TBPW: A Modular Framework for Pedagogical Electronic Structure Codes

Todd D. Beaudet, Dyutiman Das, Nichols A. Romero, William D. Mattson, Jeongnim Kim and Richard M. Martin

Materials Computation Center

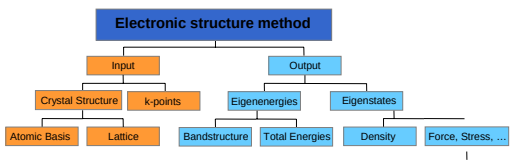
Motivation

All electronic structure codes have much in common, including a basic paradigm and set of components. The purpose of the code is to provide a framework to demonstrate the similarities, a pedagogical example of the development of such codes, and a pedagogical tool for teaching and understanding electronic structure.

Framework

- Provides common tools for crystal structures, k-point sampling, input/output, and graphics, along with examples of different bases and methods for diagonalization of the kamiltonian.
- Provides basic atomic simulation data structures.

Common Components



Electronic Structure Capabilities

TBPW is written from the ground-up in a modular style using Fortran 90. This code is composed of three distinct parts: common tools, tight binding (TB) and plane wave (PW) methods.

TB – Tight-Binding

Implemented using a rotation matrix formalism allows the use of orbitals with arbitrary angular momentum.

PW – Plane-Wave

Implemented using empirical pseudopotentials (non-self-consistent). Options of diagonalization via direct-inversion, or conjugate gradient method with optional fast Fourier transform (FFT).

Diagonalization Methods

Two alternative ways of diagonalization are provided:

1. Standard LAPACK routine $O(N_{basis}^3)$
   For TB or PW
2. Conjugate Gradient using FFT for evaluating $H\Psi$ $O(N_{electron} N_{basis} \ln N_{basis})$.
   For PW only

Conjugate Gradient scheme

- Constrained minimization of a quadratic functional , $\langle\Psi|H|\Psi\rangle$, with $|\Psi\rangle$ finally converging to the lowest eigenvector allowed by the constraints.
- The constraints reduce efficiency, but in practice sufficient accuracy is obtained.

Why FFT?

The Hamiltonian is constructed in the plane-wave basis, where a potential is not diagonal but the kinetic energy is. $H|\Psi\rangle = T|\Psi\rangle + V|\Psi\rangle$, or $[H\Psi]_G = T_G\Psi_G + \Sigma_{G'}V(G-G')\Psi_{G'}$

- The first term is a scalar multiplication, the second becomes a scalar multiplication in the real space by the convolution theorem

$$\Sigma_{G'}V(G-G')\Psi_{G'} = FFT^{-1}[V_r\Psi_r]$$

Since $N_B << N$, FFT is inexpensive compared to a full $O(N^3)$ diagonalization.

Relation to other codes

This is a simple example of the algorithm as used in ABINIT – similar to the one used in VASP

Tight Binding Method

Slater-Koster scheme implemented in a rotation matrix form that permits use of real (or complex) spherical harmonic basis with arbitrary angular momenta.

- Hamiltonian matrix elements are given by

$$H_{i\alpha j\beta}(\vec{k}) = \sum_{\vec{R}_j} \exp\left(i\vec{k}\cdot(\vec{R}_j - \vec{R}_i)\right)\left[T^*(\phi_{ij},\theta_{ij})K(R_{ij})T(\phi_{ij},\theta_{ij})\right]_{\alpha\beta}$$

where $T(\phi_{ij},\theta_{ij})$ is the rotation operator

$$T(\phi_{ij},\theta_{ij}) = \exp(-i\theta_{ij}L_y)\exp(-i\phi_{ij}L_z)$$

K is the Slater-Koster matrix in the standard configuration.

- Easy input for widely used parameters such as the Harrison "universal parameters"

Relation to other codes

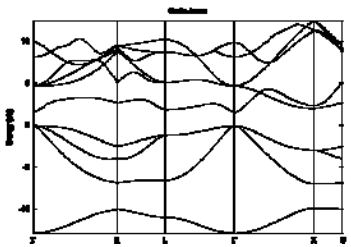
A simple example of local orbitals – as used in SIESTA

Input/Output

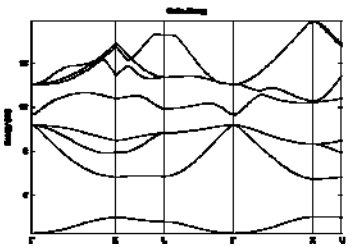
Input: Keyword format with structure and keywords similar to **SIESTA** – also close to **ABINIT**

Output: Standard data files or convenient plots using **Gnuplot** – density on a grid – can use **Xcrysden** and other plotting packages

Sample Bands: Gallium Arsenide – TB - PW

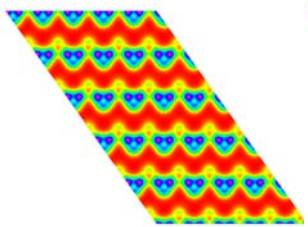


TB parameters from
J.-M. Jancu et al, *Phys. Rev. B* **57** 6493 (1998)

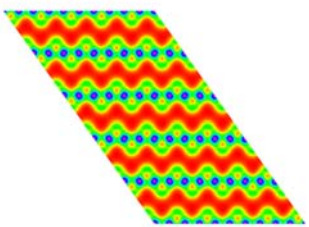


PW empirical pseudopotential from
Zhang, et al., *Phys. Rev. B* **48** 11204 (1993).

Sample Densities: PW – GaAs – Si



GaAs



Si

Plotted using Xcrysden. See A.Kokalj, Comp. Mater. Sci., 2003, Vol. 28, p. 155. Code available from http://www.xcrysden.org/

Conclusions

- A useful code for Teaching – Used in Summer schools
- A useful code for many calculations – semiconductor bands, nanotubes, . . .
- Electronic structure codes fit well into the component based code paradigm
- Fortran 90 is sufficient for component based programming, but requires careful planning.