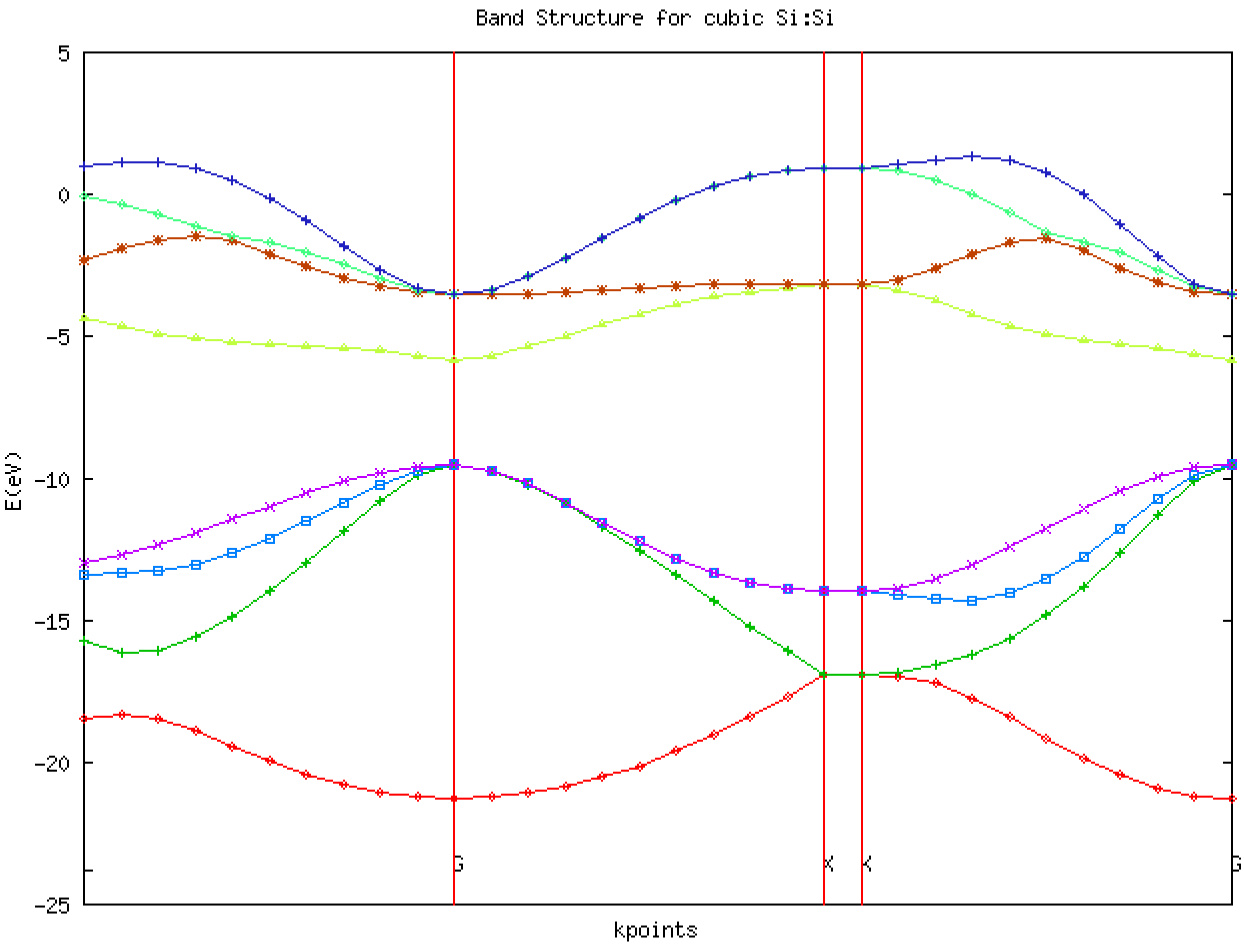
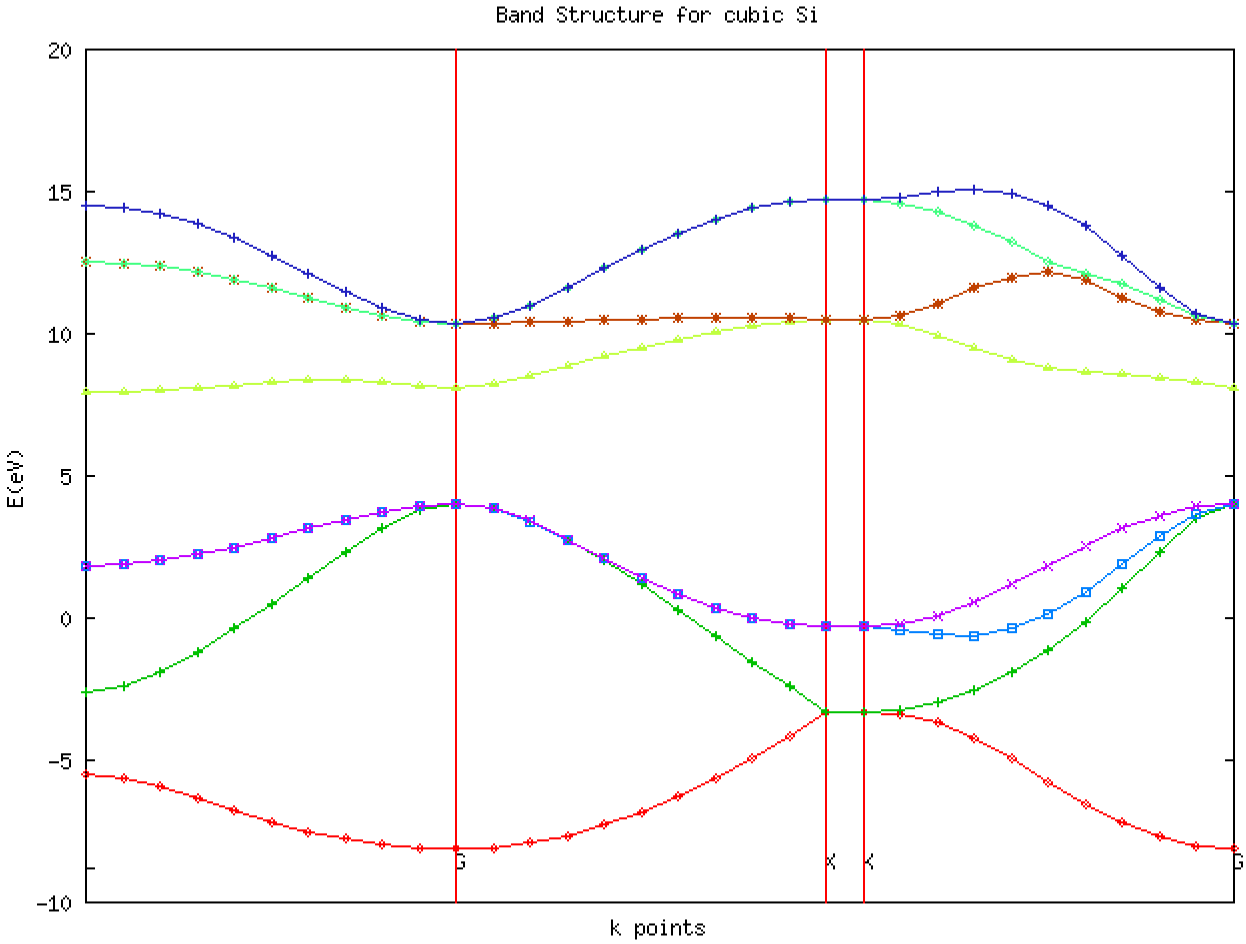
Tight Binding Programs for Computing the Band Structure of Semiconductors

