

广义共轭余差法的通信避免算法

金之雁¹, 杨磊¹, 林隽民², 王哲²

1. 中国气象科学研究院, 北京 100081

2. 英特尔中国有限公司, 北京 100081

摘要: 广义共轭余差法是一种用于求解非对称线性方程组的有效算法。为减少算法中的全局通信, 首创性地提出了“通信避免的广义共轭余差法”, 避免了迭代过程中的全局通信, 使算法中的全局通信总次数降低了一个数量级, 同时减少了约50%的计算量(计算量的具体减少比例与计算规模相关)。大规模测试中(最大16 384进程), 新算法最高达到了原算法3倍的运算速率。进一步分析表明, 新算法在各种并行规模下的运算速率和可扩展性都优于原算法。在较小并行规模下, 新算法的优势主要来源于计算量的减少。在较大并行规模下, 新算法的优势主要来源于全局通信量的减少。

关键词: 通信避免算法; 广义共轭余差法; 并行计算; 全球区域一体化数值预报模式; 曙光-派计算集群

文献标志码: A **中图分类号:** O246 **doi:** 10.3778/j.issn.1002-8331.1810-0413

金之雁, 杨磊, 林隽民, 等. 广义共轭余差法的通信避免算法. 计算机工程与应用, 2020, 56(3): 74-79.

JIN Zhiyan, YANG Lei, LIN Junmin, et al. Communication avoiding algorithm of generalized conjugate residual method. Computer Engineering and Applications, 2020, 56(3): 74-79.

Communication Avoiding Algorithm of Generalized Conjugate Residual Method

JIN Zhiyan¹, YANG Lei¹, LIN Junmin², WANG Zhe²

1. Chinese Academy of Meteorological Sciences, Beijing 100081, China

2. Intel China Ltd, Beijing 100081, China

Abstract: Generalized Conjugate Residual (GCR) method is an effective algorithm to solve asymmetrical linear systems. To reduce global communication of the algorithm, this paper puts forward the “communication avoiding generalized conjugate residual method”, avoids the global communication in the iterative process, global communication in the algorithm is reduced by an order of magnitude, at the same time reduces about 50% of the computation (the reduction of computation and communication related to scale). In the massively experiment (the maximum process is 16 384), the new algorithm is up to three times as fast as the original algorithm. Further analysis shows that: the new algorithm is faster and more scalable than the original algorithm in various parallel scales. In the case of small parallel scale, the advantages of the new algorithm mainly come from the reduction of computing. In the case of large parallel scale, the advantage of the new algorithm mainly comes from the reduction of global communication.

Key words: communication avoiding algorithm; generalized conjugate residual method; parallel computing; global/regional assimilation and prediction system; Sugon_π cluster

1 引言

广义共轭余差法 (Generalized Conjugate Residual method, GCR)^[1] 是一种用于求解非对称线性方程组的

有效算法。中国气象局的新一代“全球区域一体化数值预报模式 (Global/Regional Assimilation and Prediction System, GRAPES)”^[2], 其动力框架的核心是亥姆霍兹方

作者简介: 金之雁 (1962—), 通信作者, 男, 博士, 正研级高工, 主要研究领域为数值气象预报的并行计算, E-mail: yanglejinju@163.com; 杨磊 (1985—), 男, 硕士, 主要研究领域为高性能计算; 林隽民 (1978—), 男, 博士, 主要研究领域为高性能计算; 王哲 (1976—), 男, 博士, 主要研究领域为高性能计算。

收稿日期: 2018-11-01 **修回日期:** 2018-12-18 **文章编号:** 1002-8331(2020)03-0074-06

CNKI 网络出版: 2019-01-16, <http://kns.cnki.net/kcms/detail/11.2127.TP.20190114.1730.022.html>

程求解器,求解器所采用的就是GCR算法。和其他类似算法一样,GCR算法存在的一个制约可扩展性的问题就是密集的全局通信。随着CPU单节点运算能力的不断提高,这个问题已经成为了制约整个数值模式速率提高的主要瓶颈。

