

# 分块带边结构线性规划并行算法

杨林峰<sup>1</sup> 李陶深<sup>1</sup> 李捷<sup>2</sup> 陈燕<sup>1</sup>

(广西大学 南宁 530004)<sup>1</sup> (广西职业技术学院计算机与电子信息工程系 南宁 530226)<sup>2</sup>

**摘要** 基于内点算法(Interior Point Method, IPM)框架,导出具有分块带边结构系数矩阵的线性规划(Linear Programming, LP)问题的简化和最简修正方程,并证明最简修正方程的对角分块具有正定性。结合正定矩阵的 Cholesky 分解和解耦技术设计了修正方程的并行求解方法,给出了 LP 的并行内点算法结构。集群环境下的数值实验表明,所提算法具有很好的加速比和可扩展性,适合求解大规模结构化 LP 问题。

**关键词** 线性规划,分块带边矩阵,并行算法,解耦,最简修正方程

**中图分类号** TP301.6 **文献标识码** A

## Parallel Algorithm of Block Bordered Linear Programming

YANG Lin-feng<sup>1</sup> LI Tao-shen<sup>1</sup> LI Jie<sup>2</sup> CHEN Yan<sup>1</sup>

(Guangxi University, Nanning 530004, China)<sup>1</sup>

(Department of Computer, Electronics and Information Engineering, Guangxi Polytechnic, Nanning 530226, China)<sup>2</sup>

**Abstract** This paper presented the simpler and simplest correction equation of linear programming(LP) with block bordered coefficient matrix based on the framework of interior point method(IPM). And the diagonal sub-matrix in the simplest correction equation was proved to be symmetric positive definite. Parallel IPM algorithm for LP was presented after a parallel solver for correction equation was proposed by integrating decoupling and Cholesky factorization of symmetric positive definite matrix. The simulations in the cluster show that the proposed method is very promising for large scale LP problems due to its excellent speed up and scalability.

**Keywords** Linear programming, Block bordered matrix, Parallel algorithm, Decoupling, Simplest correction equation

## 1 引言

LP 是处理在线性等式和不等式约束下的线性目标函数的极值问题,被广泛地应用于军事、工业、运输、通讯、城市规划、生产计划、经济管理和政府的科学决策等方面。LP 的相关技术还被应用于组合规划、网络规划、自动控制以及计算机科学等领域并成为这些学科的有效工具。1984 年, N. Kar-markar 提出了一个具有多项式时间复杂度的 IPM 算法,并指出该算法在问题规模较大时计算效率将全面超过具有非多项式时间复杂度的单纯形法。随着世界范围内从理论和计算方面对 IPM 算法进行广泛深入的研究,各种更有效的改进和扩展算法相继问世,包括原始对偶内点法(Primal Dual IPM, PDIPM)<sup>[1]</sup>、预测校正内点法<sup>[1]</sup>、多中心校正内点法<sup>[2]</sup>等。

目前并行 LP 求解领域的主要研究方向是<sup>[3]</sup>:

全新设计:根据问题的规模和组合本质推动设计新的全局优化问题并行求解技术,设计全新的具有内在并行性的 LP 求解算法。该类并行算法往往与实际问题的本质有较强关联,实现难度大,但往往具有较好的并行加速比和可扩展性<sup>[4,5]</sup>。

集成设计:对 LP 求解算法的线性代数计算内核进行并行化,即把并行的线性代数算法封装到优化代码中,实现并行 LP 求解<sup>[6,7]</sup>。

在 LP 问题求解中,线性代数运算占有很重要的地位。特别是采用 IPM 来实现 LP 求解时,线性代数问题对其更为重要。与此同时,集成设计的并行 LP 算法设计方式对最优化算法理论要求不高,可以有效借鉴现有的并行线性代数算法成果<sup>[8,9]</sup>,十分适合并行计算、计算机科学领域的学者进行研究,是目前并行 LP 算法研究的热点。

本文研究工作也归属为并行算法的集成设计一类。首先对采用 IPM 算法求解 LP 问题进行了讨论和分析,并针对一类常见的具有分块带边结构系数矩阵的 LP 问题,导出其简化修正方程与最简修正方程,并结合最简修正方程结构给出了一种基于解耦的 LP 并行算法。

## 2 线性规划内点算法

### 2.1 问题模型

LP 模型如下:

$$\min c^T x$$

到稿日期:2010-10-20 返修日期:2011-02-24 本文受国家自然科学基金(60963022),广西自然科学基金项目(桂科自 0832056),广西高校人才小高地建设创新团队资助计划(桂教人[2007]71 号),广西研究生教育创新计划资助项目(105930901022)资助。

