

基于预处理 BICGSTAB 法的电力系统潮流并行计算方法

宋晓喆¹, 魏国², 李雪², 王长江², 孙福寿¹, 李振元¹

(1. 国网吉林省电力有限公司, 吉林 长春 130000; 2. 东北电力大学电气工程学院, 吉林 吉林 132012)

摘要: 为实现大规模电力系统潮流的准确、快速求解, 以非精确牛顿法为基础, 提出一种基于 CPU-GPU 异构平台的电力系统潮流并行计算方法。修正方程组的求解是牛拉法潮流计算中最为耗时的部分, 提升修正方程组的求解效率可有效提升潮流计算效率。为此, 根据雅可比矩阵的不对称不定性, 采用稳定双正交共轭梯度(bi-conjugate gradient stabilized, BICGSTAB)法进行修正方程组的求解。进一步, 为改善 BICGSTAB 法的收敛性, 根据雅可比矩阵的稀疏性和类对角占优性, 提出一种改进 PPAT(Preconditioner with sparsity Pattern of A^T , PPAT)预处理器和改进 Jacobi 预处理器相结合的两阶段预处理方法, 并对雅可比矩阵进行预处理, 提升 BICGSTAB 法的收敛性能。然后, 将上述潮流算法移植到 CPU-GPU 异构平台, 实现电力系统潮流的并行求解。最后, 通过不同测试系统算例对所提方法进行验证、分析。结果表明, 所提潮流并行计算方法可实现电力系统潮流的准确、快速求解。

关键词: 潮流计算; 非精确牛顿法; 雅可比矩阵; BICGSTAB 法; 预处理器; CPU-GPU 异构平台

Parallel power flow computing in power grids based on a preconditioned BICGSTAB method

SONG Xiaozhe¹, WEI Guo², LI Xue², WANG Changjiang², SUN Fushou¹, LI Zhenyuan¹

(1. State Grid Jilin Electric Power Co., Changchun 130000, China; 2. School of Electrical Engineering, Northeast Electric Power University, Jilin 132012, China)

Abstract: In order to solve the large-scale power flow problem accurately and quickly, a parallel algorithm for power flow calculation based on the inexact Newton and CPU-GPU heterogeneous platform is proposed. The efficiency of power flow computing can be effectively enhanced by improving the solving efficiency of the correction equations which are the most time-consuming part of the Newton-Raphson method. For this reason, the Bi-Conjugate Gradient Stabilized (BICGSTAB) method is adopted to solve the correction equations according to the asymmetric and indefinite characteristics of a Jacobian matrix. Then, in order to improve the convergence performance of the BICGSTAB method, a two-step preconditioner for the Jacobian matrix is proposed given the characteristic that the Jacobian matrix is analogous to a sparse diagonally dominant matrix. The convergence performance of the BICGSTAB method can be improved with the two-step preconditioner that consists of the improved Preconditioner with sparsity Pattern of A^T (PPAT) preconditioner and the improved Jacobi preconditioner. Next, the above power flow algorithm is transplanted to a CPU-GPU heterogeneous platform to achieve power flow parallel computing. Finally, different test systems are further used to verify and analyze the performance of the proposed method. The results validate that the proposed power flow algorithm can solve the power flow calculation problem accurately and quickly.

This work is supported by National Natural Science Foundation of China (No. 51607033 and No. 51677023).

Key words: power flow; inexact Newton method; Jacobian matrix; BICGSTAB method; preconditioner; CPU-GPU heterogeneous platform

0 引言

潮流计算是电力系统安全稳定分析、故障计算

以及经济调度等问题的基础^[1-4]。准确、快速的潮流计算结果是电网安全、可靠运行的重要保证。随着区域电网互联规模持续扩大、可再生能源大规模并网、电力电子设备大规模应用, 电力系统潮流计算的规模和复杂度急剧增加^[5-9]。同时, 电网智能化进程不断加深, 电力系统精细化运行的要求不断提高, 运行人员对电力系统潮流计算的精度和效率提出了更高的要求。

基金项目: 国家自然科学基金项目资助(51607033, 51677023); 国网吉林省电力有限公司科技项目资助“基于空间相关性的新能源送出关键断面安全性评估与控制研究”(SGJL0000DKJS 2000287)

牛拉法是目前采用最为广泛的电力系统潮流计算方法,该方法基于逐次线性逼近原理将非线性潮流方程的求解转化为对线性修正方程的求解^[10]。而修正方程组的求解是牛拉法中最为耗时的部分,约占整个潮流计算耗时的 80%^[11]。因此,减少修正方程组的求解耗时是提升电力系统潮流计算效率的关键。

当前,修正方程组的求解主要采用直接法^[12]和迭代法^[13]。直接法基于稀疏矩阵分解技术,逐一求解各修正量的解,此过程中各修正量间具有极强的数据相关性,不易实现修正方程组求解的并行化^[14]。同时,由于直接法在对雅可比矩阵进行分解时,存在非零元素的填充问题,当系统规模增大时,会导致数据存储量大幅增加,不利于大规模互联电力系统潮流的快速求解。相对直接法,迭代法在求解修正方程组时,通过不断修正解向量,使修正量的解向量逐步逼近潮流方程实际解向量。该过程主要涉及稀疏矩阵和向量间运算,数据相关性弱,具有良好的并行特性,适用于处理高维度修正方程组。

迭代法主要分为古典迭代法与 Krylov 子空间迭代法两类^[15]。古典迭代法收敛性较差,目前已较少应用;Krylov 子空间迭代法^[16]因其优良的收敛性与数值稳定性,已在潮流计算^[17-21]、小干扰分析^[22]、暂态稳定性分析^[23]等领域得到应用。但每种 Krylov 子空间法均有一定适用范围,采用 Krylov 子空间法求解潮流修正方程组时,针对雅可比矩阵的不对称不定特征,常用的 Krylov 子空间法为广义最小残差 (generalized minimal residual, GMRES) 法^[24]和 BICGSTAB 法^[25]。GMRES 法可较精确实现修正方程组的求解,且具有良好的收敛性,但在构造 Krylov 子空间基底时需要长递推式,系统规模增大时将导致迭代次数、数据存储量和计算量急剧增加,不利于高维度修正方程组的快速求解。相对 GMRSE 法, BICGSTAB 法以其存储空间稳定性和短递推特性等优势,已在高维度修正方程组的求解中得到应用。但采用 BICGSTAB 法求解修正方程组时,需对潮流雅可比矩阵进行预处理^[26],压缩其特征值分布区间,从而提高 BICGSTAB 法收敛性。目前,预处理器设计仍是制约 BICGSTAB 法应用于电力系统潮流并行计算的一个瓶颈。

针对上述问题,本文提出一种非精确牛顿法^[27]与两阶段预处理 BICGSTAB 法相结合的电力系统潮流并行计算方法。首先,根据牛拉法潮流计算中雅可比矩阵特性,采用 BICGSTAB 法进行潮流修正

方程组的求解,同时采用改进 PPAT 预处理器与改进 Jacobi 预处理器相结合的两阶段预处理方法对雅可比矩阵进行预处理,改善雅可比矩阵特征值分布,提升 BICGSTAB 法收敛性;然后,将整个算法移植到 CPU-GPU 异构平台, CPU 负责控制牛拉法整个计算流程, GPU 负责其中修正方程组的并行求解,充分发挥异构平台各自特点,实现电力系统潮流并行计算,以提高计算效率;最后,通过不同测试系统对所提算法准确性和有效性进行验证、分析。

1 非精确牛顿法潮流计算

电力系统潮流计算是指在电力系统正常运行方式下,根据给定的系统运行方式求解电力系统的稳态运行状态^[28]。牛拉法是目前采用广泛的潮流计算方法^[29-30],极坐标下电力系统节点功率方程为

$$\begin{cases} P_i = V_i \sum_{j=1}^n V_j (G_{ij} \cos \delta_{ij} + B_{ij} \sin \delta_{ij}) \\ Q_i = V_i \sum_{j=1}^n V_j (G_{ij} \sin \delta_{ij} - B_{ij} \cos \delta_{ij}) \end{cases} \quad (1)$$