一般来说,算法的开销可以归结为两部分:数据的计算和数据的移动。其中数据的移动包括节点内部的移动和节点间通过网络的移动。要减少这种密集的数据移动带来的开销,加州大学伯克利分校的Demmel教授最早于2007年提出了通信避免(Communication Avoiding,CA)算法^[3]。CA算法的思路是:构造能包含迭代过程中的所有长向量的Krylov子空间,利用短向量的迭代替代长向量的迭代,进而避免在迭代过程中进行通信。该思路具体应用于对不同的算法的改造,会产生相应的不同的CA算法。目前针对GCR算法,相关工作仍是空白。

数值预报模式的运算速率是提高预报分辨率的客观前提,“神威太湖之光”、“曙光派”等高性能计算集群的部署已经使GCR算法中的全局通信成为提升GRAPES模式整体速率的最大瓶颈。为此,本文借鉴Demmel教授提出的思路,首创性地对GCR算法进行CA改造,提出CA-GCR算法。新算法的通信次数较之原算法降低了一个数量级,同时,计算量也有一定减少。并且在中国气象局最新部署的“曙光派”计算集群上进行了大规模测试(最大规模16384进程),在可扩展性(本文中所有“可扩展性”均指“强可扩展性”)和运算速率上表现出了相对于原算法巨大的优势,最高达到原算法3倍的加速比。

2 相关工作

CA算法由Demmel教授提出,之后又由其本人及学生具体应用于共轭梯度法、广义最小剩余法等经典算法的改造,多数表现出了对通信量和计算量的双重作用,即同时减少两者的开销。Demmel教授的学生Grigori于2008年将CA算法应用于一般高斯消元法^[4];Demmel于2010年对LU分解进行CA优化^[5];Hoemmen于2010年对广义最小剩余法进行CA优化^[6];Maryam于2013年在图形处理单元方面实践CA算法^[7]……教授及其学生的研究成果现已涵盖多数主流算法,但GCR算法的相关工作仍是空白,目前国内外都没有相应方案。

3 算法描述

3.1 原始算法

2008年曹建文等人对GRAPES模式中亥姆霍兹方程的系数矩阵进行ILU分解,形成了“带预条件的广义

共轭余差法”(Preconditioned Generalized Conjugate Residual method, P-GCR)^[8],目前已应用于业务运行。在本文中,以该算法为原始算法。其具体算法如下:

算法1 带预条件的广义共轭余差法(P-GCR)

待解方程组 $Ax=b$, 解的初猜值 x_0 ,

余差 r_i , 下降方向向量 p_i ,

预条件矩阵 M^{-1} , 重启间隔 s ,

系数 α_k, β_j^k

$x_0, r_0 = b - Ax_0, z_0 = M^{-1}r_0, p_0 = z_0$

while not converged do

1. for $k=0:s-1$, do

2. $\alpha_k = r_k^T A p_k / p_k^T A^T A p_k$

3. $x_{k+1} = x_k + \alpha_k p_k$

4. $r_{k+1} = r_k - \alpha_k A p_k \Leftrightarrow$

$$z_{k+1} = z_k - \alpha_k M^{-1} A p_k$$

if converged, exit

5. $\beta_j^k = -z_{k+1}^T A^T A p_j / p_j^T A^T A p_j$

($j=0, 1, \dots, k$)

6. $p_{k+1} = z_{k+1} + \sum_{j=0}^k \beta_j^k p_j$

7. end for

8. $r_0 = r_s, p_0 = p_s$

end while

算法1中, A, M^{-1} 是 n 阶方阵(n 的大小取决于具体算例), x, r, p, z 是长度为 n 的向量。

算法1中第2步的 $p_k^T A^T A p_k$ 和第5步的 $z_{k+1}^T A^T A p_j$ 分别是一次长向量点乘计算。长向量的点乘需要进行并行化计算,各个进程计算结果进行全局范围的求和,由此要各产生一次全局通信,因此每 s 步迭代中(本文中取 $s=10$),将进行 $2 \times s$ 次全局通信。并且通信间存在严格的依赖关系,无法通过合并而减少。