杨林峰(1979—),男,博士生,讲师,主要研究方向为并行最优化算法及其在电力系统中的应用, E-mail: ylf@gxu.edu.cn;李陶深(1957—),男,博士,教授,主要研究方向为分布式计算、无线 Mesh 网络;李捷(1979—),女,硕士生,讲师,主要研究方向为分布式计算技术, E-mail: janelee@gxzyjy.com(通信作者)。



展开如下:

$$L_i l_i = h_i, i=1, \dots, p \quad (22)$$

$$L_{p+1} l_{p+1} = h_{p+1} - \sum_{i=1}^p F_i l_i \quad (23)$$

$$L_{p+1}^T \Delta y_{p+1} = l_{p+1} \quad (24)$$

$$L_i^T \Delta y_i = l_i - F_i^T \Delta y_{p+1}, i=1, \dots, p \quad (25)$$

上述分解-回代过程中,式(17)、式(18)都分别为  $p$  个独立的子问题,可以并行进行计算;式(22)为  $p$  个独立的下三角线性方程组,可以并行进行求解;式(25)为  $p$  个独立的上三角线性方程组,也可以并行进行求解。

### 3.2 算法步骤

步骤0 初始化,取  $\delta \in (0, 1), \epsilon = 10^{-6}, k=0, k_{\max}=50$ , 给定初值  $(x^0, y^0, s^0)$ 。

步骤1 计算对偶间隙  $g = (x^k)^T (s^k)$ 。

步骤2 如果  $k \geq k_{\max}$ , 迭代不收敛, 则结束; 如果  $g < \epsilon$ , 输出最优解, 则结束。否则进入步骤3。

步骤3 计算扰动因子  $\mu = \delta \cdot g/n$ 。

步骤4 修正方程并行计算:

步骤4.1 分别并行进行式(17)、式(18)所示的各  $p$  个计算任务;

步骤4.2 进行式(19)所示计算, 完成修正方程分解;

步骤4.3 并行进行式(22)的回代计算, 并计算式(23), 完成第一次回代运算;

步骤4.4 类似地计算式(24)后再并行计算式(25), 进行第二次回代, 得到  $\Delta y$ , 再按式(13)、式(11)得到  $\Delta x$  及  $\Delta s$ , 完成整个修正方程(8)的求解。

步骤5 按式(9)确定原始与对偶步长。

步骤6 按式(10)更新原始、对偶变量,  $k=k+1$ ; 转步骤1。

## 4 算法实现与测试

### 4.1 实验平台

本文采用 Matlab Parallel Computing Toolbox(PCT)<sup>[11]</sup> 搭建集群。集群有4个节点, 节点基本信息为: Intel core 2 Duo 处理器(E7200), 2.53GHz; 2GB 内存; 安装 Windows XP 操作系统, Matlab 7.8。节点间采用 100MB 以太网互联。

PCT 是 MATLAB 中的并行与分布式处理组件, 提供了高级构造, 如并行 for 循环、并行算法、基于 MPI 的函数以及用于作业和任务管理的低级构造。利用并行计算工具箱开发分布式和并行应用程序时, 该工具箱使应用程序能够在包含最多4个本地作业的桌面进行原型开发, 并且通过 MATLAB 分布式计算引擎, 可将这些应用程序调整到一个集群上的多台计算机。本文采用 PCT 基于主从模式将核心计算进行任务划分, 并均衡地提交至各个节点进行计算。

### 4.2 实验结果

本文采用电力系统中经典的预想事故最优潮流问题作为计算实例, 并将其所有的非线性项采用一阶近似进行线性化, 得到其具有分块带边结构的近似 LP 模型。选取了两个计算实例: IEEE 30 节点系统和 300 节点系统, 两个系统的系数矩阵 A 分别为 2625 阶和 29328 阶分块带边方阵。

为对比说明本文所提算法的有效性, 分别采用文献[12, 13]中的异步 Gauss-Seidel 方法和并行 LU 分解方法来直接求解修正方程(8)并实现整个内点算法。简称这两种算法为并行的 GS-IPM 和 LU-IPM 算法。

表1给出了3种算法并行求解300节点系统的计算结果比较, 其中黑体为表中最好结果。由于GS-IPM和LU-IPM算法是直接求解修正方程(8), 本文所提算法求解的是最简修正方程(14), 其规模比式(8)要小, 因此, 所提算法计算时间比前两种要少。同时, 由于所提算法充分考虑了最简修正方程的分块带边结构特征, 其并行求解加速比明显高于前两种算法。图1给出了本文算法求解30节点系统和300节点系统的并行加速比情况。如图1所示, 对于两个计算实例, 随着处理器核的数目的增加, 算法加速比都能够得到稳步提升, 使得计算时间得以减少; 同时大系统(300节点)的并行加速比增加更为明显, 主要原因是采用 Matlab PCT 编写的并行程序存在并行任务启动开销, 这一开销在小系统中显得十分明显, 对加速比产生较大的影响, 而在大系统求解中则影响不大。

