

第一性原理软件北京原子技术模拟工具包的重构进展

马英晋 张坦 何莲花 金钟

Progress of refactoring first principle package of Beijing Simulation Tool for Atom TEchnology

Ma Ying-Jin Zhang Tan He Lian-Hua Jin Zhong

引用信息 Citation: *Acta Physica Sinica*, 69, 043101 (2020) DOI: 10.7498/aps.69.20191658

在线阅读 View online: <https://doi.org/10.7498/aps.69.20191658>

当期内容 View table of contents: <http://wulixb.iphy.ac.cn>

您可能感兴趣的其他文章

Articles you may be interested in

Al原子在Si表面扩散动力学的第一性原理研究

First-principles study on the diffusion dynamics of Al atoms on Si surface

物理学报. 2019, 68(20): 207302 <https://doi.org/10.7498/aps.68.20190783>

石墨烯/BiOI纳米复合物电子结构和光学性质的第一性原理模拟计算

First-principle calculation on electronic structures and optical properties of hybrid graphene and BiOI nanosheets

物理学报. 2018, 67(11): 116301 <https://doi.org/10.7498/aps.67.20172220>

基于第一性原理的含空位 α -Fe和H原子相互作用研究

Investigation of interaction between α -Fe metal and H atom by ab-initio method

物理学报. 2020, 69(5): 053101 <https://doi.org/10.7498/aps.69.20191775>

基于第一性原理的新型非线性光学晶体探索

First principle study of nonlinear optical crystals

物理学报. 2018, 67(11): 114203 <https://doi.org/10.7498/aps.67.20180189>

锑烯吸附金属Li原子的密度泛函研究

Density functional study of metal lithium atom adsorption on antimonene

物理学报. 2019, 68(2): 026802 <https://doi.org/10.7498/aps.68.20181648>

第一性原理软件北京原子技术 模拟工具包的重构进展*

马英晋¹⁾²⁾ 张坦³⁾⁴⁾ 何莲花^{1)2)†} 金钟¹⁾²⁾

1) (中国科学院计算机网络信息中心, 北京 100190)

2) (中国科学院计算科学应用研究中心, 北京 100190)

3) (北京凝聚态物理国家研究中心, 中国科学院物理研究所, 北京 100190)

4) (中国科学院大学物理科学学院, 北京 100049)

(2019年10月29日收到; 2019年12月3日收到修改稿)

基于第一性原理的理论方法的研究, 代表了材料计算、分子模拟等领域的科学高地, 相应的第一性原理计算软件直接关系到该领域相关理论、算法的积累. 本文汇报了我们在重构第一性原理计算模拟软件——北京原子技术模拟工具包(BSTATE)的一些最新进展. 重构的核心思想是降低用户使用门槛、扩展软件适用范围、增加软件对于流行计算框架的支持. 基于此思路, 在BSTATE原有Makefile编译系统的基础上添加了CMake编译环境, 并支持各种数学函数库的自动和交互式配置; 通过在原有内置泛函基础上增添Libxc泛函库的支持, 使BSTATE支持的泛函数量有了数量级上的增长; 分析测试BSTATE在集群的并行特点, 并以更新数学库接口(FFTW3、Cufftw)的形式提供对于流行异构框架的初步支持.

关键词: 第一性原理软件, 密度泛函理论, 北京原子技术模拟工具包, 软件重构

PACS: 31.15.E, 31.15.eg, 31.15.-p

DOI: 10.7498/aps.69.20191658

1 引言

电子结构计算的理论依据是量子力学理论. 量子力学建立了微观粒子运动变化的基本规律, 将人类对自然的认识扩展到原子尺度的微观世界. 非相对论量子力学最流行的表述形式是Schrödinger的波动力学形式, 它的核心是波函数的概念及其运动方程Schrödinger方程. 但电子体系的Schrödinger方程是高维奇异问题, 很难直接求解. 于是人们又通过各种手段对Schrödinger方程进行等价转换或近似, 其中最基本的近似是绝热近似或称Born-Oppenheimer近似. 它的基本思想是将整个问题分

成两部分来考虑, 考虑电子运动时原子核是处在它们的瞬时位置上, 而考虑核的运动时则不考虑电子在空间的具体分布. 目前实际计算中用于求解Schrödinger方程的各种方法大都是在这个近似基础上进行的. 根据是否使用经验参数和实验数据, 求解Schrödinger方程的近似方法主要分为3类: 第一性原理方法、半经验方法和经验方法. 第一性原理方法是指不使用任何经验参数和实验数据, 只利用基本物理常数和原子量计算Schrödinger方程的方法, 求解后得到体系的一系列性质, 如晶体结构、结合能和电子结构等.

众所周知, 化学和材料科学是现代科学技术和国民经济发展的重要基石, 基于第一性原理的分

* 国家重点研发计划(批准号: 2017YFB0202202)、中国科学院信息化专项(批准号: XXH13506-403)和国家自然科学基金(批准号: 21703260)资助的课题.

† 通信作者. E-mail: helh@sccas.cn

子、材料模拟是研究化学反应和材料物性最重要的方法之一, 为目标功能导向的材料设计提供了可能. 第一性原理计算方法在材料的性质研究和预测方面发挥着越来越大的作用, 已经成为继实验和理论分析之外的最为重要的研究方法和手段之一. 尤其是对于新型的拓扑材料的预测和研究, 第一性原理方法发挥了无可替代的作用.