3.2 CA-P-GCR算法及推导

CA算法的基本思路:

(1)证明迭代过程中的长向量都存在于同一个Krylov子空间中,因此可以用空间下一组坐标(短向量)来代替长向量。

(2)把短向量代入原始算法,用短向量的迭代取代相应长向量的迭代,以此实现迭代过程中的通信避免。

CA-P-GCR算法的推导:

定义Krylov子空间

$$K_{s+1}(A, v) = \text{span}\{v, Av, A^2v, \dots, A^s v\}$$

(v 为任意向量)

在算法1中,当 $k=0$ 时,由第3步得:

$$x_1 - x_0 \in K_1(M^{-1}A, p_0)$$

由第4、6步得:

$$z_1 \in K_2(M^{-1}A, p_0),$$

$$p_1 \in K_2(M^{-1}A, p_0)$$

同理, 当 $k=s$ 时, 有结论:

$$\begin{cases} x_s - x_0 \in K_s(M^{-1}A, p_0) \\ z_s, p_s \in K_{s+1}(M^{-1}A, p_0) \end{cases} \quad (1)$$

选取适当向量建立基

$$V_{n, 2s+1} = [v_{p0}, v_{p1}, \dots, v_{ps}, v_{z0}, \dots, v_{z(s-1)}]$$

其中, 选取 $v_{p0} = p_0, v_{z0} = z_0$

v_{pi} 和 v_{zi} 由下式产生:

$$v_{p(z)i} = \alpha M^{-1}A v_{p(z)(i-1)} + \beta v_{p(z)(i-1)} + \gamma v_{p(z)(i-2)}$$

其中 $i \geq 2$ 。

α, β, γ 的值依所采用的不同基而定, 较复杂的基(如切比雪夫基、牛顿基)之间具有更低的相关性, 可以支持更高的重启间隔 s , 但同时在计算 α, β, γ 时也会带来一定的计算量。根据实际情况, 在本例中, 选取最简单的单项式基, 即: $\alpha = 1, \beta = 0, \gamma = 0$

根据结论式(1), 可以设:

$$z_i = Vm_i, p_i = Vn_i, x_i - x_0 = Vl_i$$

其中, 短向量 m_i, n_i, l_i 的长度为 $2s+1$ 。

将以上设定代入原算法中, 代入第2步得:

$$\alpha_k = m_k^T V^T M^T A V n_k / n_k^T V^T A^T A V n_k$$

代入第5步得:

$$\beta_j^k = -m_{k+1}^T V^T A^T A V n_j / n_j^T V^T A^T A V n_j$$

设有 Gram 矩阵:

$$G = V^T A^T A V$$

$$G_m = V^T M^T A V$$

于是:

$$\alpha_k = m_k^T G_m n_k / n_k^T G n_k$$

$$\beta_j^k = -m_{k+1}^T G n_j / n_j^T G n_j$$

其中 G, G_m 均为 $2s+1$ 阶方阵。

由第3步得:

$$x_{k+1} = x_k + \alpha_k p_k \Leftrightarrow$$

$$Vl_{k+1} = Vl_k + \alpha_k Vn_k \Leftrightarrow$$

$$V(l_{k+1} - l_k - \alpha_k n_k) = 0$$

根据生成矩阵 A 的气象学原理, 当 $p_0 \neq z_0$ 时, V 的各分量线性无关, 方程有唯一零解; 当 $p_0 = z_0$ 时, V 的秩为 $s+1$, 短向量的相应坐标也保持相等, 方程相当于一个 $s+1$ 列方程, 同样只有唯一零解, 因此有:

$$l_{k+1} = l_k + \alpha_k n_k$$

同理, 由第6步得:

$$p_{k+1} = z_{k+1} + \sum_{j=0}^k \beta_j^k p_j \Leftrightarrow$$

$$Vn_{k+1} = Vm_{k+1} + \sum_{j=0}^k \beta_j^k Vn_j \Leftrightarrow$$

$$n_{k+1} = m_{k+1} + \sum_{j=0}^k \beta_j^k n_j$$

由第4步得:

$$z_{k+1} = z_k - \alpha_k M^{-1}A p_k \Leftrightarrow$$

$$Vm_{k+1} = Vm_k - \alpha_k M^{-1}A Vn_k$$

设有换基矩阵 T 满足 $M^{-1}A V = VT$, 得:

$$Vm_{k+1} = Vm_k - \alpha_k VTn_k \Leftrightarrow$$

$$m_{k+1} = m_k - \alpha_k Tn_k$$

其中, 换基矩阵 T 在单项式基下为:

$$T = \left[\begin{array}{ccc|ccc} 0 & & & & & \\ 1 & 0 & & & & \\ & \dots & \dots & & & \\ & & & 1 & 0 & \\ \hline & & & & 0 & \\ O_{s,s+1} & & & & 1 & \dots \\ & & & & & 1 & 0 \end{array} \right]$$

其广义形式根据公式 $v_{p(z)i} = \alpha M^{-1}A v_{p(z)(i-1)} + \beta v_{p(z)(i-1)} + \gamma v_{p(z)(i-2)}$ 而定。

经过这样的替代, 在迭代过程中, 不再有长向量出现(只在计算 Gram 矩阵过程中出现长向量), 于是, “通信避免的带预条件的广义共轭余差法”(Communication Avoiding Preconditioned Generalized Conjugate Residual method, CA-P-GCR, 以下简称 CA 算法)描述如下:

算法2 通信避免的带预条件的广义共轭余差法

$$x_0, r_0 = b - Ax_0, z_0 = M^{-1}r_0, p_0 = z_0$$

while not converged do

1. Calculate

$$V = [p_0, M^{-1}A p_0, (M^{-1}A)^2 p_0, \dots,$$

$$(M^{-1}A)^s p_0, z_0, M^{-1}A z_0, \dots, (M^{-1}A)^{s-1} z_0]$$

2. Let $G = V^T A^T A V$

$$G_m = V^T M^T A V$$

3. $m_0 = [0_{s+1}, 1, 0_{s-1}]^T$

$$n_0 = [1, 0_{2s}]^T$$

$$l_0 = [0_{2s+1}]^T$$

4. for $k=0:s-1$, do

5. $\alpha_k = m_k^T G_m n_k / n_k^T G n_k$

6. $l_{k+1} = l_k + \alpha_k n_k$

7. $m_{k+1} = m_k - \alpha_k Tn_k$

8. $\beta_j^k = -m_{k+1}^T G n_j / n_j^T G n_j$

9. $n_{k+1} = m_{k+1} + \sum_{j=0}^k \beta_j^k n_j$

10. end for

11. $z_s = Vm_s, p_s = Vn_s, x_s = x_0 + Vl_s$

12. $z_0 = z_s, p_0 = p_s$

end while

算法2中, 0_s 表示 s 个实数0组成的行向量, 只有在

第2步与第3步之间进行一次全局通信,即每 s 次迭代进行一次全局通信。

为减少计算量,本文又对算法2中第11、12步修改如下:

$$11. x_s = x_0 + VI_s$$

$$12. r_0 = b - Ax_s, z_0 = M^{-1}r_0, p_0 = z_0$$

修改前,第12步新生成的 $z_0 \neq p_0$,其后的第1、2步的计算过程中要对 z_0 、 p_0 分别进行稀疏矩阵向量乘、点乘。修改后,第12步新生成的 $z_0 = p_0$,其优点是大幅减少了后续第1、2步的计算量。相应的代价是总迭代次数会有小幅增加,因为这样修改之后的算法和原算法在数学上已经不再等价。但是由于CA算法的迭代次数只能是 s 的整数倍,实际运行后发现,实际的迭代次数并没有因为以上修改而增加。