式中: V_i 、 V_j 为节点 i 和 j 的电压幅值; δ_{ij} 为节点 i 与 j 的电压相角差; P_i 与 Q_i 分别为节点 i 的有功与无功注入量。

式(1)可描述为形如 $F(x)=0$ 的形式,其中 x 为状态变量 V 与 δ 组成的列向量。将式(1)按照泰勒级数展开,保留其线性部分,舍弃二次及以上高次项,得到线性修正方程组为

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & N \\ F & L \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta V/V \end{bmatrix} \quad (2)$$

式中: $H = \frac{\partial \Delta P}{\partial \delta}$; $N = \frac{\partial \Delta P}{\partial V} V$; $F = \frac{\partial \Delta Q}{\partial \delta}$; $L = \frac{\partial \Delta Q}{\partial V} V$ 。

式(2)可进一步描述为

$$A \Delta x = b, A \in R^{n \times n}, \Delta x, b \in R^n \quad (3)$$

式中: $A = \begin{bmatrix} H & N \\ F & L \end{bmatrix}$; $\Delta x = \begin{bmatrix} \Delta \delta \\ \Delta V/V \end{bmatrix}$; $b = \begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix}$ 。

通过求解式(3)所示修正方程组,可得牛拉法第 m 次迭代下的修正量 $\Delta x^{(m)}$,进而修正迭代变量 $x^{(m)}$:

$$x^{(m+1)} = x^{(m)} - \Delta x^{(m)} \quad (4)$$

在给定最大迭代次数内不断迭代求解式(3)中修正量 Δx ,然后根据式(4)修正 x ,使 x 逐步逼近式(1)所示潮流方程的真实解,直至 $F(x^{(m)})$ 满足收敛条

件,即完成潮流计算。

在上述牛拉法潮流计算过程中,式(3)所示修正方程组的求解是最为耗时的部分,减少该部分计算耗时可大幅提高潮流计算效率。传统牛拉法采用基于 LU 分解的直接法精确求解修正方程组,当系统规模较大时,直接法的数据存储量和计算量急剧增加,严重影响了潮流计算效率。实际上,每次求解修正方程组的目的并不是获取修正方程组的精确解,而是为了使牛拉法的迭代变量逐步逼近非线性潮流方程的真实解。因此,在保证牛拉法收敛前提下,可不必精确求解修正方程组,而是采用迭代求解法获取修正方程组的满足误差要求的近似解,并通过松弛迭代法收敛条件,降低计算精度,以有效减少迭代法的迭代次数,提升计算效率,这种方法称为非精确牛顿法。

$$\mathbf{F}'(\mathbf{x}_n)\mathbf{s}_n = -\mathbf{F}(\mathbf{x}_n) \quad (5)$$

传统牛拉法每次迭代时均需求解式(5)所示线性方程组的精确解 $\mathbf{s}_n$,并通过 $\mathbf{x}_{n+1}=\mathbf{x}_n+\mathbf{s}_n$ 修正迭代变量。而非精确牛顿法不需精确求解式(5),而是获取式(5)的近似解,形如:

$$\begin{cases} \mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{s}_n \\ \mathbf{F}'(\mathbf{x}_n)\mathbf{s}_n = -\mathbf{F}(\mathbf{x}_n) + \mathbf{r}_n \\ \|\mathbf{r}_n\|/\|\mathbf{F}(\mathbf{x}_n)\| \leq \eta \end{cases} \quad (6)$$

式中, $\eta \in [0,1]$ 为一给定的常数控制序列。通过选择适当的 η 值,可有效控制迭代法求解线性方程组的迭代次数,避免出现“过解”现象^[31]。

非精确牛顿法是一种双层迭代模型,由外层牛拉法迭代和内层求解修正方程组的线性迭代求解法组成。采用非精确牛顿法进行潮流计算主要有如下优点:

1) 迭代法的精度可控。在不影响牛拉法收敛性的前提下,通过降低迭代法的计算精度,有效减少迭代次数,提高计算效率。

2) 迭代法具有良好的可伸缩性。相对直接法,迭代法能大幅减少数据存储量,降低计算复杂度。

3) 迭代法易于并行计算。迭代法主要涉及稀疏矩阵与向量间的运算,数据相关性弱,具有良好的并行特性。

在采用非精确牛顿法进行潮流计算时,选择合适的迭代法求解修正方程组,可有效减少迭代次数,提高潮流计算效率。因此,如何依据特定的修正方程组选择合适的迭代法,并构造预处理器改善迭代法的收敛性,成为本文研究的重点。

2 BICGSTAB 法与预处理方法

2.1 BICGSTAB 法

如上文所述,为提高潮流计算效率,本文采用非精确牛顿法进行大规模互联电力系统潮流计算,无需精确求解修正方程组,而是采用合适的迭代求解法获取修正方程组的近似解。由于雅可比矩阵通常呈现出不对称不定特征,适合于采用 BICGSTAB 迭代法求解式(3)所示修正方程组。

设 $\Delta\mathbf{x}^{(0)}$ 是修正量的迭代初始值,采用 BICGSTAB 法求解修正方程组时,需从搜索空间 $\Delta\mathbf{x}^{(0)} + \mathbf{K}_k$ 中寻找一个近似解 $\Delta\mathbf{x}^{(k)}$,并使得残差向量 $\mathbf{r}_k = \mathbf{b} - \mathbf{A}\Delta\mathbf{x}^{(k)}$ 满足 $\mathbf{r}_k \perp \mathbf{L}_k$ 。其中, $\mathbf{K}_k(\mathbf{A}, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \dots, \mathbf{A}^{k-1}\mathbf{r}_0\}$ 为 $\mathbf{A}$ 与 $\mathbf{r}_0$ 构成的 k 维 Krylov 子空间, $\mathbf{L}_k = \mathbf{K}_k(\mathbf{A}^T, \mathbf{r}_0)$ 是另一个 k 维子空间,称为约束空间。有关 BICGSTAB 法的计算步骤及其中各变量的更新策略详见文献[32]。

最终,得到 BICGSTAB 法第 k 次迭代下,式(3)所示修正方程组的解 $\Delta\mathbf{x}^{(k)}$,其可表示为

$$\Delta\mathbf{x}^{(k)} = \Delta\mathbf{x}^{(k-1)} + \alpha_k \mathbf{p}_k + \omega_k (\mathbf{r}_{k-1} - \alpha_k \mathbf{A} \mathbf{p}_k) \quad (7)$$

上述采用 BICGSTAB 法求解修正方程组的过程主要涉及稀疏矩阵和向量间的计算^[33],在稀疏矩阵和向量间计算中,各行(列)向量及其各元素之间数据相关性弱,适合于并行处理。

2.2 预处理方法

采用 BICGSTAB 法求解修正方程组时,通常需要根据雅可比矩阵的性质构造合适的预处理器,并对雅可比矩阵进行预处理,以有效改善雅可比矩阵特征值分布,大幅提升 BICGSTAB 法的收敛性和计算效率。

雅可比矩阵预处理技术是指利用一个或多个与雅可比矩阵同维度的矩阵(称为预处理器)作用于修正方程组,压缩雅可比矩阵特征值分布区间,有效提升 BICGSTAB 法求解修正方程组的收敛性。对于一个预处理器 $\mathbf{M}$,其作用于公式(3)所示的修正方程组时,主要的预处理形式和类别如表 1 所示^[34]。