表1 并行计算结果比较(300节点系统)

核的数目	GS-IPM		LU-IPM		本文方法	
	计算时间/s	加速比	计算时间/s	加速比	计算时间/s	加速比
1	48.39	1	42.46	1	40.83	1
2	44.87	1.08	40.18	1.06	31.35	1.30
4	42.66	1.13	39.15	1.08	24.76	1.65
6	40.15	1.21	38.88	1.09	21.19	1.93
8	38.01	1.27	38.42	1.11	20.08	2.03

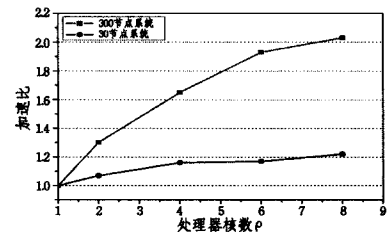


图1 并行计算加速比比较

结束语 本文基于 IPM 思想, 导出了具有分块带边结构的 LP 问题的简化和最简修正方程, 并根据其结构, 结合解耦思想, 设计了该特殊问题的并行算法。本文取得的主要结论如下:

1) 大规模 LP 问题的约束方程系数矩阵往往具有结构化和稀疏特点。充分讨论和利用其结构特征, 导出修正方程系数矩阵的重要特性, 例如对称正定性, 对提高 LP 问题的求解效率十分重要。

2) LP 问题的并行求解过程中, 对分块结构进行解耦, 能够有效提高并行算法的加速比和可扩展性。

在 Matlab 集群环境下对算法的测试表明, 算法具有很好的加速比和可扩展性, 适合求解具有分块带边结构的大规模 LP 问题。后续工作将围绕算法细粒度并行展开, 以进一步提高算法的可扩展性。

## 参考文献

- [1] Wright S J. Primal-dual Interior-Point Methods [M]. Philadelphia, USA: SIAM, 1997: 83-104
- [2] Marco C, Jacek G. Further development of multiple centrality correctors for interior point methods[J]. Computational Optimization and Applications, 2008, 41(3): 277-305
- [3] Migdalasa A, Toraldo G, Kumard V. Nonlinear optimization and parallel computing[J]. Parallel Computing, 2003, 29(4): 375-391
- [4] Palomares G, Rodriguez J. New sequential and parallel derivative-free algorithms for unconstrained minimization[J]. SIAM

[5] Sagastizabal C A, Solodov M V. Parallel variable distribution for constrained optimization [J]. Computational Optimization and Applications, 2002, 22: 111-131

[6] Donghoon L, Wiswall M. A parallel implementation of the simplex function minimization Routine [J]. Computational Economics, 2007, 30(2): 171-187

[7] 陈政洪, 郁松年. 一个基于 QR 分解的并行原-对偶内点算法 [J]. 应用科学学报, 2004, 22(4): 549-552

[8] 骆志刚, 李晓梅. 块三对角线性方程组的一种分布式并行算法 [J]. 计算机学报, 2000, 23(10): 1028-1034

[9] 段治健, 杨永, 马欣荣, 等. 求解带状线性方程组的一种并行算法

[10] Gondzio J, Sarkissian R. Parallel interior point solver for structure linear programs [J]. Mathematical Programming, 2003, 96(3): 561-584

[11] Matlab Parallel Computing Toolbox 4 [EB/OL]. http://www.mathworks.com/access/helpdesk/help/pdf\_doc/distcomp/distcomp.pdf, 2010-03-24

[12] Chern M Y, Murata T. A fast algorithm for concurrent LU decomposition and matrix inversions [C] // Proc. 1983 Int'l Conf. on Parallel Processing. New York: IEEE, 1983: 79-86

[13] 陈国良. 并行算法的设计与分析 [M]. 北京: 高等教育出版社, 2008

(上接第 192 页)