3.3 解的唯一性与正确性

待解方程 $Ax=b$ 的解的唯一性由 A 、 b 的属性决定,而 A 、 b 的生成依赖于相关的气象学原理。根据相关气象学原理可以确定,待解方程的解存在且唯一。

新算法与旧算法求得的方程解并不完全一致,其原因至少包括以下2条:

- (1)两种算法迭代次数不同。
- (2)算法2中对第11、12步的修改。

其中,原因(1)在所有CA类算法中普遍存在。但是模式动力框架对方程求解器模块的要求,并不是不同算法解的一致,而是仅仅要求收敛。后续的动力框架部分和物理过程部分的计算,也都仅以方程求解器的收敛为前提。

3.4 新旧算法计算量对比

对比原算法和CA-P-GCR算法,计算量对比如表1。

表1 新旧算法的计算量

	P-GCR	CA-P-GCR
稀疏矩阵向量乘	$2 \times s$	$s+1$
点乘	$0.5 \times s^2 + 1.5 \times s$	$0.5 \times s^2 + 2.5 \times s$
全局通信	$2 \times s$	1

对比发现:每 s 步迭代(即每个restart周期),CA-P-GCR中全局通信的次数减少为原算法的 $1/(2 \times s)$ 。稀疏矩阵向量乘减少了约一半,点乘计算量增加 s 次。按照以往测试结果,稀疏矩阵向量乘的计算量远高于点乘。因此预测CA算法将在计算、通信两方面同时优于原算法。同时,相比原算法,计算部分更加集中,存在进行访存优化、提高计算访存比的潜力。

4 数值实验

4.1 算例信息

本测试的测试对象是中国气象局的新一代“全球

区域一体化数值预报模式(GRAPES)”,其动力框架的核心是求解亥姆霍兹方程 $Ax=b$,本测试共运行288步,每步进行1次方程求解,每次求解过程中, A 、 b 不变, x 初猜值继承上一次的最终结果(第一次初猜值取0向量)。本测试采用 0.05° 水平分辨率的全球算例,垂直层数60层,时间步长150 s,物理过程关闭。 A 是 1.56×10^9 阶方阵, x 、 b 是长度为 1.56×10^9 的向量。因为 A 、 x 、 b 以及迭代过程中出现的 z 、 p 、 r 规模较大,所以计算时按节点分别存储。 A 是稀疏矩阵,每一行中有19个非0元。

4.2 计算集群信息

本测试所用计算集群是中国气象局2018年部署的“曙光-派”计算集群,采用Intel的SKL Gold 6142处理器,主频2.6 GHz,32核/节点,每个核心运行1个进程,纯MPI并行。每节点内存为192 GB的DDR4内存,操作系统RHEL 7.4,编译器Intel PSXE 2017u2。

5 实验结果

5.1 收敛速率对比

测试中,每种算法各进行方程求解288次,每次求解的迭代次数各不相同。经统计得:CA算法平均迭代次数为34次,原算法平均迭代次数为28次。现以4 096进程下一次典型求解过程为例,如图1所示。

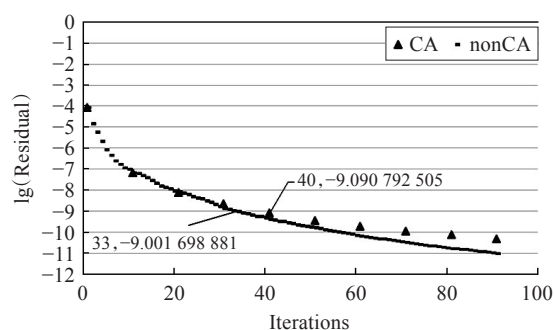


图1 新旧算法收敛速率

说明:本文以 r_0 全体元素的平方和作为残差(由于三维空间格点总数不变,因此没有进行平均、开方运算),收敛阈值定为 1.0×10^{-9} ,图1中纵坐标为残差的常用对数。由图1可知,本次求解过程中,随着迭代次数的增加,CA算法的收敛速率略慢于原算法。同时,CA算法的迭代次数为不连续的、10的整数倍。原算法达到收敛阈值的迭代次数是33次,CA算法达到收敛阈值的次数是40次。