表 1 预处理的形式和类别

Table 1 Forms and categories of preconditioning		
	$\mathbf{M} \approx \mathbf{A}$	$\mathbf{M} \approx \mathbf{A}^{-1}$
左预处理	$\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b}$	$\mathbf{M}\mathbf{A}\mathbf{x} = \mathbf{M}\mathbf{b}$
右预处理	$\mathbf{A}\mathbf{M}^{-1}\mathbf{y} = \mathbf{b}, \mathbf{x} = \mathbf{M}^{-1}\mathbf{y}$	$\mathbf{A}\mathbf{M}\mathbf{y} = \mathbf{b}, \mathbf{x} = \mathbf{M}\mathbf{y}$
分裂预处理	$\mathbf{M}_1^{-1}\mathbf{A}\mathbf{M}_2^{-1}\mathbf{y} = \mathbf{M}_1^{-1}\mathbf{b}, \mathbf{x} = \mathbf{M}_2^{-1}\mathbf{y}$	$\mathbf{M}_1\mathbf{A}\mathbf{M}_2\mathbf{y} = \mathbf{M}_1\mathbf{b}, \mathbf{x} = \mathbf{M}_2\mathbf{y}$
使用方式	求解线性方程组	矩阵向量乘积

通常,衡量一个预处理方法的优劣有如下两个重要标准: 1) 预处理器构造耗时应尽量少,即预处理

理器构造时间和 BICGSTAB 法求解预处理后修正方程组的时间之和应不超过不对雅可比矩阵进行预处理直接采用 BICGSTAB 法求解原修正方程组的时间; 2) 预处理后雅可比矩阵特征值分布更为集中或条件数尽量小, 使得迭代法易于收敛。每种预处理器均有一定的适用范围, 预处理器作用于修正方程组时, 需根据其中雅可比矩阵特性进行构造。式(3)中潮流雅可比矩阵由四个分块稀疏矩阵组成, 且具有类对角占优性^[35]。因此, 本文将根据这些特性构造预处理器。

在采用 BICGSTAB 法求解修正方程组时, 由于雅可比矩阵维数较高, 特征值分布散乱, 导致 BICGSTAB 法收敛性较差。为此, 本文将采用图 1 所示的预处理方法, 以期有效改善雅可比矩阵特征值的空间分布, 提升潮流计算效率。

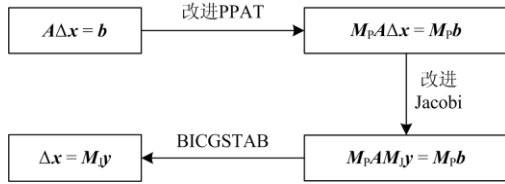


图 1 两阶段预处理 BICGSTAB 法计算流程

Fig. 1 Calculation flow of BICGSTAB method with two-step preconditioner

如图 1 所示, 本文首先提出一种改进 PPAT 预处理器 M_P , 对式(3)中雅可比矩阵 A 进行左预处理, 得到图 1 所示系数矩阵 $A_P = M_P A$, 以改善雅可比矩阵特征值分布, 但对于较大规模系统, A_P 的特征值仍较为分散; 为此, 进一步提出一种改进 Jacobi 预处理器 M_J , 再对 A_P 进行右预处理, 得到图 1 中系数矩阵 $A_{PJ} = M_P A M_J$, 大幅改善 A_P 特征值空间分布, 提升 BICGSTAB 法计算效率; 最后, 采用 BICGSTAB 迭代法实现上述两次预处理后线性方程组的并行求解, 最终得到式(3)所示修正方程组的解。为方便叙述, 本文将这种对雅可比矩阵先进行改进 PPAT 预处理, 后进行改进 Jacobi 预处理的方法称为两阶段预处理。

2.2.1 改进 PPAT 预处理器

式(3)中雅可比矩阵特征值分布散乱, 导致求解修正方程组的 BICGSTAB 法收敛性较差。为提高 BICGSTAB 法的鲁棒性与收敛性, 本文首先对雅可比矩阵进行改进 PPAT 预处理, 该预处理器基于 PPAT 预处理器改进得到。由式(3)可知, 雅可比矩阵 A 的转置 A^T 为

$$A^T = \begin{bmatrix} H^T & F^T \\ N^T & L^T \end{bmatrix} \quad (8)$$

PPAT 预处理器旨在构造与 A^T 具有相同稀疏模式的近似逆。有关 PPAT 预处理器的构造过程详见文献[36]。由于 A 某些行的非零元素较多, 导致所需存储空间较大。为减少数据存储器, 本文对 PPAT 预处理器稀疏模式进行了适当简化, 提出一种改进 PPAT 预处理器, 具体改进如下:

快速解耦法中有功和无功雅可比矩阵 B' 、 B'' 组成的分块对角矩阵 B 可作为雅可比矩阵 A 的良好预处理器, 该预处理器可表示为

$$B = \begin{bmatrix} B' & 0 \\ 0 & B'' \end{bmatrix} \quad (9)$$

式中, B 是 A 的近似 Schur 补^[37]。

借鉴上述特点, 设 M_0 是由 A^T 的子方阵 H^T 、 L^T 组成的分块对角矩阵:

$$M_0 = \begin{bmatrix} H^T & 0 \\ 0 & L^T \end{bmatrix} \quad (10)$$

式中, M_0 也为 A 的近似 Schur 补。

以 M_0 作为初始稀疏模式, 完成 PPAT 预处理器的构造, 即为本文所提改进 PPAT 预处理器 M_P 。 M_P 保持了雅可比矩阵 A 的类对角占优特性, 但其非零元素数量约为 A 的一半, 减少了数据的存储量和计算量。

改进 PPAT 预处理器 M_P 可表示为

$$M_P = \begin{bmatrix} M_{P1} & 0 \\ 0 & M_{P4} \end{bmatrix} \quad (11)$$

式(3)所示修正方程经改进 PPAT 预处理后为

$$M_P A \Delta x = M_P b \quad (12)$$

则式(12)中修正量 Δx 的系数矩阵 A_P 为

$$A_P = M_P A = \begin{bmatrix} M_{P1} H & M_{P1} N \\ M_{P4} F & M_{P4} L \end{bmatrix} = \begin{bmatrix} A_{P1} & A_{P2} \\ A_{P3} & A_{P4} \end{bmatrix} \quad (13)$$

经过上述改进 PPAT 预处理后, 求解式(3)所示修正方程组转化为求解式(12)所示与式(3)同解的线性方程组。

2.2.2 改进 Jacobi 预处理器

雅可比矩阵 A 经改进 PPAT 预处理后, 其特征值分布可得到有效改善, 但对于较大规模系统, 预处理后系数矩阵 A_P 特征值仍较为分散。为进一步改善 A_P 的特征值分布, 提升 BICGSTAB 法计算效率, 本节对 Jacobi 预处理器的稀疏模式进行了适当扩充, 提出一种改进 Jacobi 预处理器, 记为 M_J , 具体改进如下。

由式(13)可知, 系数矩阵 A_P 由四个分块矩阵组成, 虽然其子矩阵 A_{P2} 、 A_{P3} 不是方阵, 但 A_{P2} 、 A_{P3} 存在满秩子方阵 A_{P2}^* 、 A_{P3}^* 。因此, 将 A_{P1} 、 A_{P4} 、 A_{P2}^* 、

A_{p3}^* 的主对角线元素取出, 并按照其在 A_p 中的位置, 重新组成稀疏矩阵 A_J 。 A_J 可以描述为四分块矩阵, 形如:

$$A_J = \begin{bmatrix} S & T \\ U & W \end{bmatrix}, T = \begin{bmatrix} 0 \\ T^* \end{bmatrix}, U = \begin{bmatrix} 0 & U^* \end{bmatrix} \quad (14)$$

式中, S 、 T^* 、 U^* 、 W 均为对角方阵。各矩阵对角元素满足:

$$S_{ii} = A_{p1_i}, T_{ii}^* = A_{p2_i}^*, U_{ii}^* = A_{p3_i}^*, W_{ii} = A_{p4_i} \quad (15)$$