当前我国在材料计算、反应模拟领域仍然缺乏核心的竞争力和影响力, 一个重要的原因就是成熟的第一性原理系列计算软件的缺失, 仍处于利用国外成熟商业软件, 如: GAUSSIAN^[1], VASP^[2] (Vienna Ab-initio Simulation Package, 维也纳从头算模拟程序包), 进行研究工作. 这也导致相关基础人才储备不足、后劲缺乏, 影响到计算研究进一步的发展. 虽然目前国内的许多研究小组已经在该领域做出了出色的工作, 也有越来越多的相关程序包发布 (如 BDF^[3], ABACUS^[4], BSTATE^[5] 等), 但是目前有关这方面的研究积累仍然不足, 尤其是国产的适用于大型计算集群、支持异构框架的大规模并行软件仍然缺失. 这极大限制了可计算模拟的材料尺度或者分子大小, 进而制约了相关材料和化学科学的发展.

BSTATE (Beijing Simulation Tool for Atom Technology), 中文名为北京原子技术模拟工具包, 是基于密度泛函理论和平面波赝势 (包括模守恒赝势^[6]和超软赝势^[7,8]) 方法的第一性原理计算程序包, 主体程序在 20 世纪 90 年代由中国科学院物理所方忠研究员及其合作者共同开发. BSTATE 主要用于周期性晶体结构下能量、电子密度、能带结构等的计算模拟, 进而得到物质的电、磁、光等其他特性. 除了可以很好地处理一般的基态电子结构计算, 该程序包的长处在于能比较好的处理各种磁有序、轨道序、强自旋轨道耦合体系, 包含了独创的处理强关联体系的 LDA+Gutzwiller^[9-11] 方法, 大幅提高了计算效率^[5]. 基于 BSTATE 程序包, 以物理所凝聚态研究组为主体的研究人员针对 Bi₂Te₃, Bi₂Se₃ 和 Sb₂Te₃ 等拓扑材料进行了第一性原理的计算, 其研究表明这些体系是首次发现的三维拓扑绝缘体, 并且在其二维磁性掺杂薄膜中能实现量子化的反常霍尔效应. 这也为极低能量耗散的自旋电子器件设计指出了一个新的发展方向^[12-15].

虽然 BSTATE 程序包已经在拓扑绝缘体的计

算中取得了很好的成绩, 但是 BSTATE 的使用主要局限在少数课题组内, 有必要进行进一步的推广. 当前 BSTATE 局限性的主要在于用户门槛较高; 泛函数量较为单一, 不好满足特定材料的计算要求; 以及暂时没有考虑流行的硬件框架. 针对上述问题, 我们拟对中科院物理所的第一性原理材料计算软件包 BSTATE 进行重构, 以期提高软件的易用性和适用范围、降低用户门槛、优化其对流行计算框架的支持.

2 BSTATE 程序的基本原理

BSTATE 程序为大型的第一性原理程序包, 有近 30 万行的 Fortran 代码. 程序使用了 FFTW (the Faster Fourier Transform in the West) 函数库、Intel MKL (Math Kernel Library) 等多种数学函数和 MPI (Message Passing Interface) 来提高计算效率. 程序的核心部分是对交换关联势做了 LDA^[16] (Local Density Approximation) 或者 GGA^[17,18] (Generalized Gradient Approximations) 近似的 KS (Kohn-Sham) 方程的自治求解^[19].

2.1 Schrödinger 方程与 DFT 理论

波恩-奥本海默近似下, 体系中电子的定态可以由满足非含时 Schrödinger 方程的 N 电子波函数 $\Psi(\mathbf{r}_1, \dots, \mathbf{r}_N)$ 来描述, 即

$$\begin{aligned} H\Psi &= [T + V + U]\Psi \\ &= \left[\sum_i^N -\frac{\hbar^2}{2m} \nabla_i^2 + \sum_i^N V(\mathbf{r}_i) + \sum_{i<j}^N U(\mathbf{r}_i, \mathbf{r}_j) \right] \Psi \\ &= E\Psi. \end{aligned} \quad (1)$$

其中, H 为体系哈密顿算符, 包含动能 T (拉普拉斯算符 ∇)、势能 V 、电子相互作用 U 三部分. 由于电子之间的相互作用会让体系成为很难处理的多体问题, 无法精确求解. DFT 可以将含 U 的多体问题转化为不含 U 的单体问题, 从而成为解决此类问题的一个有效方法. 密度泛函理论中, 关键的变量为粒子的密度, 即

$$\begin{aligned} \rho(\mathbf{r}) &= N \int d^3 r_2 \int d^3 r \dots \int d^3 r_N \\ &\quad \cdot \Psi^*(\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_N) \Psi(\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_N), \end{aligned} \quad (2)$$

(2) 式的关系也可以反过来, 即给出电子的密度 $\rho(\mathbf{r})$, 原则上也可以计算出对应的波函数 Ψ . 基于

此, Hohenberg 与 Kohn(HK 定理^[20]) 指出存在电子密度 $\rho(\mathbf{r})$ 的总能量泛函 $E(\rho)$, 其中 $E(\rho)$ 可以写成外势 V_{ext} 的形式, 即

$$E[\rho] = \int \rho(\mathbf{r})V_{\text{ext}}d\mathbf{r} + J[\rho] + G[\rho], \quad (3)$$

其中, V_{ext} 由 $\rho(\mathbf{r})$ 唯一确定, $J[\rho]$ 为经典的库伦排斥能. 此时可以通过最小化 $E(\rho)$, 来获得体系的能量、电子密度以及其他所有性质. 相比于计算量随粒子数指数次增长的波函数方法, 电子密度方法的计算量与电子数目无关, 仅与在三维空间 (x, y, z) 内找到电子的概率有关. 由于泛函 $G(\rho)$ 是未知的, HK 定理并没有给出计算 $E(\rho)$ 的实际方法.