在整个测试中,CA算法的平均迭代次数高于原算法的原因主要有以下3条:

- (1)CA算法的迭代次数只能是 s 的整数倍(本文中 s 全部取10)。
- (2)所有CA算法由于基的条件数的限制,都存在最

高收敛精度降低,收敛速率小幅变慢的问题。Carson于2014年针对各种“通信避免算法”中出现的这类问题进行了详细分析,并给出了“余差替代”解决方案^[9]。本文所提出的算法也存在相同的问题,但由于“余差替代”方案开销较大,因此没有采纳。

(3)前文中修改第12步,改变生成新的 z_0 、 p_0 的方法,使得 z_0 、 p_0 相等。这就改变了基的条件数,进一步减慢了收敛速率。

由于要兼顾通信量和计算量, s 的选取受到一定限制,使得以上(2)、(3)两条的效果被(1)所覆盖,因此(2)、(3)两条并没有实质上增加总的迭代次数^[10]。

5.2 总时间对比

本测试并行规模从2 048进程到16 384进程,GRAPES模式的整体运行时间(图2)和亥姆霍兹方程求解器的运行时间及加速比(图3)如下所示。

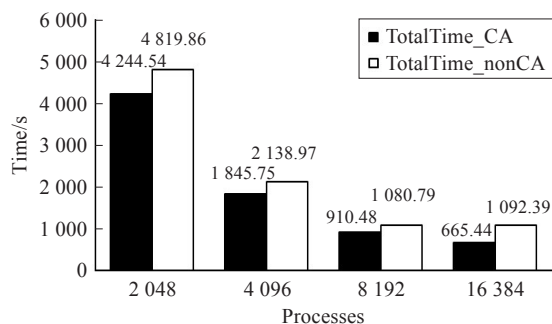


图2 新旧算法的模式总运行用时及加速比

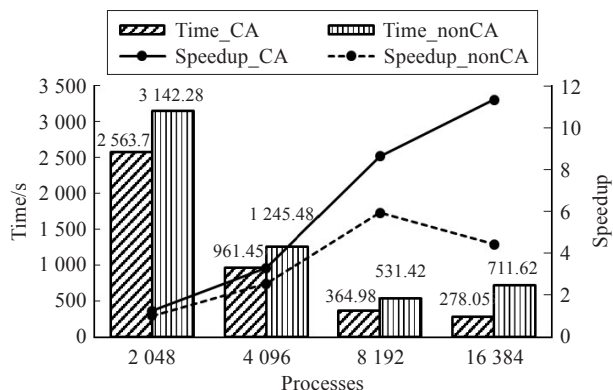


图3 新旧算法的求解器运行用时及加速比

对比图2和图3可知,方程求解器的时间消耗占据了整个数值模式的重要部分(从42%到65%)。求解器之外的部分并没有做修改,因此差别不明显。求解器算法的优化,导致了模式整体运行速率最高1.64倍的加速比(相对于同规模下的原算法)^[11]。

由表1可知,CA算法在计算量、通信量上都优于原算法,从图3也可知从2 048进程到16 384进程上,CA算法的用时都少于原算法,并且这种优势随着并行规模扩展而扩大。从图3的加速比可以看出,当规模扩展到16 384进程时,CA算法的加速比达到了原算法的3

倍^[12]。对比新旧算法求解器部分的可扩展性,两种算法在规模较小时都表现出良好的可扩展性,当规模较大时,CA算法仍然保持可扩展性,而原算法随规模扩展而速度降低。