A_J 是高度稀疏的矩阵, 其子矩阵 S 与 W 为对角矩阵, 子矩阵 T 与 U 中各行和列的非零元素至多不超过 1 个。对 A_J 求逆, 即得到改进 Jacobi 预处理器 M_J 。该运算不会引入任何新的非零元素, 故不会引起数据存储量的变化。此外, M_J 的构造过程主要涉及高度稀疏矩阵的运算, 相对 Jacobi 预处理器, 改进方法增加的计算量极少。

式(12)经改进 Jacobi 预处理后为

$$A_p M_J y = M_p b \quad (16)$$

则式(16)中待求量 y 的系数矩阵 A_{PJ} 为

$$A_{PJ} = A_p M_J \quad (17)$$

此时, 求解式(16)得到的列向量 y 并非式(3)所示修正方程组的解 Δx , 但二者满足图 1 中关系:

$$\Delta x = M_J y \quad (18)$$

求解式(16)和式(18), 即可得到式(3)所示修正方程组的解, 进而依据式(4)修正迭代变量 x , 并更新式(1)所示的非线性潮流方程, 不断迭代, 直至计算结果满足精度要求, 实现电力系统潮流计算。

需指出的是: 由于牛拉法每次迭代, 雅可比矩阵均会更新, 若每次更新后重新形成上述两种预处理器, 则会增加计算成本, 影响计算效率。上述两种预处理器都是雅可比矩阵的稀疏近似逆, 潮流求解时, 若初值位于潮流解的较小邻域内, 每次更新后雅可比矩阵变化较小^[38]。因此, 在满足计算精度要求且不影响预处理性能前提下, 本文所提两种预处理器将不随雅可比矩阵更新而更新, 仅根据牛拉法首次迭代生成的雅可比矩阵形成。

3 电力系统潮流并行计算

如上文所述, 两阶段预处理 BICGSTAB 法具有良好的并行特性, 有利于修正方程组的并行求解, 本节进一步将上述算法移植到 CPU-GPU 异构平台上, 以实现潮流计算的并行化。

3.1 并行计算架构与数据存储方式

CPU-GPU 异构硬件平台由中央处理器(Central Processing Unit, CPU)和图形处理器(Graphics Processing Unit, GPU)构成, 可充分发挥 CPU 超强

的逻辑运算能力与 GPU 超强的数值并行计算能力^[39], 特别适合于处理数值计算量密集的电力系统潮流计算问题。但是, 基于 CPU-GPU 异构平台的潮流计算需借助并行计算架构才能实现 CPU 与 GPU 的协调配合, 进而实现潮流并行计算。统一计算设备架构(Compute Unified Device Architecture, CUDA)是目前较为成熟的并行计算架构, 也是本文采用的并行计算架构。

CUDA 的并程序需要 CPU 与 GPU 合作执行, CPU 将计算所需数据传到 GPU, GPU 则进行并行计算任务的执行, 再将计算所得数据传回 CPU。CPU 与 GPU 通过 PCI-Express 总线传输数据, 由于数据传输会消耗一定的时间, 影响计算效率, 故应尽量避免数据频繁交互。本文所用预处理器仅根据牛拉法首次迭代产生的雅可比矩阵生成, 而不随雅可比矩阵更新而更新, 有利于削减数据存储量与传输量, 提升潮流求解效率。

此外, 牛拉法中雅可比矩阵和本文所用预处理器均为高度稀疏矩阵, 若采用满矩阵存储方式, 将带来大量不必要的存储和计算。采用压缩存储的方式, 不但能节约存储, 还可有效降低计算量。因此, 本文采用三元组法存储潮流计算时出现的稀疏矩阵, 仅包含非零元素的数值与位置信息。

3.2 潮流并行计算步骤

本文提出的基于 CPU-GPU 异构平台的电力系统潮流并行计算方法由 CPU 部分和 GPU 部分组成, 其整体框架如图 2 所示。其中, CPU 负责牛拉法的整个流程控制, 属于本文算法外层迭代; GPU 负责修正方程组的并行迭代求解, 属于本文算法内层迭

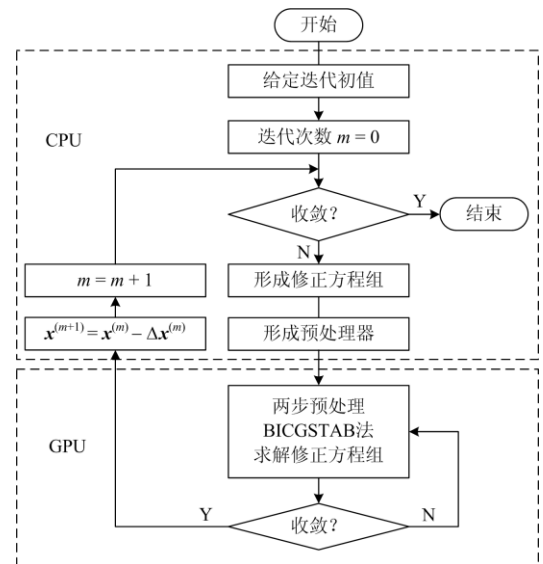


图 2 潮流算法框架

Fig. 2 Framework of power flow algorithm

代。本文算法框架下潮流并行计算流程如图 3 所示, 其实现步骤如下:

1) 读取电力系统的原始节点数据, 构建节点导纳矩阵, 得到式(1)中的 G_{ij} 、 B_{ij} , 给定节点电压幅值 V 和相角 δ 的初始值。

2) 设置迭代次数 $m=0$, 计算精度 ε_1 , 最大迭代次数 N_1 。

3) 将 V 和 δ 代入式(1), 判断 $\|f(x)\| < \varepsilon_1$ 是否成立, 若成立, 退出牛拉法潮流计算, 算法收敛; 否则, 继续步骤 4)。

4) 判断 m 是否达到设定值 N_1 , 如已达到, 退出牛拉法潮流计算, 算法不收敛; 若未达到, 令 $m=m+1$ 。

5) 根据式(1), 可得式(3)所示的修正方程组。采用两阶段预处理 BICGSTAB 法基于 GPU 进行式(3)所示修正方程组求解的具体步骤如下。

(1) 若 $m=1$, 构造 M_P 与 M_J , 继续步骤 2); 否则, 跳过此步骤, 直接执行步骤(2)。

(2) 使用 M_P 与 M_J 对 A 进行两阶段预处理, 将得到的新方程组各部分传入 GPU。

(3) 设置 y 初值 $y^{(0)}$, 允许误差 ε_2 , 求解初始残差 $r_0 = M_P b - A_{PJ} y^{(0)}$ 。

(4) 选择 p_0 使得 $(r_0, p_0) \neq 0$, 一般取 $p_0 = r_0$, 并令 $p_1 = r_0$, $k=1$ 。

(5) $\alpha_k = (r_{k-1}, p_k) / (A_2 p_k, p_k)$, $q_k = r_{k-1} - \alpha_k A_{PJ} p_k$, $\omega_k = (q_k, A_{PJ} q_k) / (A_{PJ} q_k, A_{PJ} q_k)$ 。

(6) 进而计算 $y^{(k)} = y^{(k-1)} + \alpha_k p_k + \omega_k q_k$, $r_k = q_k - \omega_k A_{PJ} q_k$ 。

(7) 若 $\|r_k\|_2 \leq \varepsilon_2$, 迭代收敛, 转至步骤(9); 否则, 继续步骤(8)。

(8) 继续计算 $\beta_k = (\alpha_k / \omega_k) g(r_k, p_k) / (r_{k-1}, p_k)$, $p_{k+1} = r_k + \beta_k (p_k - \omega_k A_{PJ} p_k)$, 令 $k=k+1$, 转至步骤(5)。