2.2 KS-DFT 与泛函近似

KS 理论下 $G[\rho]$ 可以表示为 $T_s[\rho] + E_{\text{xc}}[\rho]$ 的形式^[21,22], 其中 $T_s[\rho]$ 是假想无相互作用体系的动能泛函, $E_{\text{xc}}[\rho]$ 是交换关联能量. $E_{\text{xc}}[\rho]$ 的求解需要交换关联势 ($\nu_{\text{xc}}(r) = \delta E_{\text{xc}}[\rho]/\delta\rho(r)$) 的近似形式, 其最简单的近似就是局域密度近似, 即 LDA. LDA 近似下 ρ 用均匀电子气模型来处理. LDA 交换关联能 $E_{\text{xc}}^{\text{LDA}}$ 只与 ρ 有关, 表示为

$$E_{\text{xc}}^{\text{LDA}}[\rho] = \int \varepsilon_{\text{xc}}(\rho(\mathbf{r}))\rho(\mathbf{r})d\mathbf{r}, \quad (4)$$

其中, $\varepsilon_{\text{xc}}(\rho(\mathbf{r}))$ 是均匀原子气单粒子的交换关联能. 在 LDA 中, 具有相反自旋的电子占据相同的 KS 空间轨道.

为了校正电子密度分布不均匀带来的误差, 将表示电子密度不均匀性的密度梯度 ($\nabla\rho$) 作为半局域变量, 引入到 E_{xc} , 此为含广义密度梯度校正的近似即 GGA, 得到的泛函即为 GGA 泛函. GGA 交换关联能 $E_{\text{xc}}^{\text{GGA}}$ 是 ρ 和 $\nabla\rho$ 的泛函, 即 $E_{\text{xc}}^{\text{GGA}}[\rho, \nabla\rho]$; 考虑到电子自旋后 GGA 泛函需要由四个参数确定, 即 $E_{\text{xc}}^{\text{GGA}}[\rho_{\uparrow}, \rho_{\downarrow}, \nabla\rho_{\uparrow}, \nabla\rho_{\downarrow}]$; 也因此相比于 LDA 泛函, GGA 泛函在计算结合能方面明显提高了计算精度.

除了 LDA 和 GGA 泛函, meta-GGA 和 hybrid-GGA^[23,24] 以及双杂化泛函都是重要的泛函形式. 当前 BSTATE 暂时没有用到后三种形式的泛函. 泛函确定后, 体系的波函数需要通过求解 $H\varphi = E\varphi$ 形式的 KS 方程来得到, 具体形式为

$$\left(-\frac{\hbar^2}{2m}\nabla^2 + \nu_{\text{eff}}(r)\right)\varphi_i(r_i) = \varepsilon_i\varphi_i(r_i), \quad (5)$$

其中 ν_{eff} 为泛函确定后的有效势.

2.3 BSTATE 程序框架

作为第一性原理计算程序包, BSTATE 程序包的核心部分是对交换关联势做了 LDA 或者 GGA 近似的 KS 方程的自洽求解. 其基本流程示意图如图 1 所示.

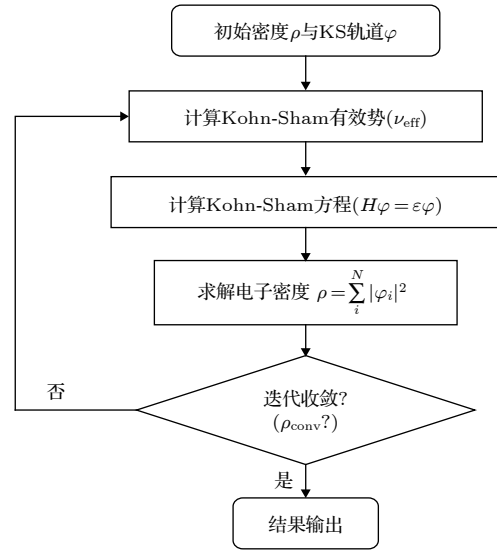


图 1 BSTATE 计算流程示意图
Fig. 1. Flowchart of BSTATE package.

以上数值求解的过程中必须把连续量离散化, 如系统实空间原胞划分成足够细的网格, k 空间的布里渊区的 k 点离散化. 在迭代求解 KS 方程的过程中需要初始化电荷密度 ρ 与 KS 轨道, 可以利用单原子的电子密度分布作为初始值. 有 HK 定理保证, KS 势可以表示成电荷密度的泛函; 所以很容易得到离散网格上的 KS 势, 进而可以求解固定 KS 势场的单电子本征方程获得 KS 轨道信息. 一般情况, 求解出来的 KS 轨道与初始 KS 轨道是不相同的. 轨道求解出来以后可以根据 DFT 波函数为单行列式的性质求得电子密度信息 ρ . 获得新的电子密度后, 使用 HK 定理重新计算 KS 势, 重新求解 KS 方程获得 KS 轨道信息, 最后重新利用 KS 轨道和电子排布获得电子密度信息. 更新重复上面的过程, 直到所求出的新的电荷密度与输入的电荷密度在判据范围内一致, 即可认为收敛. 这时候体系的波函数信息包括 KS 轨道和电子占据, 在此基础上可以得到体系的总能量、电子态密度、能带图、布洛赫波、费米面、光电导等一系列结果.

