块三对角线性代数方程组的 GPU 加速求解

一、研究背景

计算流体力学（Computational Fluid Dynamics，简称 CFD）是 21 世纪流体力学领域的重要技术之一，CFD 相当于“虚拟”地在计算机做实验，用以模拟仿真实际的流体流动情况。而其基本原理则是数值求解控制流体流动的微分方程，得出流体流动的流场在连续区域上的离散分布，从而近似模拟流体流动情况。

随着计算机技术的推广普及和计算方法的新发展，几十年来 CFD 技术取得了蓬勃的发展。由于数值模拟相对于实验研究有很独特的优点，比如成本低，周期短，能获得完整的数据，能模拟出实际运行过程中各种所测数据状态，对于设计、改造等商业或实验室应用起到重要的指导作用。近年来，随着计算机容量的大大提高、先进的数值计算技术的出现以及各种湍流模型的提出，CFD 技术已经广泛应用于航空航天、化工、能源、环境、水利工程、气象预测等领域。美国海空军下一代 F-35 战斗机所使用的附面层分离进气道是 CFD 的成果之一。附面层分离进气道通过特殊设计形状的突起分离流速较慢的附面层以改善涡轮风扇发动机的进气流场。此设计比传统的附面层隔板方法可以减轻数百公斤重量，同时在一定速度范围内能够维持很好的分离效率。

二、数学问题描述

基于连续介质假设的计算流体力学控制方程是非线性偏微分方程组，必须进行数值求解。其本质是偏微分方程经由数值方法离散成一个线性代数系统的求解。在实际的工程背景下，为了避免离散得到的大型稀疏矩阵所增加的存储开销，往往通过对矩阵元素的重新排列，将其重组为块状或带状矩阵。以二维系统为例，在如图 1 所示的网格化计算区域上（假设 x 方向有 m 个节点，y 方向有 n 个节点）

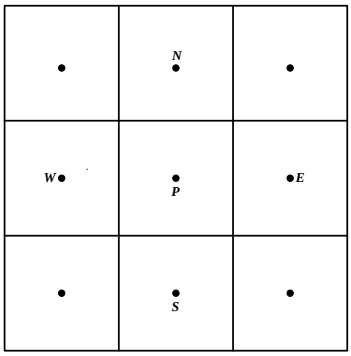

图 1 计算区域网格化

运用偏微分方程离散化方法，通用变量 ϕ 在任一点 P 处将满足如下离散方程

$$c_P\phi_P = c_E\phi_E + c_W\phi_W + c_S\phi_S + c_N\phi_N + d_P$$

其中，E、W、S、N 分别表示与 P 点相邻的东、西、南、北计算节点，c 和 d 分别表示系数和常数项。

将待求变量沿 y 方向逐行存储，对应的未知向量可表示为

$$\vec{X}^T = (\vec{x}_1^T, \vec{x}_2^T, \cdots, \vec{x}_n^T)$$

其中

$$\vec{x}_j^T = (\phi_{1,j}, \phi_{2,j}, \cdots, \phi_{m,j})$$

对应的离散方程组系数矩阵

$$M = \begin{pmatrix} B_1 & C_1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ A_2 & B_2 & C_2 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & A_{n-1} & B_{n-1} & C_{n-1} \\ 0 & 0 & 0 & \cdots & 0 & 0 & A_n & B_n \end{pmatrix}$$

是一个块三对角矩阵，每一行有五个非零元素，其中子矩阵 A、C 是对角矩阵，B 是三对角矩阵

$$A_j = \begin{pmatrix} a_{1,1}^{(j)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & a_{2,2}^{(j)} & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & a_{m-1,m-1}^{(j)} & 0 \\ 0 & 0 & \cdots & 0 & 0 & a_{m,m}^{(j)} \end{pmatrix}$$

$$C_j = \begin{pmatrix} c_{1,1}^{(j)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & c_{2,2}^{(j)} & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & c_{m-1,m-1}^{(j)} & 0 \\ 0 & 0 & \cdots & 0 & 0 & c_{m,m}^{(j)} \end{pmatrix}$$

$$B_j = \begin{pmatrix} b_{1,1}^{(j)} & b_{1,2}^{(j)} & 0 & 0 & \cdots & 0 & 0 & 0 \\ b_{2,1}^{(j)} & b_{2,2}^{(j)} & b_{2,3}^{(j)} & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & b_{m-1,m-2}^{(j)} & b_{m-1,m-1}^{(j)} & b_{m-1,m}^{(j)} \\ 0 & 0 & 0 & \cdots & 0 & 0 & b_{m,m-1}^{(j)} & b_{m,m}^{(j)} \end{pmatrix}$$