(9) 根据式(15)求得 $\Delta x^{(m)}$ 。

6) 将 GPU 计算所得 $\Delta x^{(m)}$ 传回 CPU, 根据式(4)修正迭代量 V 和 δ , 转至步骤 3)。

4 算例分析

为验证本文所提电力系统潮流并行计算方法的准确性和有效性, 本节分别以表 2 所示的不同测试系统进行验证、分析。测试所用计算平台为 Dell 塔式服务器, CPU 为 Inter Xeon E5-2620, 主频 2.10 GHz, 内存 32 GB; GPU 为 NVIDIA Tesla P100, 拥有 3 584 个单精度、1 792 个双精度核心, 单、双精度浮点性能为 5.3Teraflops、10.6Teraflops, 同时还搭配了 4096-bit 16 GB HBM2 高带宽显存, 支持 CUDA 9.0 标准。测试程序均基于 Matlab R2018a 软件和 MATPOWER 6.0 工具箱^[40]。除特殊说明外, 以下算例中牛拉法的计算精度为 10^{-5} , 求解修正方程组的 BICGSTAB 法计算精度为 10^{-5} 。

表 2 测试系统

Table 2 Test systems

系统	节点数	来源
case1354pegase	1 354	欧洲电网
case2383wp	2 383	波兰电网
case3120sp	3 120	波兰电网
case5738	5 738	两个 case2869pegas(欧洲电网) 拼接平衡节点
case9241pegase	9 241	欧洲电网
case_ACTIVSg10k	10 000	美国 WECC 电力系统
case11624	11 624	case9241pegase 和 case2383wp 拼接平衡节点
case13659pegase	13 659	欧洲电网
case_ACTIVSg25k	25 000	美国 NE/Mid-Atlantic 电力系统
case_ACTIVSg70k	70 000	美国东部互联电力系统

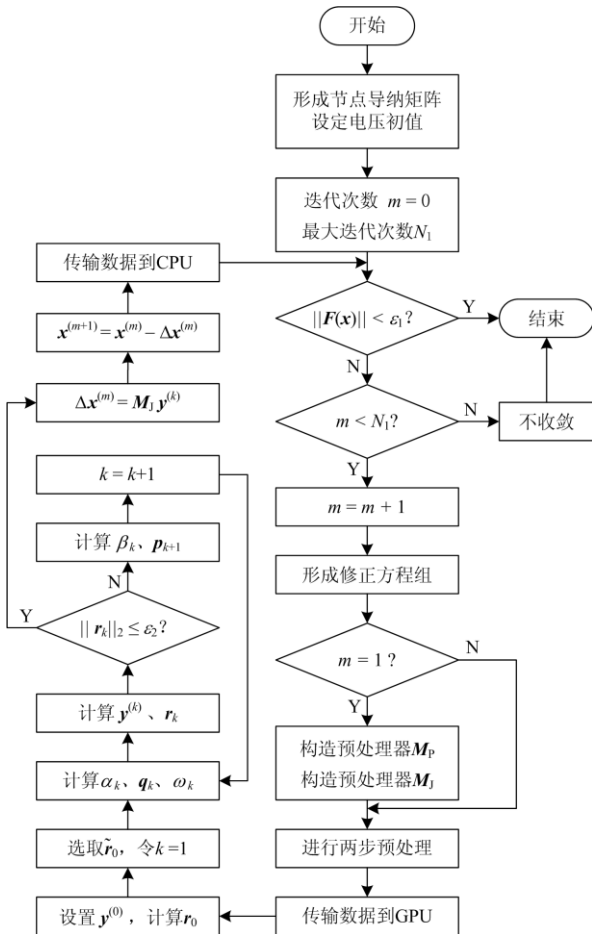


图 3 本文所提潮流算法流程图

Fig. 3 Flowchart of the proposed power flow algorithm

4.1 潮流算法准确性验证

为检验所提潮流并行计算方法的准确性，本节分别对比了传统牛拉法与本文所提算法对不同测试系统的潮流计算结果。图 4 为二者误差最大时的 PQ 节点电压幅值，表 3 详细地给出了误差最大时两种算法所得 PV 节点无功功率和节点电压相角的数值及相对误差。

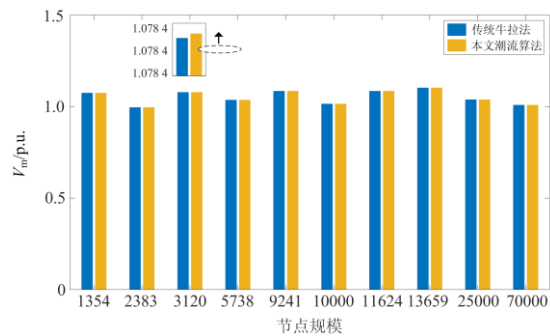


图 4 误差最大时的 PQ 节点电压幅值
Fig. 4 Voltage magnitude of PQ bus at the maximum error

显然，由图 4 可知：对于不同规模的测试系统，本文算法计算的 PQ 节点电压幅值与传统牛拉法基本一致，二者只存在极小误差。进一步，由表 3 可知：对于不同规模的测试系统，本文所提算法得到的 PV 节点无功功率和节点电压幅值与传统牛拉法近似相等，二者的最大相对误差均小于 0.064947‰。验证了所提潮流并行算法具有较高的计算精度。

4.2 预处理方法有效性验证

为检验所提改进 PPAT 预处理、改进 Jacobi 预处理与两阶段预处理的有效性，本节对比了采用不同预处理的潮流算法对不同测试系统进行潮流计算的迭代次数，结果如表 4 所示。

由表 4 的牛拉法迭代次数对比结果可知：对于同一系统，牛拉法只需迭代 3~6 次即可收敛。对比不同预处理的 BICGSTAB 法的平均迭代次数可知：改进预处理器相比于原预处理器，能有效减少 BICGSTAB 法的迭代次数，拓宽适用范围；对于不

表 3 误差最大时的无功功率与电压相角

Table 3 Reactive power and voltage phase-angle at the maximum error

系统	节点	无功功率			节点	电压相角		
		传统牛拉法/ Mvar	本文潮流算法/ Mvar	相对 误差/‰		传统牛拉法/ (°)	本文潮流算法/ (°)	相对 误差/‰
case1354pegase	7267	265.953 360	265.953 330	0.001 128	5131	-26.204 302	-26.204 297	0.001 908
case2383wp	2115	-2.925 625	-2.925 608	0.058 107	1568	-5.038 276	-5.038 278	0.007 095
case3120sp	121	61.894 119	61.894 123	0.000 646	861	-11.052 040	-11.052 038	0.001 793
case5738	1788	2.750 257	2.750 257	0.000 021	268	27.680 242	27.680 242	0.000 014
case9241pegase	8340	-15.243 123	-15.243 222	0.064 947	1386	62.269 014	62.269 014	0.000 022
case_ACTIVSg10k	13561	-20.465 436	-20.465 435	0.000 489	10080	-45.476 958	-45.476 958	0.000 110
case11624	8340	15.243 123	15.243 122	0.000 656	4691	-33.661 086	-33.661 086	0.000 033
case13659pegase	6103	198.639 102	198.639 105	0.000 027	803	-16.595 547	-16.595 547	0.000 015
case_ACTIVSg25k	55556	2.148 525	2.148 525	0.000 013	55543	-14.535 550	-14.535 551	0.000 112
case_ACTIVSg70k	27039	81.227 015	81.227 029	0.001 724	28146	-67.928 744	-67.928 740	0.000 589