3 BSTATE 程序分析与重构进展报告

3.1 编译系统

重构前的 BSTATE 仅能采用 GNU make 进行程序构建和管理. Makefile 文件描述了整个工程的编译、连接等规则. 使用 Makefile 时, BSTATE 的编译步骤是:

1) 通过预编译指令 (GNU 的 `cpp` 或者 `intel fpp`), 生成编译时使用的 Fortran 代码. 值得注意的是, BSTATE 的 `mlwf` (Maximally Localised Wannier Functions) 部分需要使用自由格式的预编译; 其余需要使用固定格式的预编译. 此外, 预编译后实际生成的代码会受条件编译的控制.

2) 预编译后的源代码 (`.f90`, `.for`) 被统一为 `.f` 后缀的 Fortran 代码. BSTATE 编译的过程中首先会编译整个程序所需要的模块函数 (module), 之后继续编译和连接所有的程序文件.

针对不同的计算机构架和编译环境, BSTATE 有着不同的 Makefile 模板. 并支持 Intel MKL 和 FFTW2 等一系列的函数库. 针对不同的编译环境, 用户需要重新修改所有的编译选项. 这对于非计算机专业的用户存在不低的使用门槛, 同时使得跨平台和交叉编译复杂化.

重构后的 BSTATE 在保留旧版 Makefile 编译流程的基础上, 提供全新的 CMake 编译系统支持^[25]. CMake 全称是“cross platform make”, 是一个开源的跨平台自动化构建系统. 新版的 BSTATE 使用指定名为 `CMakeLists.txt` 的配置文件来控制软件的预编译、编译、连接等流程. 与此同时, CMake 天生跨平台的特征, 使得仅需要对于 `CMakeList.txt` 进行简单配置, 原则上即可以生成不同目标平台 (如 Linux, Windows, Mac) 的程序文件. 这可以大大简化跨平台和交叉编译方面的工作. 重构后的 BSTATE 编译系统的特点主要有:

1) 保留原版 Makefile 的编译流程. 使用 `cmake` 或者 `ccmake` 时可以指定预编译环境 (`cpp`, `fpp`). 预编译时可以指定条件编译的选项, 生成编译 BSTATE 时使用的有效代码.

2) 内置了不同优化等级的预编译和编译方案 (如 `debug`, `release` 等); 支持多线程编译.

3) 提供 `Libxc`, `FFTW2/3`, `CUFFT/CUFFTW`

等外接函数库开关选项; 支持不同语言的混合编译.

4) 通过 `cmake-curses-gui` 提供图形交互式 (GUI) 的预编译和编译支持.

5) 通过对“`MATH_ROOT`”或类似关键环境变量的查找, 提供对于常见数学函数库的自动搜索和支持功能. 该功能默认开启, 并按照“`MKL`, `ESSL`, `OPENBLAS`, `ATLAS`, `ACML`, `SYSTEM_NATIVE`”的默认顺序进行数学函数库的自动配置.

通过以上的调整, 重构后的 BSTATE 可以通过 CMake 实现自动和快速编译. 比如终端中键入 `FC = mpif90 ccmake [BSTATE_CMakeList.txt]`, 即可开启预编译和编译的 GUI, 进行自动配置或按提示修改编译选项.

在表 1 中我们也总结了 BSTATE 新老编译系统的对比. 可以看出重构后的编译系统相对于旧版, 无论是在代码维护、扩展性、用户使用等方面均有长足的进步.

表 1 重构前后 BSTATE 编译系统的对比
Table 1. Comparison of BSTATE compilation system.

项目	重构前	重构后
编译系统	GNUMake	CMake
图形GUI	不支持	支持
跨平台	手工修改Makefile文件提供支持	原生支持
数学库	手工配置	自动配置
外置函数库	手工配置	支持自动配置
异构支持	否	是
高级编译选项	手工配置	支持GUI配置
多线程编译	不支持	支持
用户门槛	高	低

3.2 接口升级、函数库整合

重构前的 BSTATE 程序已经可以调用多种函数库来实现额外的功能或加速计算过程, 例如傅里叶变换 FFTW 函数库, 但其仅支持老旧的 FFTW2 接口. 因此, 首先在 BSTATE 里面增加 FFTW3^[26] 的调用接口, 以便使用常用的 SSE2 和 AVX 指令集, 加速运算. 同时, 也初步接入了基于 CUDA 的傅里叶变换函数库 (`CUFFTW` 和 `CUFFT`), 已可取得同 CPU 版本一致的计算结果, 但是效率仍有很大的提升空间.