属性名	属性值(离散化)
属性 3 贷款历史状况	1: 无贷款记录或所有贷款均按时返还; 2: 在本银行的所有贷款均按时返还; 3: 迄今为止现存贷款按时返还; 4: 过去曾延迟还款; 5: 存在危帐或仍存在贷款(非本银行)
属性 4 贷款用途	1: 新车; 2: 二手车; 3: 家具设备; 4: 收音机或电视机; 5: 家庭用品; 6: 维修; 7: 教育; 8: 度假; 9: 接受再培训; 10: 经商; 11: 其他用途
属性 5 贷款数额	\
属性 6 储蓄存款 账户状况	1: 账户余额 < 100 马克; 2: 100 马克 ≤ 账户余额 < 500 马克; 3: 500 马克 ≤ 账户余额 < 1000 马克; 4: 账户余额 ≥ 1000 马克; 5: 未知或无储蓄存款
属性 7 现工作就业时间	1: 失业; 2: < 1 年; 3: [1 年, 4 年]; 4: [4 年, 7 年]; 5: ≥ 7 年
属性 8 分期付款占月收入百分比	\
属性 9 个人状况 及性别	1: 男性离异或分居; 2: 女性离异、分居或结婚; 3: 男性单身; 4: 男性结婚或同居; 5: 女性单身
属性 10 其他债务或保证金	1: 无; 2: 联合申请人; 3: 保证人
属性 11 现居住状况	\
属性 12 财产状况	1: 有房产不动产; 2: 无房产不动产, 有社保储蓄协议或养老保险; 3: 无房产不动产, 无社保或养老保险, 有汽车或其他(不在属性 6 范围内); 4: 未知或无财产
属性 13 年龄	\
属性 14 其他分期付款计划	1: 银行; 2: 商店; 3: 无
属性 15 房屋状况	1: 租住; 2: 自有; 3: 免费使用
属性 16 在本银行已有存款数目	\
属性 17 工作状况	1: 失业、无技能或非本地居民; 2: 无技能的本地居民; 3: 技术工人或公务员; 4: 经理、自由职业者、高级雇员或官员
属性 18 应抚养人数	\
属性 19 电话	1: 无; 2: 有或已注册
属性 20 是否外籍劳工	1: 是; 2: 否

## 5.2 训练与测试

本文的 MNN-CREDIT 信用评估模型在实际评估中发现:

1) 学习模型对数据过分特化, 如果使用训练数据导出分类法, 然后信用评估分类法, 可能错误地导致过于乐观的估计。

2)  $k$ -折交叉确认是基于给定数据随机选择划分的, 常用评估分类法准确率的技术。

因此, 为了更准确地评估 MNN-CREDIT 模型的准确率, 把 1000 个数据分成 4 个互不相交的子集数据集, 每个子集各 250 个数据。把这 4 个子集按 3:1 的比例组成训练集和测试集, 进行 4-折交叉确认。4 个信用预测结果的平均值与朴

素贝叶斯模型、支持向量机和  $k$ -近邻判别分析方法( $k=17$ )信用预测结果的精度进行了比较, 如图 5 所示。

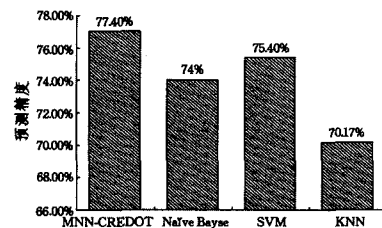


图 5 各信用评估模型的预测精度比较

从图 5 可知, MNN-CREDIT 模型较支持向量机, 其预测精度提高了 1.64%; 较朴素贝叶斯模型, 其预测精度提高了 3.04%; 较  $k$ -近邻判别分析方法( $k=17$ ), 其预测精度提高了 6.87%。

**结束语** 信用评估对于银行控制风险具有十分重要的意义。传统的信用评估方法存在一些缺陷, 本文针对信用评估对象的特点, 提出了适应信用评估的 MNN-CREDIT 模型, 并利用该模型对德国信用数据库的真实数据进行实证分析。结果表明, MNN-CREDIT 模型较传统的信用评估模型, 取得了较高的预测精度, 证明了本文提出的基于多进化神经网络的信用评估模型的有效性。

## 参考文献

[1] 吴江, 唐常杰, 段磊, 等. 基于基因表达式编程的信用评估模型挖掘方法 [J]. 计算机应用, 2007, 27(4): 877-880

[2] 石庆焱, 靳云汇. 个人信用评分的主要模型与方法综述 [J]. 统计研究, 2003(8): 36-39

[3] 王雅林, 赵欣, 黄伟平. KNN 近邻判别分析法在个人信用评估中的应用姜明辉 [J]. 数量经济技术经济研究, 2004(2): 143-147

[4] Krogh A, Vedelsby J. Neural Network Ensembles, Cross Validation, and Active Learning [M]. Advance in Neural Information Processing Systems, 1995, 7: 231-238

[5] Radolph G. Convergence Analysis of Canonical Genetic Algorithms [J]. IEEE Transactions on Neural Network, 1994, 5(1): 96-101

[6] Tsymbal A, Pechenizkiy M, Cunningham P. Diversity in Search Strategies for Ensemble Feature Selection [J]. Information Fusion, 2005, 6(1): 83-98

[7] 赵静娴, 杨宝臣. 一种基于神经网络和决策树的信用评估新方法 [J]. 武汉科技大学学报, 2005, 7(2): 36-39

[8] 张彤, 向南峰. 基于 BP 网络的数字识别方法 [J]. 重庆理工大学学报: 自然科学版, 2010, 24(3): 43-47