表 4 使用不同预处理器时潮流算法的迭代次数

Table 4 Iteration times of power flow algorithm with different preconditioners

系统	PPAT 预处理	改进 PPAT 预处理	Jacobi 预处理	改进 Jacobi 预处理	两阶段预处理
case1354pegase	不收敛	3/162	3/655	3/328	3/84
case2383wp	5/1384	5/271	不收敛	5/910	5/133
case3120sp	不收敛	5/497	5/6479	5/1264	5/198
case5738	不收敛	6/915	不收敛	6/590	6/216
case9241pegase	不收敛	6/1176	不收敛	不收敛	6/472
case_ACTIVSg10k	不收敛	4/559	不收敛	不收敛	4/361
case11624	不收敛	6/1280	不收敛	不收敛	6/465
case13659pegase	不收敛	不收敛	不收敛	不收敛	5/838
case_ACTIVSg25k	不收敛	不收敛	不收敛	不收敛	4/3701
case_ACTIVSg70k	不收敛	不收敛	不收敛	不收敛	5/3501

注：“/”前后分别为外层牛拉法迭代次数与内层预处理 BICGSTAB 法求解修正方程组的平均迭代次数

同规模的测试系统,两阶段预处理 BICGSTAB 法的迭代次数均最少,且对于单独采用改进 PPAT 预处理或改进 Jacobi 预处理时不收敛的大规模测试系统,采用两阶段预处理仍可收敛。由此可见:两阶段预处理潮流算法所需迭代次数少,鲁棒性强,为大规模电力系统潮流的快速求解提供了可行方案。

4.3 预处理后矩阵特征值

为观察改进 PPAT 预处理、改进 Jacobi 预处理与两阶段预处理对雅可比矩阵特征值的影响,图 5 以 case1354pegase 系统为例,对比牛拉法首次迭代产生的雅可比矩阵在不同预处理后的特征值分布。

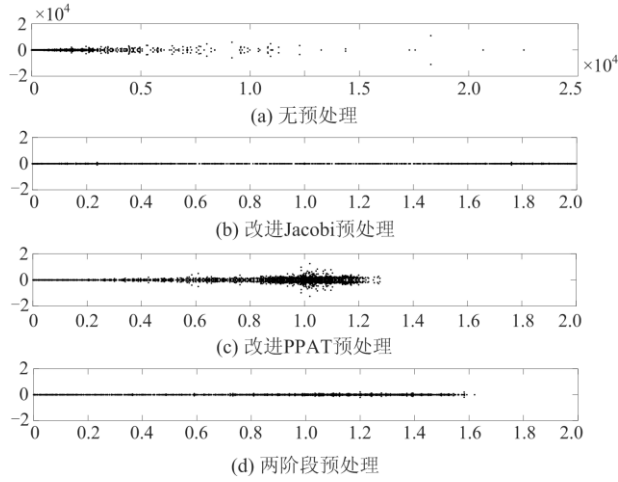


图 5 不同预处理后雅可比矩阵的特征值分布

Fig. 5 Eigenvalue spectrum of Jacobian matrix after different preconditioning

由图 5 中不同预处理后雅可比矩阵特征值分布结果可知:所提两阶段预处理相对改进 PPAT 预处理与改进 Jacobi 预处理,可使雅可比矩阵特征值分别在实轴方向和虚轴方向上分布更为密集。该结果表明:本文所提两阶段预处理可有效改善雅可比矩阵特征值空间分布,使其更为密集,有利于提高 BICGSTAB 法计算效率。

4.4 潮流算法的收敛性分析

为分析潮流算法内层迭代中两阶段预处理 BICGSTAB 法的收敛性,图 6 以 case3120sp 测试系统为例,对比了所提潮流算法内层迭代中两阶段预处理 BICGSTAB 法在不同计算精度下求解修正方程组的迭代次数。

由图 6 中 BICGSTAB 法的平均迭代次数变化曲线可知:两阶段预处理 BICGSTAB 法求解修正方程组的平均迭代次数随着计算精度要求的提高而增

加,二者基本呈线性关系。同时, BICGSTAB 迭代法首次迭代的迭代次数也呈现相似的变化趋势,且

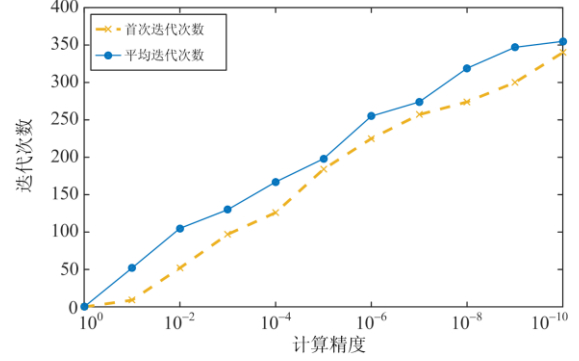


图 6 两阶段预处理 BICGSTAB 法收敛性分析

Fig. 6 Convergence analysis of BICGSTAB with two-step preconditioner

两条收敛曲线均较为光滑,有利于进行收敛性判断。

图 7 进一步以 case3120sp 测试系统为例,对比了在不同 BICGSTAB 法计算精度要求下,采用不同预处理的潮流算法的计算耗时。由图 7 计算结果可知:采用两阶段预处理的潮流算法相比采用单一预处理的潮流算法耗时更少,且精度要求越高效果越明显;当计算精度提高到 10^{-12} 时,采用改进 Jacobi 预处理的潮流算法会出现不收敛的情况,而采用两阶段预处理的潮流算法依然可以收敛。该结果表明:对于计算精度要求较高的问题,采用两阶段预处理的潮流算法具有更快的计算速度和更强的鲁棒性。

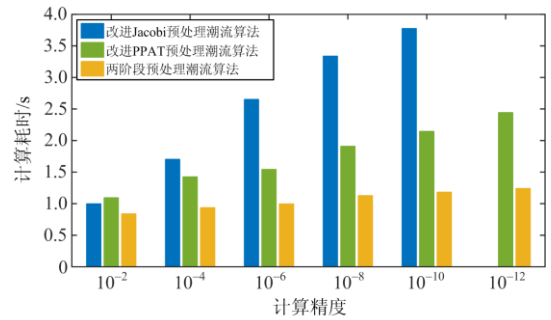


图 7 不同计算精度要求下不同预处理潮流算法计算耗时

Fig. 7 Computational time of power flow algorithm with different preconditioner under different calculation accuracy

4.5 计算耗时对比

为评估本文所提潮流算法计算效率,图 8 对比了本文基于 CPU-GPU 异构平台的两阶段预处理潮流算法与传统牛拉法的计算耗时。

由图 8 可知:系统规模较小时,传统牛拉法的

计算耗时较少；随着系统规模的扩大，传统牛拉法的计算耗时急剧增加，而本文基于 CPU-GPU 异构平台的两阶段预处理潮流算法在计算耗时上的优势逐渐显现，且系统规模越大，优势越显著。

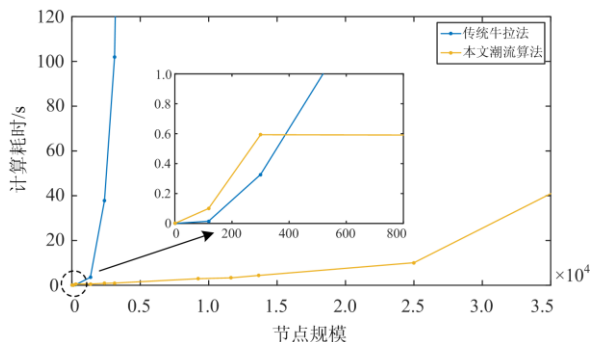


图 8 不同潮流算法的计算时间

Fig. 8 Computational time of different power flow algorithms

为分析 CPU-GPU 异构平台加速情况，以 CPU 平台上两阶段预处理潮流算法计算耗时为基准值，统计其与 CPU-GPU 异构平台上两阶段预处理潮流算法计算耗时的比值(记为加速比)，分析 CPU-GPU 异构平台对潮流算法的加速情况，结果如表 5 所示。