重构前的 BSTATE 已经支持 LDA, GGA,

LDA+U, GGA+U 等计算. 但其仅通过固定的代码支持 `ldapw91` 和 `ggapbe` 两个泛函. 因此将 `Libxc` 库^[27] 接入 `BSTATE`, 使其可以支持流行的各种泛函. `Libxc` 是当前最为全面的交换关联泛函库. 该泛函库由 `ETSF`(European Theoretical Spectroscopy Facility) 维护, 并几乎收录了主流期刊公开报道的所有交换关联泛函.

`Libxc` 内部函数相互依赖, 调用需要编译为库文件后才能在子程序中调用. 因此重构时主要通过预编译的方式在多个子程序中配置, 然后再进行调用. 在重构的 `BSTATE` 程序中, `LDA` 和 `GGA` 泛函通过新添加的 `extern_functionals.f90` 接口程序接入. 对于 `LDA` 泛函, 势能仅仅依赖于电子的密度 (即 $E_{xc}^{LDA} = E_{xc}^{LDA}[\rho(\mathbf{r})]$); 对于 `GGA` 泛函, 势能会依赖于电子的密度和电子密度的梯度 (即 $E_{xc}^{GGA} = E_{xc}^{GGA}[\rho(\mathbf{r}), \nabla\rho(\mathbf{r})]$). 接口程序定位在 `BSTATE` 中的 `xc_gga.f90` 源程序中. 除了电子密度、梯度等信

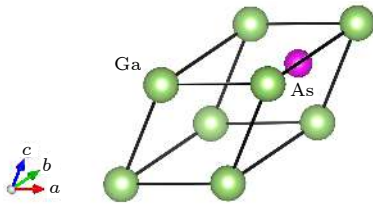


图 2 闪锌矿结构的 GaAs 双原子化合物
Fig. 2. GaAs compound with zinc blende structure.

息, k 点信息、自旋等也要在接口中传递.

作为测试, 使用砷化镓体系 (GaAs, 见图 2) 和石墨烯片段作为测试的算例. 测试的结果表明重构后的 `BSTATE` 在 `DFT` 计算中性能可以与之前持平 (表 2 和表 3), 无论是单核、多核还是 `MPI` 并行计算, 性能均没有明显差距. 此外, 受益于 `FFTW3` 对于 `SSE2` 和 `AVX` 系列指令集的支持, 在 `Intel` 异构众核平台 (`MIC`) 的计算结果会有一些的提升.

表 2 `Libxc` 版本与原始版本性能比较
Table 2. Benchmarks between `BSTATEs` with/without `Libxc`.

项目	单核/s	多核/s
<code>BSTATE</code>	42.4	22.6
<code>BSTATE+Libxc</code>	43.6	23.4
性能比	0.97	0.97

* 测试机器为 `AMD A10 PRO-7800 B R7` (4核); `GaAs` 体系

表 3 `FFTW3+Libxc` 版本与原始 `FFTW2` 版本性能比较
Table 3. Benchmarks between `V.FFTW2` and `V.FFTW3+ Libxc`.

项目	<code>CPU</code> 平台/s	<code>MIC</code> 平台/s
<code>FFTW2</code>	1181	1717
<code>FFTW3+Libxc</code>	1179	1593
性能比	1.00	1.08

* `CPU`平台为 `Intel 至强 E7-4830v3` (56核); 石墨烯体系
* `MIC`平台为 `Intel Phi-7210` (64核); 石墨烯体系

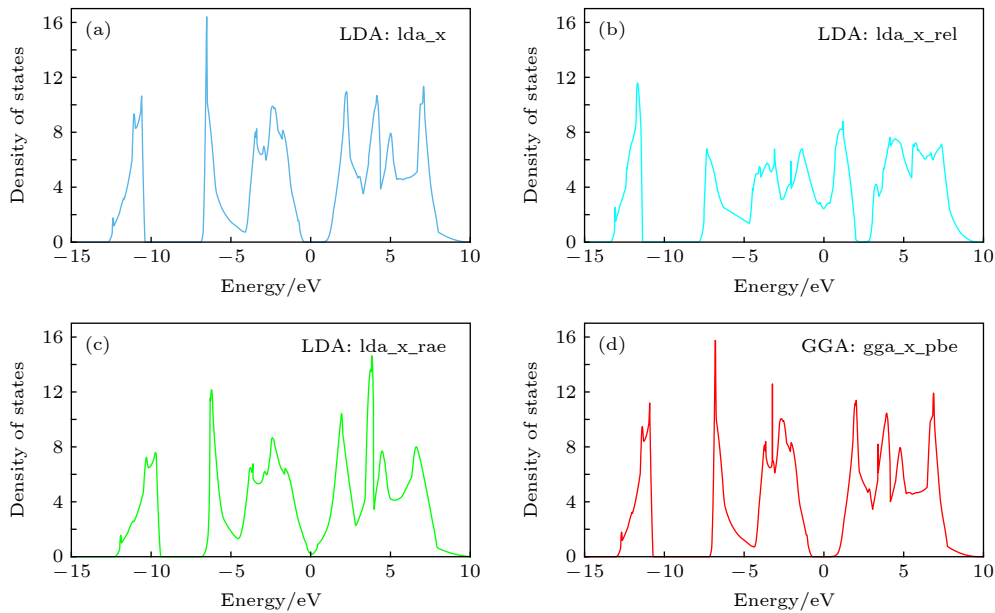


图 3 多种泛函计算的 GaAs 的态密度 (a) (b) (c) (d)
Fig. 3. Calculated density of states (DOS): (a); (b); (c); (d).