Band Structure of Si computed with harrison.py and chadicohen.py programs:





Click here for the harrison.py program, which computes the band structure of semiconductors.

I've also written a slight modification of this program, called "chadicohen.py", which can be obtained here. This program is the tight binding program that Chadi and Cohen outline in their 1975 paper.

Click here for instructions on how to run either of these programs under Windows95, 98, or NT.

The following is from the notes in the programs.

harrison.py:

Tight-binding band structure of II-VI, III-V, and IV semiconductors. Based on Harrison's version of Chadi/Cohen's approach.

usage:

harrison.py [options]

Options:

```
-c #  The atomic number for the semiconductor cation (default=14)
-a #  The atomic number for the semiconductor anion (default=14)
-n #  The number of points in each Brillouin zone region
      (default=10)
-h    Print this help screen and exit
-P    Output a postscript image of the band structure
-G    Output a GIF of the band structure
```

chadicohen.py:

Tight binding structure for C, Si, Ge, GaAs, and ZnSe semiconductors. Based on Chadi and Cohen's 1975 paper.

usage:

chadicohen.py [options]

Options:

```
-s xx The Structure to compute; currently supported are:
      C     Diamond
      Si    Silicon
      Ge    Germanium
      GaAs  Gallium Arseninde
      ZnSe  Zinc Selenide
-n #  The number of points in each Brillouin zone region
      (default=10)
-h    Print this help screen and exit
-P    Output a postscript image of the band structure
-G    Output a GIF of the band structure
```

Caveats:

1. The parameters in the code are simply taken from the respective references. No checking is performed to make sure that they work for the case of interest
2. Similarly, no checking is done to insure that the species you input make sense in a diamond structure. I.e., you could input GaAl to harrison.py and the program would give you a (meaningless) answer.
3. This program assumes that Gnuplot is installed, and is started by the command "gnuplot". If this isn't the case on your system edit path_to_gnuplot accordingly
4. This program assumes that /usr/bin/env python can find python on your system. If not, edit the first line of this file accordingly.
5. This program assumes that the Numeric Extensions to Python (see http://numpy.sourceforge.net) are installed, and are in your $PYTHONPATH.

References:

1. D.J. Chadi and M.L. Cohen, *Tight Binding Calculations of the Valence Bands of Diamond and Zincblende Crystals.* Phys. Stat. Sol. (b) 68, 405 (1975).
2. W.A. Harrison, *Electronic Structure and the Properties of Solids: The Physics of the Chemical Bond.* Dover Publications, Inc., NY, 1989

General program notes:

The band gap here is greatly over-estimated (by almost exactly a factor of two, which suggests an obvious workaround). No one seems to publish tight-binding band gaps, so maybe they all do a bad job. The width of the valence band compares well with Chadi and Cohen. They, however, see much more dispersion. I suppose this is an exercise in fitting.

Here are some sample band gaps (in eV, from Kittel) that may aid in fitting:

```
C    indirect  5.4
Si   indirect  1.17
Ge   indirect  0.744
Sn   direct    0.00
GaAs direct    1.52
GaP  indirect  2.32
GaN  ??        3.5  (not from Kittel)
InP  direct    1.42
InAs direct    0.43
```

Last modified: Thu Mar 17 08:07:32 2005