表 5 潮流计算时间与 CPU-GPU 异构平台加速比

Table 5 Power flow computing time and speedup ratio of CPU-GPU heterogeneous platform

系统	CPU 平台 计算耗时/s	CPU-GPU 异构平 台计算耗时/s	加速比
case1354pegase	0.584	0.589	0.99
case2383wp	0.839	0.852	0.98
case3120sp	1.186	0.999	1.19
case5738	2.595	1.406	1.85
case9241pegase	7.913	2.952	2.68
case_ACTIVSg10k	4.330	2.207	1.96
case11624	9.356	3.329	2.81
case13659pegase	15.773	4.409	3.58
case_ACTIVSg25k	71.668	10.078	7.11
case_ACTIVSg70k	379.816	148.962	2.55

由表 5 可知：对于规模较小系统，基于 CPU-GPU 异构平台的并行潮流计算耗时比基于 CPU 平台的串行潮流计算耗时更久，主要原因是 GPU 并行加速时间难以弥补 CPU 与 GPU 间的数据交互时间；随着系统规模的增大，基于 CPU-GPU 平台的并行潮流计算优势逐渐显现，与基于 CPU 平台的串行潮流计算相比，最高加速比可达 7.11 倍。由此可见：本文基于 CPU-GPU 异构平台的并行潮流算法在大规模互联电力系统潮流计算中具有较高的效率，能够很好地实现大规模互联电力系统快速

潮流计算。

5 结论

本文提出一种非精确牛顿法与两阶段预处理 BICGSTAB 法相结合的大规模区域互联电力系统潮流计算方法，并基于 CPU-GPU 异构平台实现电力系统潮流的并行计算。通过不同测试系统对所提方法进行验证、分析，相关结论如下：

1) 所提潮流并行计算方法可实现电力系统潮流的准确求解，与传统牛拉法潮流计算结果相比，二者相对误差极小；

2) 所提改进 PPAT 预处理与改进 Jacobi 预处理相结合的雅可比矩阵两阶段预处理方法，可有效改善雅可比矩阵特征值分布，改善 BICGSTAB 法的收敛性和鲁棒性；

3) 所提潮流并行计算方法可实现大规模电力系统的快速潮流计算，与传统牛拉法相比，计算效率提升显著；

4) 所提基于 CPU-GPU 异构平台的并行潮流算法相对基于 CPU 平台的串行潮流算法，对大规模互联电力系统潮流计算效率提升显著。

大规模电力系统结构复杂，节点连接方式灵活多样，再加上新能源的大规模并网和直流输电技术的广泛应用，对潮流计算效率产生了较大影响，在今后的研究中，将结合电力系统的此类特点，从电力系统本身出发，研究更为高效的潮流算法。同时，还可尝试将潮流算法应用到电力系统的其他计算问题，如 $N-1$ 安全校核、连续潮流和稳定性计算等，以提高此类计算效率。

参考文献

- [1] 陈德扬, 李亚楼, 江涵, 等. 基于道路树分层的大电网潮流并行算法及其 GPU 优化实现[J]. 电力系统自动化, 2014, 38(22): 63-69.
CHEN Deyang, LI Yalou, JIANG Han, et al. A parallel power flow algorithm for large-scale grid based on stratified path trees and its implementation on GPU[J]. Automation of Electric Power Systems, 2014, 38(22): 63-69.
- [2] 吴素农, 于金镒, 田庄, 等. 基于扩展节点法的交直流混合电网统一潮流算法[J]. 电力系统保护与控制, 2018, 12(14): 46-53.
WU Sunong, YU Jinyi, TIAN Zhuang, et al. A unified power flow algorithm for AC/DC hybrid networks using modified augmented nodal analysis[J]. Power System Protection and Control, 2018, 12(14): 46-53.
- [3] 韩轩, 邱晓燕, 沙熠, 等. 基于变置信度机会约束规划的风电并网优化调度[J]. 高压电器, 2018, 54(10):

- 179-186, 193.
- HAN Xuan, QIU Xiaoyan, SHA Yi, et al. Optimal dispatching under grid-connection of wind farm based on variable confidence chance-constrain model[J]. High Voltage Apparatus, 2018, 54(10): 179-186,193.
- [4] 刘克文, 张国芳, 袁琛, 等. 基于图计算的快速非线性迭代法求解潮流计算[J]. 电力信息与通信技术, 2018, 16(10): 19-24.
- LIU Kewen, ZHANG Guofang, YUAN Chen, et al. Fast nonlinear iterative method based on graph calculation for power flow calculation[J]. Electric Power Information and Communication Technology, 2018, 16(10): 19-24.
- [5] 胡博, 周家启, 刘洋, 等. 基于预条件处理 GMRES 的不精确牛顿法潮流计算[J]. 电工技术学报, 2007, 22(2): 98-104.
- HU Bo, ZHOU Jiaqi, LIU Yang, et al. Inexact Newton flow computation based on preconditioned GMRES method[J]. Transactions of China Electrotechnical Society, 2007, 22(2): 98-104.
- [6] 李婷, 胥威汀, 刘向龙, 等. 含高比例可再生能源的交直流混联电网规划技术研究综述[J]. 电力系统保护与控制, 2019, 47(12): 177-187.
- LI Ting, XU Weiting, LIU Xianglong, et al. Review on planning technology of AC/DC hybrid system with high proportion of renewable energy[J]. Power System Protection and Control, 2019, 47(12): 177-187.
- [7] 黄景光, 陈波, 林湘宁, 等. 基于乌鸦搜索算法的孤岛微网多目标优化调度[J]. 高压电器, 2020, 56(1): 162-168.
- HUANG Jingguang, CHEN Bo, LIN Xiangning, et al. Multi-objective optimal operation of islanded microgrid based on crow search algorithm[J]. High Voltage Apparatus, 2020, 56(1): 162-168.
- [8] 赵文清, 祝玲玉, 高树国, 等. 基于多源信息融合的电力变压器故障诊断方法研究[J]. 电力信息与通信技术, 2018, 16(10): 25-30.
- ZHAO Wenqing, ZHU Lingyu, GAO Shuguo, et al. Research on fault diagnosis method of power transformer based on multi-source information fusion[J]. Electric Power Information and Communication Technology, 2018, 16(10): 25-30.
- [9] ZHU Jiahui, CHEN Panpan, GU Chenghong, et al. Techno-economic analysis of MJ class high temperature superconducting magnetic energy storage (SMES) systems applied to renewable power grids[J]. Global Energy Interconnection, 2018(2): 172-178.
- [10] 唐坤杰, 董树锋, 宋永华. 基于不完全 LU 分解预处理迭代法的电力系统潮流算法[J]. 中国电机工程学报, 2017, 37(1): 55-62.
- TANG Kunjie, DONG Shufeng, SONG Yonghua. Power flow algorithm based on an iterative method with incomplete LU decomposition preconditioning[J]. Proceedings of the CSEE, 2017, 37(1): 55-62.
- [11] 夏沛, 汪芳宗. 大规模电力系统快速潮流计算方法研究[J]. 电力系统保护与控制, 2012, 40(9): 38-42.
- XIA Pei, WANG Fangzong. Study on fast power flow calculation methods of large-scale power systems[J]. Power System Protection and Control, 2012, 40(9): 38-42.
- [12] 张永杰, 孙秦. 大型稀疏线性方程组符号 LU 分解法[J]. 计算机工程与应用, 2007, 43(28): 29-30.
- ZHANG Yongjie, SUN Qin. Symbol LU decomposition method of large scale sparse linear equations[J]. Computer Engineering & Applications, 2007, 43(28): 29-32.
- [13] KULKARNI A Y, PAI M A, SAUER P W. Iterative solver techniques in fast dynamic calculations of power systems[J]. International Journal of Electrical Power & Energy Systems, 2001, 23(3): 237-244.
- [14] 刘洋, 周家启, 谢开贵, 等. 基于 Beowulf 集群的大规模电力系统方程并行 PCG 求解[J]. 电工技术学报, 2006, 21(3): 105-111.
- LIU Yang, ZHOU Jiaqi, XIE Kaigui, et al. Parallel PCG solution of large scale power system equations based on Beowulf cluster[J]. Transactions of China Electrotechnical Society, 2006, 21(3): 105-111.
- [15] 刘洋, 周家启, 谢开贵, 等. 预条件处理 CG 法大规模电力系统潮流计算[J]. 中国电机工程学报, 2006, 26(7): 89-94.
- LIU Yang, ZHOU Jiaqi, XIE Kaigui, et al. The preconditioned CG method for large scale power flow solution[J]. Proceedings of the CSEE, 2006, 26(7): 89-94.
- [16] VORSTY H A. Iterative solution of linear systems in the 20th century[J]. Journal of Computational and Applied Mathematics, 2001, 123(1): 1-33.
- [17] LI X, LI F. GPU-based two-step preconditioning for conjugate gradient method in power flow[C] // 2015 IEEE Power & Energy Society General Meeting, July 26-30, 2015, Denver, USA: 9-13.
- [18] 刘宽, 王淳, 伍惠铖, 等. 一种含 PV 节点的配电网线性潮流计算方法[J]. 电力系统保护与控制, 2019, 47(3): 17-22.
- LIU Kuan, WANG Chun, WU Huicheng, et al. A linear power flow calculation method for distribution networks with PV nodes[J]. Power System Protection and Control, 2019, 47(3): 17-22.
- [19] LI X, LI F, CLARK J M. Exploration of multifrontal method with GPU in power flow computation[C] // 2013 IEEE Power & Energy Society General Meeting, July 21-25, 2013, Vancouver, Canada: 1-5.
- [20] YANG Y, SONG A, LIU H, et al. Parallel computing of multi-contingency optimal power flow with transient