使用整合 Libxc 和 FFTW3 的 BSTATE 版本, 我们在原有的 lpw91(LDA) 和 ggapbe(GGA) 泛函的基础上使用相对论 (lda_x_rel^[28]) 和自适应 (lda_x_rae^[29]) 的 LDA 泛函计算了 GaAs 的态密度. 其计算结果列于图 3, 可以看出对于该体系不同的泛函会得到整体相似的态密度结果, 主要区别在 -6 eV、0—4 eV 能量区间.

3.3 并行分析和测试

提高第一性原理计算模拟软件的计算效率一直是计算科学领域的一大挑战. 其主要原因是电子的强相互作用导致能量和波函数只能在哈密顿构造完之后才可以求解; 而且构造和求解的过程中也涉及各种数据结构或者数值代数方面的问题. BSTATE 软件在设计之初就已经很好地考虑这些方面的问题, 并已经可以通过 MPI 模块按照 k 点进行分布式求解. 所谓 k 点并行化, 即将 k 点均匀分配给 (并行环境中的) 多个 CPU, CPU 之间的数据交换较少, 计算效率高.

为了详细分析 BSTATE 的并行性能, 我们选用 GaAs 晶体和石墨烯片段分别进行小规模和大规模的测试计算. 对于较小体系如 GaAs 晶体, 使用个人 PC 在多核 (4 线程) 计算, 分析过程中的各个大类模块所占时间比例统计于图 4 中 (编译采用-O2 优化). 可以看出, 傅里叶变换时间占比接

近 58%, 紧随其后为特征值求解, 时间占比约为 12% 左右, 态密度、球贝塞尔函数计算部分合计占比均为 15% 左右. 对于较大体系如石墨烯片段的

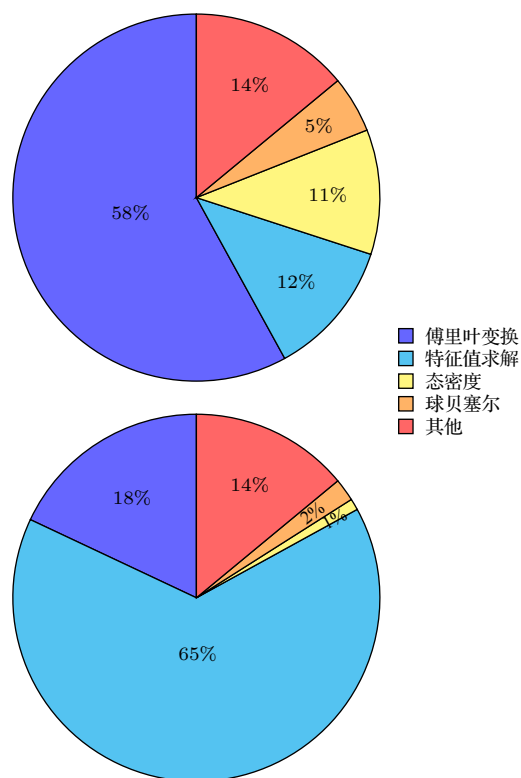


图 4 个人 PC (左图) 及集群 MPI (右图) 测试中 BSTATE 各模块的时间占比统计

Fig. 4. Time percentages of subroutines in BSTATE using PC (left) and Cluster (right).

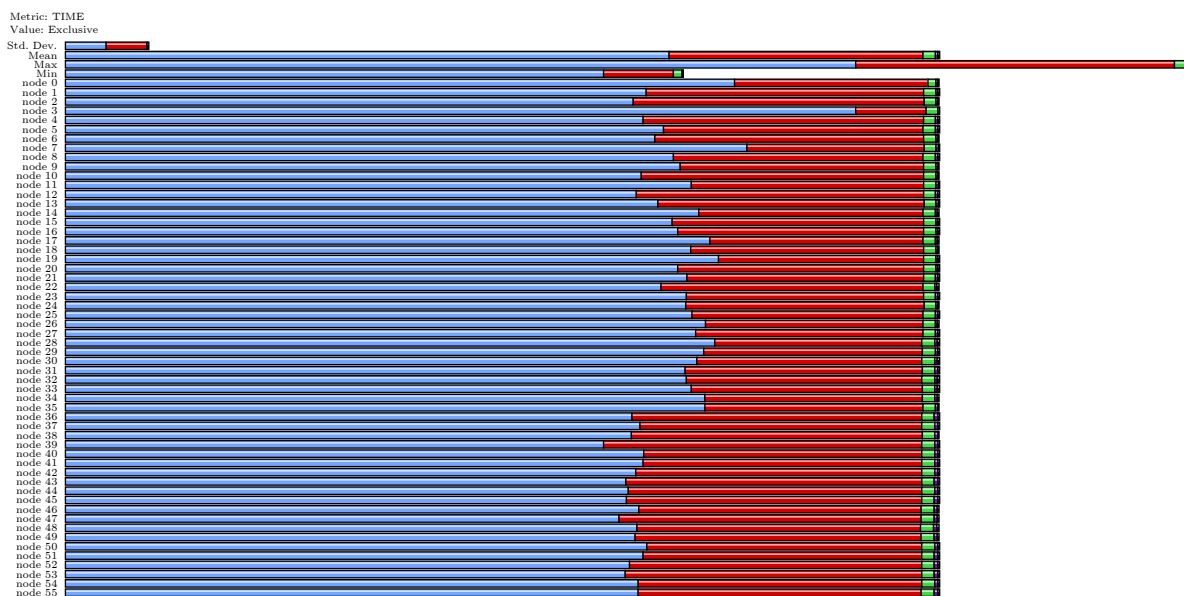


图 5 BSTATE 的 MPI 多线程并行分析; 其中蓝色、红色、黄色、黑色分别代表 application(BSTATE), MPI_barrier, MPI_reduce 以及 MPI_allreduce.