5.3 主要计算、通信部分时间对比

并行规模较小时,计算开销(包括稀疏矩阵向量乘、点乘)占据主要部分,其用时远高于通信。并且随着规模的扩展,计算部分的加速呈现“超线性”^[13],例如:4 096进程的并行规模是2 048进程的2倍,但稀疏矩阵向量乘用时仅为2 048进程的1/4~1/3。推测其主要原因是规模较小情况下,内存压力较大,导致访存命中率偏低。随着并行规模扩大,通信用时逐渐占据了主要部分,CA算法的优势主要体现在通信上,16 384进程下,原算法的通信开销已经远远超过了计算开销。

另外,对比图2、图4,随着并行规模的扩展,原算法的全局通信用时最高占据了模式整体运行时间的50%,由此说明了通信避免的必要性。

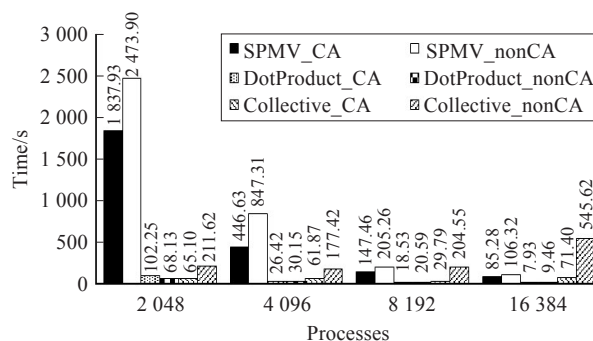


图4 新旧算法的主要计算及通信用时

6 结论

GRAPES模式使用原算法的亥姆霍兹方程求解器受限于全局通信,其扩展能力止步于8 192进程。本文通过通信避免算法改造,以小幅降低收敛速率为代价,同时改善了求解器的运行速率和扩展能力,在16 384规模平台上的实验显示出3倍于原算法的加速效果^[14]。

综合以上结果可知:CA算法可以极大地改善方程求解器的可扩展性,并且减少运算、通信的时间开销,在各种计算规模下都有着相比于原算法明显的优势^[15]。在较小规模下的优势主要来源于计算的减少,在较大规模下的优势主要来源于全局通信的减少。同时,会导致收敛速率的小幅降低。

参考文献:

- [1] 徐树方. 矩阵计算的理论与方法[M]. 北京: 北京大学出版社, 1995: 183-187.
- [2] 沈桐立. 数值天气预报[M]. 北京: 气象出版社, 2015: 398.

- [3] Demmel J, Hoemmen M F, Mohiyuddin M, et al. Avoiding communication in computing Krylov subspaces, UCB/EECS-2007-123[R]. California US: University of California, Berkeley. EECS Department, 2007.
- [4] Demmel J, Grigori L, Xiang H. Communication-avoiding Gaussian elimination[C]//Proceedings of the 2008 ACM/IEEE Conference on Supercomputing, 2008: 1-12.
- [5] Demmel J, Grigori L, Xiang H. A communication optimal LU factorization algorithm, EECS-2010-29[R]. California US: University of California, Berkeley. EECS Department, 2010.
- [6] Hoemmen M. Communication-avoiding Krylov subspace methods[D]. California US: UC Berkeley. Computer Science Division, 2010.
- [7] Maryam M. Communication-avoiding Krylov techniques on graphic processing units[J]. IEEE Transactions on Magnetics, 2013, 49(5): 1749-1752.
- [8] 刘宇, 曹建文. 适用于 GRAPES 数值天气预报软件的 ILU 预条件子[J]. 计算机工程与设计, 2008, 29(3): 731-734.
- [9] Carson E, Demmel J. A residual replacement strategy for improving the maximum attainable accuracy of s-step Krylov subspace methods[J]. SIAM Journal on Matrix Analysis and Applications, 2014, 35(1): 22-43.
- [10] 朱宇兰. 基于 GPU 通用计算的并行算法和计算框架的实现[J]. 山东农业大学学报(自然科学版), 2016, 47(3): 473-476.
- [11] 赵旭东, 梁书秀, 孙昭晨, 等. 基于 GPU 并行算法的水动力数学模型建立及其效率分析[J]. 大连理工大学学报, 2014(2): 204-209.
- [12] 刘永和, 冯锦明, 徐文鹏. 分布式水文模型的 GPU 并行化及快速模拟技术[J]. 水文, 2015(4): 20-26.
- [13] 董莹, 葛万成, 陈康力. CUDA 并行计算的应用研究[J]. 信息技术, 2010(4): 11-15.
- [14] 谭郁松, 伍复慧, 吴庆波, 等. 面向 CPU/MIC 异构架构的 K-Means 向量化算法[J]. 计算机科学与探索, 2014, 8(6): 641-652.
- [15] 周洪斌. 基于 OpenMP 求解 QAP 的并行粒子群优化算法[J]. 微型机与应用, 2010, 29(10): 84-86.

