6	2.04	0.68	0.132 6	2.64	0.66	0.107 6	3.25	0.65	0.092 5	3.78	0.63
5×10^6	3.14	1	1	1.359 3	2.31	0.77	1.129 4	2.78	0.70	0.940 0	3.34	0.67	0.805 6	3.90	0.65

由表 1 可以看出:

(1) 问题规模对效率 (加速比) 的影响

随着数据规模的增大, 并行加速比逐渐增大, 并行效率逐渐提高; 当数据规模较小时, 加速比和效率均比较低, 这是由于数据量小时, 计算时间少, 通信时间在整个运行时间中占相当大的比重; 而随着数据规模的增大, 通信开销在整个计算过程中显得微不足道, 加速比和效率都达到了好的效果, 充分显示了并行计算的优越性. 如图 1 图 2 所示.

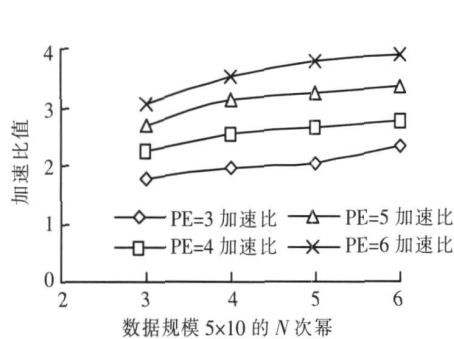


图 1 不同数据规模下的加速比

Fig.1 The speedup in different data scale

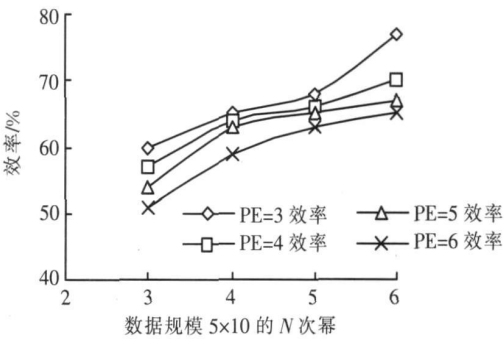


图 2 不同数据规模下的效率

Fig.2 The efficiency in different data scale

(2) 同一规模数据处理器数对效率 (加速比) 的影响

对于同一规模数据, 计算时间随处理器的增加而减少, 并行加速比逐渐增大, 而并行效率逐渐降低. 这符合并行计算的一般规律, 这是因为求解问题的规模一定, 计算量不变, 而通信量随着 CPU 的增多而增加, 从而造成了效率的下降. 图 3 图 4 图 5 分别表示固定数据规模为 5×10^6 时, 在不同节点处理器数目下的运行时

间、加速比和效率变化.

(3) 可扩展性

1993年, Grama A Y 等^[9]提出了度量并行算法可扩展性的恒等效率方法: 在处理器增加的情况下, 只要适当地增加问题规模, 可以保持算法的效率不变. 由下图可看出, 随着处理器的增加, 效率曲线略有下降, 可见算法可扩展性较好.

股票市场在我国不断发展, 因而对股票交易数据的分析和预测研究具有重大的理论意义和诱人的应用价值. 而股票预测需要通过大量的数据分析才能得到, 本文实验采用长江

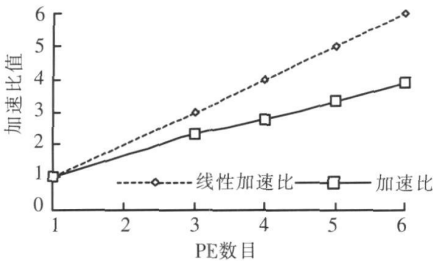


图 4 不同 PE 数目下的加速比
Fig.4 The speedup indifferent PE

电力 (600900) 股票数据, 进行 1 到 100 次正交多项式随机拟合, 测试结果见表 2

由表 2 可见, 随着正交多项式次数的增加, 并行加速比和效率逐渐增加. 特别是多项式次数达到 100 次时, 效率为 71%. 因此也可以看出当函数模型增加时, 并行算法相对串行算法的优越性也越来越明显.

4 小结

由前面的讨论可知, 将并行思想引入多项式拟合, 提高了运行速度; 和串行算法相比, 并程序大大缩短了计算时间; 但并行程序的执行速度和效率并不是随着处理器的增多与串程序成线性比例关系的, 只有数据规模较大时, 才能更好地体现出并行计算的效益. 而对于本文研究的海量数据, 拟合的计算量会很大, 这时就可以利用并行拟合来提高计算的速度和效率.

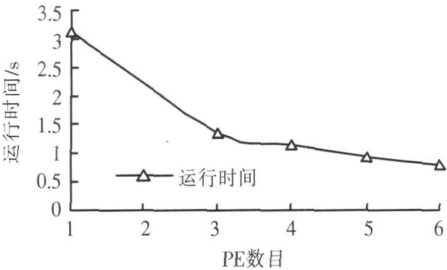


图 3 不同 PE 数目下的运行时间
Fig.3 The running time in different PE

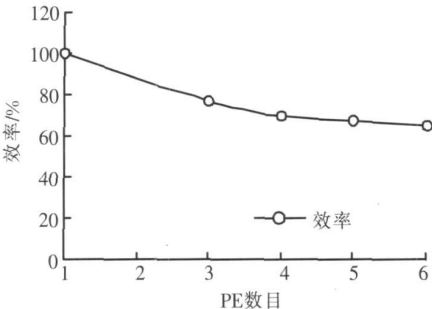


图 5 不同 PE 数目下的效率
Fig.5 The efficiency in different PE

表 2 股票数据正交多项式的并行拟合测试结果
Table 2 The test results of orthogonal polynomial parallel fitting for stock data

多项式 次数	串行	并行 (P = 6)		
	平均时间 /s	平均时间 /s	加速比	效率 %
5	7. 15	2. 971 300	2. 41	40
10	11. 23	4. 127 544	2. 72	45
25	23. 28	8. 382 394	2. 78	46
50	39. 45	13. 365 243	2. 95	49
100	82. 09	19. 367 252	4. 24	71

[参考文献]

[1] 孙济洲, 樊莉亚, 孙敏, 等. 改进的并行高斯全主元消去法 [J]. 天津大学学报, 2006, 39 (9): 1115- 1119.
[2] 孟祥杰, 张理论, 曾泳泓. 分布式存储环境下矩阵转置并行算法研究 [J]. 计算机工程与科学, 1999, 21 (5): 67- 71.
[3] 李晓梅, 罗晓广. 大型矩阵特征值问题并行计算研究概况 [J]. 指挥技术学院学报, 2001, 12 (3): 1- 5.
[4] 章勤, 李品, 肖舸. 海量数据的曲线拟合并行算法及实现 [J]. 华中理工大学学报, 2000, 28 (10): 85- 87.
[5] 梁立, 肖飞, 郜松. 最佳模型拟合的并行算法 [J]. 计算机科学, 2007, 34 (6): 294- 295.
[6] 张宝琳. 发展并行数值方法 [J]. 高等学校计算数学学报, 1997 (1): 1- 6.
[7] 邱颖豫, 杜根远. 并行计算中的负载均衡问题研究 [J]. 福建电脑, 2006 (12): 3- 4.
[8] 孟洁, 王小鸽, 李三立. 并行计算性能的分析与预测 [J]. 计算机科学, 1999, 26 (2): 14- 17.
[9] Grama A Y, Gupta A, Kumar V. Isoefficiency: measuring the scalability of parallel algorithms and architectures [J]. IEEE Parallel & Distributed Technology, 1993, 1 (3): 12- 21.

[责任编辑: 丁 蓉]