Fig. 5. Analysis via TAU performance system framework for BSTATE.

计算(图4),我们使用56核MPI并行的集群进行了测试,测试结果表明特征值求解部分占用最多的时间比例,约为65%,同时傅里叶变换时间占比下降到18%左右.

在图4中集群MPI的并行模式下面,我们也通过使用TAU(Tuning and Analysis Utilities; 重构的BSTATE通过CMake可以整合TAU模块)分析了BSTATE并行时各个线程的执行情况,并示于图5中.可以看出在按照现有的基于 k 点的并行策略,BSTATE各个线程有着较好的负载均衡,各个计算核心的计算能力都得到很好的发挥.

最后,我们使用重构的BSTATE版本(即包含CMake+TAU+FFTW3+Libxc的版本),在中国科学院的“元”集群上对石墨烯片段体系进行了更大规模的测试,以验证新的版本可以稳定的执行计算任务.该测试平台基于曙光TC4600 E刀片平台,采用Intel Xeon E5 2680 V3处理器(2.5 GHz、12核).测试的结果示于图6中.可以看出,重构的BSTATE相对于重构前基于FFTW2的版本在跨节点并行方面有着一定的性能提升(0—17%左右的加速).这应该得益于新的FFTW3调用接口以及基于Libxc的泛函求解方案.同时我们也注意到当前基于 k 点的并行求解方案在核心过多的情况下效率下降很快,这也是实际计算中特别要注意的方面.

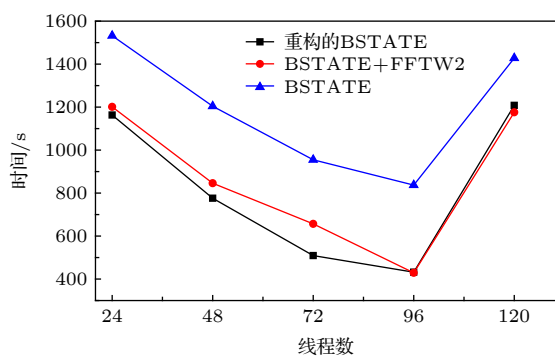


图6 重构前后的BSTATE跨节点并行测试

Fig. 6. Benchmark of MPI parallel for BSTATE with and without refactoring.

4 结论

本文汇报了我们在重构第一性计算模拟软件BSTATE的一些最新进展.根据重构的核心思想(降低用户使用门槛、扩展软件适用范围、提高软件计算效率),我们的工作主要集中在编译、配置部分

的自动化,增加泛函类别,以及优化程序接口3个方面.重构后的BSTATE通过CMake的引入可以实现跨平台、简单图形化的编译;通过Libxc和FFTW3的引入和更新实现了程序功能的扩展和对更多计算平台的支持;最后的并行分析和稳定性测试,验证了重构后的BSTATE可以更好地胜任第一性原理方面的计算任务,并在跨节点并行方面有着一定的性能提升.

在重构的进程中,也注意到BSTATE仍有较大的改进空间.比如在原理层面上的哈密顿的构造、子空间迭代等自洽迭代改进方案,以及技术层面上的CPU-GPU协同计算方面.这些需要更多的第一性原理算法上的积累和深入,也是我们后面需要进一步考虑的方面.

感谢徐顺博士关于程序优化方面的指导和讨论.