《计算机科学与探索》投稿须知

《计算机科学与探索》是由中国电子科技集团公司主管、华北计算技术研究所主办的国内外公开发行的高级学术期刊, 中国计算机学会会刊, 中国百强科技期刊, 工业和信息化部优秀科技期刊, 北大中文核心期刊, 中国科学引文数据库(CSCD)核心期刊, 中国科技论文统计源期刊(中国科技核心期刊), 中国学术期刊综合评价数据库(CAJCED)统计源期刊, 中国科技论文与引文数据库(CSTPCD)统计源期刊, 中文科技期刊数据库(VIP)收录期刊, 并被“英国《科学文摘》(SA/INSPEC)”“美国《剑桥科学文摘》(CSA)”“美国《乌利希期刊指南》(Ulrich's PD)”“《日本科学技术振兴机构中国文献数据库》(JST)”“波兰《哥白尼索引》(IC)”收录, 入选“庆祝中华人民共和国成立 70 周年精品期刊展”。月刊, 大 16 开。欢迎踊跃投稿。

刊登内容

高性能计算机、体系结构、并行处理、计算机科学新理论、算法设计与分析、人工智能与模式识别、系统软件、软件工程、数据库、计算机网络、信息安全、计算机图形学与计算机辅助设计、虚拟现实、多媒体技术及交叉学科的相互渗透和新理论的衍生等(如: 认知科学、神经信息学、量子信息学、生物信息学等)。

投稿要求

1. 稿件必须具有原创性、学术性、科学性、准确性、规范性和可读性。
2. 以稿件表达完整为准, 不硬性规定稿件字数。
3. 本刊网站(<http://fcst.ceaj.org>)提供了“论文写作模板”, 作者可参考进行撰写。
4. 采用学术论文标准格式书写, 要求文笔简练、流畅, 文章结构严谨完整、层次清晰。论文包括: 标题、作者(含电子信箱)、单位、摘要、关键词、基金资助情况、作者简介、中图分类号、正文、参考文献等, 其中前 7 项必须中英文齐全。
5. 来稿应做到定稿、定图, 应具有准确的图序和图注。
6. 清晰列出国内外参考文献。若为中文参考文献, 必须同时将其译成英文。
7. 作者在投稿时须注明是否为 CCF 会员, 若是会员, 请注明会员号。凡第一作者为 CCF 会员者, 享受版面费 85 折优惠。
8. 请提供联系人详细的通信方式, 包括姓名、通信地址、邮编、E-mail、电话等。通信方式如有变化, 请及时通知编辑部。

投稿方法

1. 通过本刊网站“作者投稿”一栏进行投稿。投稿之后, 若需查询稿件处理状况, 请登录到本刊网站, 在“作者查稿”一栏中查询。
2. 本刊网站提供“稿件处理流程”, 作者可以通过该流程了解稿件的处理程序和相关规定以及时间进度。

高水平的综述和探索类文章将优先发表!