对应的离散方程组常数向量可表示为

$$\vec{G}^T=(\vec{g}_1^T,\vec{g}_2^T,\cdots,\vec{g}_n^T)$$

其中

$$\vec{g}_j^T=(d_{1,j},d_{2,j},\cdots,d_{m,j})$$

综上所述，离散化线性代数方程组可表示为

$$M\vec{X}=\vec{G}$$

将上述结果推广至三维情形（假设 z 方向上有 k 个节点），其中待求变量沿 z 方向逐面存储，而在每个平面上沿 y 方向逐行存储，对应的未知向量可表示为

$$\vec{X}^T=(\vec{x}_1^T,\vec{x}_2^T,\cdots,\vec{x}_k^T)$$

其中

$$\vec{x}_j^T=(\vec{y}_{1,j}^T,\vec{y}_{2,j}^T,\cdots,\vec{y}_{n,j}^T)$$

$$\vec{y}_{i,j}^T=(\phi_{1,i,j},\phi_{2,i,j},\cdots,\phi_{m,i,j})$$

对应的离散方程组系数矩阵

$$M=\begin{pmatrix}
B_1 & C_1 & 0 & 0 & \cdots & 0 & 0 & 0\\
A_2 & B_2 & C_2 & 0 & \cdots & \cdots & \cdots & 0\\
0 & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots\\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots\\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots\\
\vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0\\
0 & \cdots & \cdots & \cdots & 0 & A_{k-1} & B_{k-1} & C_{k-1}\\
0 & 0 & 0 & \cdots & 0 & 0 & A_k & B_k
\end{pmatrix}$$

是一个块三对角矩阵，每一行有七个非零元素，其中子矩阵 A、C 是对角矩阵。

$$A_j=\begin{pmatrix}
a_{1,1}^{(j)} & 0 & 0 & \cdots & 0 & 0\\
0 & a_{2,2}^{(j)} & 0 & \cdots & \cdots & 0\\
\vdots & \ddots & \ddots & \ddots & \vdots & \vdots\\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots\\
0 & \cdots & \cdots & 0 & a_{m\times n-1,m\times n-1}^{(j)} & 0\\
0 & 0 & \cdots & 0 & 0 & a_{m\times n,m\times n}^{(j)}
\end{pmatrix}$$

$$C_j=\begin{pmatrix} c_{1,1}^{(j)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & c_{2,2}^{(j)} & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & c_{m\times n-1,m\times n-1}^{(j)} & 0 \\ 0 & 0 & \cdots & 0 & 0 & c_{m\times n,m\times n}^{(j)} \end{pmatrix}$$

需要特别指出的是，*B* 与二维系统对应的系数矩阵 *M* 具有相同的结构，也是一个块三对角矩阵，此处不再赘述。

对应的离散方程组常数向量可表示为

$$\vec{G}^T=(\vec{g}_1^T,\vec{g}_2^T,\cdots,\vec{g}_k^T)$$

其中

$$\vec{g}_j^T=(\vec{f}_{1,j},\vec{f}_{2,j},\cdots,\vec{f}_{n,j})$$

$$\vec{f}_{i,j}^T=(d_{1,i,j},d_{2,i,j},\cdots,d_{m,i,j})$$

国际上对三对角线性代数方程组并行算法的研究始终非常活跃，是数值并行计算方法中最重要的问题之一。这些算法可推广应用于上述块三对角线性方程组的求解。

三、三对角线性代数方程组的求解

总结以往三对角线性代数方程组的求解算法，可归纳为以下三类[1]：(1) 递推倍增（Recursive Doubling，简称 RD）算法[2]，(2) 循环约化（Cyclic Reduction，简称 CR）算法[3]，(3) 矩阵分解算法。RD 算法和 CR 算法是适用于向量计算机或共享内存并行机的并行算法。由 Wang[4]提出的分裂法属于矩阵分解算法，其基于分而治之的原则，适用于分布存储环境，受到广泛关注。

1. RD 算法

Egecioglu 等[5]将 Stone[2]提出的 RD 算法表示成扫描（前加和）形式，其适用于向量计算机，易于在 GPU 上实现。Zhang 等[6]选用 Hillis 和 Steele[7]提出的步数有效扫描算法在 GPU 上实现了 RD 算法。

2. CR 算法

CR算法[3]包括消去和回代两个步骤。消去步骤逐次将系统变量减半，直至仅存两个未知变量。回代步骤则利用当前已知变量逐次计算出其他未知变量。近期Hirshman等[8]基于CR算法开发了一个并行求解器BCYCLIC，其在多核平台上可扩展至数百个处理器，节点间通过MPI进行数据传输。并行循环约化（Parallel Cyclic Reduction，简称PCR）算法[9]是CR算法的一个变种。较之CR算法，PCR算法仅有消去步骤。