参考文献

- [1] Frisch M J, Trucks G W, Schlegel H B, et al. 2016 *Gaussian Inc. Wallingford CT*
- [2] Kresse G, Furthmüller J 1996 *Comp. Mater. Sci.* **6** 15
- [3] Liu W, Wang F, Li L 2003 *J. Theor. Comput. Chem.* **2** 257
- [4] Li P, Liu X, Chen M, Lin P, Ren X, Lin L, Yang C, He L 2016 *Comput. Mater. Sci.* **112** 503
- [5] Fang Z, Terakura K J 2002 *Phys. Condens. Mat.* **14** 3001
- [6] Troullier N, Martins J L 1991 *Phys. Rev. B* **43** 1993
- [7] Vanderbilt D 1990 *Phys. Rev. B* **41** 7892
- [8] Kresse G, Hafner J J 1994 *Phys.: Condens. Matter* **6** 8245
- [9] Deng X Y, Wang L, Dai X, Fang Z 2009 *Phys. Rev. B* **79** 075114
- [10] Rullier-Albenque F, Alloul H, Balakirev F, Proust C 2008 *Europhys. Lett.* **81** 37008
- [11] Rice T M, Ueda K 1985 *Phys. Rev. Lett.* **55** 995
- [12] Yu R, Zhang W, Zhang H J, Zhang S C, Dai X, Fang Z 2010 *Science* **329** 61
- [13] Yu R, Zhang W, Weng H M, Dai X, Fang Z 2010 *Physics* **39** 618 (in Chinese) [余睿, 张薇, 翁红明, 戴希, 方忠 2010 物理 **39** 618]
- [14] Liang Y C, Zhang Y, Guo W L, Yao Y G, Fang Z 2007 *Physics* **36** 385 (in Chinese) [梁拥成, 张英, 郭万林, 姚裕贵, 方忠 2007 物理 **36** 385]
- [15] Xu G, Dai X, Fang Z 2009 *Physics* **38** 651 (in Chinese) [徐刚, 戴希, 方忠 2009 物理 **38** 651]
- [16] Dirac P A M 1930 *Proc. Camb. Phil. Soc.* **26** 376
- [17] Perdew J P, Chevary J A, Vosko S H, Jackson K A, Pederson M R, Singh D J, Fiolhais C 1992 *Phys. Rev. B* **46** 6671
- [18] Perdew J P, Burke K, Ernzerhof M 1997 *Phys. Rev. Lett.* **77** 3865
- [19] Kohn W, Sham L J 1965 *Phys. Rev.* **140** A1133
- [20] Hohenberg P, Kohn W 1964 *Phys. Rev.* **136** B864
- [21] Huang M C 2000 *Progress in Physics* **20** 199 (in Chinese) [黄美纯 2000 物理学进展 **20** 199]
- [22] Li Z Y, H W, Yang J L 2005 *Progress In Chemistry* **17** 192 (in Chinese) [李震宇, 贺伟, 杨金龙 2005 化学进展 **17** 192]
- [23] Heyd J, Scuseria G E 2003 *J. Chem. Phys.* **118** 8207

- [24] Heyd J, Scuseria G E 2006 *J. Chem. Phys.* **124** 219906
 [25] Clemencic M, Mato P 2012 *J. Phys.: Conf. Ser.* **396** 052021
 [26] Frigo M, Johnson S G 2005 *Proceedings of the IEEE* **93** 216
 [27] Lehtola S, Steigemann C, Oliveira M J T, Marques M A L 2018 *Software X* **7** 1
 [28] Engel E, Keller S, Bonetti A F, Müller H, Dreizler R M 1995 *Phys. Rev. A.* **52** 2750
 [29] Rae A 1973 *Chem. Phys. Lett.* **18** 574

Progress of refactoring first principle package of Beijing Simulation Tool for Atom TEchnology*

Ma Ying-Jin¹⁾²⁾ Zhang Tan³⁾⁴⁾ He Lian-Hua^{1)2)†} Jin Zhong¹⁾²⁾

1) (*Computer Network Information Center, Chinese Academy of Science, Beijing 100190, China*)

2) (*Center of Scientific Computing Applications & Research, Chinese Academy of Science, Beijing 100190, China*)

3) (*Beijing National Research Center for Condensed Matter Physics, Institute of Physics, Chinese Academy of Science, Beijing 100190, China*)

4) (*School of Physical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China*)

(Received 29 October 2019; revised manuscript received 3 December 2019)

Abstract

The development of first principle methods can represent the summit of the sciences in the material computing and molecular modeling, and the corresponding first principle software packages are closely related with the accumulation of theories and algorithms in this field. In this paper, we reported our recent progress in refactoring the first principle package BSTATE. The key points in the reconstruction are lowering the doorsill, extending the scope of application, as well as adjusting package to the popular computer hardware. And as such, we updated the Makefile system to the new CMake system, in which the GUI can be used and many math libraries can be configured automatically; we added the support for the Libxc library, in which a large quantity of density functionals are included; we updated the interface for supporting GPU, in order to support the heterogeneous computing system. After refactoring, the Makefile system of BSTATE can supply both the Makefile and CMake system, and the Fourier transform libraries such as FFTW2, FFTW3, and Cufftw, the math libraries such as Intel MKL library, Openblas, and the density functional library such as Libxc, can be automatically or manually assigned. The integration of FFTW3 can slightly prompt the calculating efficiency in Intel's many integrated core (MIC) architecture, and the integration of Cufftw can supply the initial support for the graphics processing unit (GPU) architecture, respectively. The usage of Libxc library makes the BSTATE package has the capacity to use hundreds density functionals, and the usages of various functionals were demonstrated by calculating the density of states of GaAs compound. Beyond the integration of various libraries, the parallel performance of BSTATE was also investigated. It can be found that the Fourier transformation and the solving for the eigenvalue equations are the major contributions. Using the tuning and analysis utilities (TAU) tool, we found that the tasks can be well distributed in modern HPC clusters. It implied that the refactoring didn't affect the parallel efficiency of original BSTATE package. In a following benchmark test of graphene fragments, one can found that the refactored BSTATE package showed the best performance, its FFTW3 & Libxc version owns about 0–17% acceleration comparing to that of FFTW2 version.

Keywords: first principle computational software, density-functional theory, Beijing Simulation Tool for Atom TEchnology, Software refactoring

PACS: 31.15.E, 31.15.eg, 31.15.-p

DOI: 10.7498/aps.69.20191658

* Project supported by the National Key Research and Development Project of China (Grant No. 2017YFB0202202), the Informationization Program of the Chinese Academy of Science, China (Grant No. XXH13506-403), and the National Science Foundation of China (Grant No. 21703260).

† Corresponding author. E-mail: helh@sccas.cn