- stability constraints[J]. Protection and Control of Modern Power Systems, 2018, 3(2): 204-213. DOI: 10.1186/s41601-018-0095-z.
- [21] ZHOU G, BO R, CHIEN L, et al. GPU-accelerated algorithm for on-line probabilistic power flow[C] // 2018 IEEE Power & Energy Society General Meeting (PESGM), August 5-10, 2018, Portland, OR, USA.
- [22] 张逸飞, 严正, 赵文恺, 等. 基于 GPU 的分块约化算法在小干扰稳定分析中的应用[J]. 电力系统自动化, 2015, 25(22): 90-97.
ZHANG Yifei, YAN Zheng, ZHAO Wenkai, et al. Application of GPU-based block reduction algorithm in power system small-signal stability analysis[J]. Automation of Electric Power Systems, 2015, 25(22): 90-97.
- [23] YU Z, HUANG S, SHI L, et al. GPU-based JFNG method for power system transient dynamic simulation[C] // International Conference on Power System Technology, October 1-5, 2014, Chengdu, China: 8-13.
- [24] SAAD Y, SCHULTZ M H. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems[J]. SIAM Journal on Scientific and Statistical Computing, 1986, 7(3): 856-869.
- [25] VORST H A V D. Bi-CGSTAB: a fast and smoothly converging variant of bi-CG for the solution of nonsymmetric linear systems[J]. SIAM Journal on Scientific and Statistical Computing, 1992, 13(2): 631-644.
- [26] BENZI M. Preconditioning techniques for large linear systems: a survey[J]. Journal of Computational Physics, 2002, 182(2): 418-477.
- [27] DEMBO R S, EISENSTAT S C, STEIHAUG T. Inexact Newton methods[J]. Siam Journal on Numerical Analysis, 1982, 19(2): 400-408.
- [28] 王锡凡. 现代电力系统分析[M]. 北京: 科学出版社, 2003.
- [29] 唐坤杰, 董树锋, 宋永华. 一种 GPU-CPU 异构运算框架加速的实时 N-1 交流潮流计算方法[J]. 中国电机工程学报, 2018, 38(15): 36-43.
TANG Kunjie, DONG Shufeng, SONG Yonghua. A real-time N-1 AC power flow calculation method based on CPU-GPU heterogeneous computing framework[J]. Proceedings of the CSEE, 2018, 38(15): 36-43.
- [30] CHEN Y, SHEN C. A Jacobian-free Newton-GMRES (m) method with adaptive preconditioner and its application for power flow calculations[J]. IEEE Transactions on Power Systems, 2006, 21(3): 1096-1103.
- [31] 蔡大用, 陈玉荣. 用不完全 LU 分解预处理的不精确潮流计算方法[J]. 电力系统自动化, 2002, 26(8): 11-14.
CAI Dayong, CHEN Yurong. Solving power flow equations with inexact Newton methods preconditioned by incomplete LU factorization with partially fill-in[J]. Automation of Electric Power Systems, 2002, 26(8): 11-14.
- [32] 谷同祥, 安恒斌, 刘兴平. 迭代方法和预处理技术[M]. 北京: 科学出版社, 2015.
- [33] 张健飞, 沈德飞. 基于 GPU 的稀疏线性系统的预条件共轭梯度法[J]. 计算机应用, 2013, 33(3): 825-829.
ZHANG Jianfei, SHEN Defei. GPU-based preconditioned conjugate gradient method for solving sparse linear systems[J]. Journal of Computer Applications, 2013, 33(3): 825-829.
- [34] FLUECK A J, CHIANG H D. Solving the nonlinear power flow equations with an inexact Newton method using GMRES[J]. IEEE Transactions on Power Systems, 2002, 13(2): 267-273.
- [35] 唐灿, 董树锋, 任雪桂, 等. 用于迭代法潮流计算的改进 Jacobi 预处理方法[J]. 电力系统自动化, 2018, 34(12): 87-92.
TANG Can, DONG Shufeng, REN Xuegui, et al. Improved Jacobi pre-treatment method for solving iterative power flow calculation[J]. Automation of Electric Power Systems, 2018, 34(12): 87-92.
- [36] 迟利华, 刘杰, 李晓梅. 稀疏近似逆预条件子及其并行计算[J]. 计算机学报, 2000, 23(3): 255-260.
CHI Lihua, LIU Jie, LI Xiaomei. Parallel sparse approximate inverse preconditioners[J]. Chinese Journal of Computers, 2000, 23(3): 255-260.
- [37] IDEMA R, LAHAYE D J P, VUIK C, et al. Scalable Newton-Krylov solver for very large power flow problems[J]. IEEE Transactions on Power Systems, 2010, 27(1): 390-396.
- [38] 刘烨. 电力系统负荷裕度的并行计算方法研究[D]. 吉林: 东北电力大学, 2019.
LIU Ye. Parallel calculation of load margin in bulk power grid using CPU-GPU architecture[D]. Jilin: Northeast Electric Power University, 2019.
- [39] LI X, LI F. GPU-based power flow analysis with Chebyshev preconditioner and conjugate gradient method[J]. Electric Power Systems Research, 2014, 116(11): 87-93.
- [40] ZIMMERMAN R D, MURILLO-SANCHEZ C E, THOMAS R J. MATPOWER: steady-state operations, planning, and analysis tools for power systems research and education[J]. IEEE Transactions on Power Systems, 2011, 26(1): 12-19.

收稿日期: 2019-12-04; 修回日期: 2020-02-25

作者简介:

宋晓喆(1986—), 男, 博士, 高级工程师, 研究方向为电力系统稳定与控制; E-mail: songxzjl@163.com

魏 国(1995—), 男, 硕士研究生, 研究方向为电力系统高性能计算; E-mail: guowei201@aliyun.com

王长江(1989—), 男, 博士研究生, 主要研究方向为柔性直流输电的控制和安全性稳定性分析、电力系统暂态稳定。

E-mail: cjwangneepu@163.com

(编辑 姜新丽)