3. 耦合算法

Zhang 等[6]详细比较了 RD，CR 和 PCR 三种算法。CR 算法是计算量有效算法，即每一迭代步仅执行 $17n$ 次浮点数操作，但其需要进行 $2\log_2 n-1$ 次迭代。PCR 算法和 RD 算法是步数有效算法。PCR 算法每一迭代步执行 $12n\log_2 n$ 次浮点数操作，仅需要进行 $\log_2 n$ 次迭代。RD 算法每一迭代步执行 $20n\log_2 n$ 次浮点数操作，仅需要进行 $\log_2 n+2$ 次迭代。

为了提高计算效率，Zhang 等[6]提出并验证了基于上述三个算法的混合算法。

4. 改进的 Wang 分裂法

Michielse 和 vander Vorst[10]改进了 Wang 的分裂法，减少了通信开销，提高了计算与通信的重叠程度。考虑减少计算量，降低通信复杂性，迟利华和李晓梅[11]进一步改进了 Michielse 和 vander Vorst[10]的分裂算法，减少了通信次数的建立和总数据的传输量，充分利用计算与通信的重叠，提出了双向并行分裂法（DPP）算法，提高了算法的并行效率。

5. BICG 算法的并行化

骆志刚[12]研究了块三对角线性代数方程组的分布式并行求解，提出了一种调用 BLAS3 子程序且基于块运算的并行算法，该算法将方程组求解工作量合理地分配到各处理机，达到负载平衡，取得较高的并行效率。他还研究了非对称大型稀疏线性代数方程组的分布式并行求解，在提出修订的 BICG 算法的基础上，提出了一种适于并行计算的 *s*-BICG 算法。由于 BICG 算法是 Krylov 子空间迭代法中 Lanczos 双正交化方法类的基础算法，*s*-BICG 算法的提出，可为进一步研究 Lanczos 双正交化方法类的 *s*-step 推广奠定基础。

四、块三对角线性代数方程组的 GPU 求解

在 GPU 上实现上节所述的求解算法需要考虑并行扩展性和计算性能等问题。CUDA 提供了 BLAS 库（CUBLAS），方便使用者对稠密矩阵进行高效运算，用于实现线性系统的直接求解方法（如 LU 分解、Cholesky 分解等），但对于像块三对角阵这样的稀疏矩阵而言，目前还没有一个高效的 GPU 求解器。

另外一类求解方法就是迭代方法，以限定迭代次数或限定判据来逼近精确解。一般而言，迭代法具有良好的可扩展性，计算量与矩阵阶数近似成线性关系。迭代方法主要有最简单的 Jacobi 迭代法、Gauss-Seidel 迭代法或者超松弛迭代法（SOR）以及共轭梯度法（CG）等，这些方法在 GPU 上均容易实现，也有很多研究者进行了探索和开发。对于主元占优的块三角阵，如何在 GPU 上实现收敛较快及性能高效的迭代方法需要重点关注以下几方面：a）迭代中所用的数据有序读写和复用；b）以元素为操作单位的细粒度并行；c）分块时的块间数据交换；d）更深入的多 GPU 并行计算。

参考文献

[1] 迟利华. 大型稀疏线性方程组在分布式存储环境下的并行计算. 博士学位论文. 长沙: 国防科技大学研究生院, 1998

[2] H. S. Stone. An efficient parallel algorithm for the solution of a tridiagonal linear system of equations. *Journal of the ACM*, 1973, 20(1): 27–38

[3] R. W. Hockney. A fast direct solution of Poisson’s equation using Fourier analysis. *Journal of the ACM*, 1965, 12(1): 95–113

[4] Wang H H, A Parallel method for tridiagonal equations. ACM Trans. on Math. Software, 1981, 7(2): 170–183

[5] Ö. Egecioglu, C. K. Koc, A. J. Laub. A recursive doubling algorithm for solution of tridiagonal systems on hypercube multiprocessors. *Journal of Computational and Applied Mathematics*, 1989, 27: 95–108

[6] Y. Zhang, J. Cohen, J. D. Owens. Fast tridiagonal solvers on the GPU. In: PPoPP ’10: Proceedings of the 15th ACM SIGPLAN symposium on principles and practice of parallel programming. ACM, New York, 2010: 127–136

[7] W. D. Hillis, G. L. Steele Jr. Data parallel algorithms. *Communications of the ACM*, 1986, 29(12): 1170–1183

[8] S. Hirshman, K. Perumalla, V. Lynch, and R. Sanchez, Bcyclic: A parallel block tridiagonal matrix cyclic solver. *Journal of Computational Physics*, 2010, 229(18): 6392–6404

[9] R. W. Hockney, C. R. Jesshope. Parallel Computers. Adam Hilger, Bristol, 1981

[10] Michielse P, van der Vorst. Data transport in Wang's partition method. *Paralel Computing*, 1988, 7(l): 87–95

[11] 迟利华, 李晓梅. 三对角线性方程组的分布式并行算法. 计算机研究与发展, 1998, 35(11): 1004–1007

[12] 骆志刚. 典型结构大型线性方程组的分布式并行算法研究. 博士学位论文. 长沙: 国防科学技术大学研究生院, 2000