# LMTO tutorial

Prepared by J. M. Allred

You will need:
Physical or remote access to cava-lmto computer.
Crystallographic data

1. Open up a terminal, and create a new directory for your new crystal structure.
   ```
   $mkdir ThNi2
   ```

2. run lminit.run. in the new directory:
   ```
   $cd ThNi2
   $lminit.run
   ```
   lminit.run looks for an INIT file in the directory you run it in. The INIT file only contains a few lines, and includes the space group, atomic sites, and lattice parameters of your compound. *If there is no INIT file in the current directory, it will prompt you for information about your compound*. Since we're starting from scratch, this is what will happen.

   i) The first question is the space group. You can enter it either by its symbol or number. E.g. 'P-1' or '2' (without the quotes) are both acceptable arguments. The program will either yell at you for making a mistake, or accept the response and tell you what you entered (which is nice if you enter the number and want to make sure it corresponds to the correct space group)

   ii) Now it wants the lattice parameters. Enter in 'f' so that the program knows that you are going to give it responses in angstroms (unless you really want to enter your parameters in atomic units…). Now enter your 'a' parameter.

   iii) Depending on your space group, it may also ask for 'b', 'c', and various angles. The program is smart enough to know what to ask for. If you say it is a cubic space group, it only asks for 'a'.

   iv) After the program finishes asking for lattice parameters, it asks for your atoms types and sites. You enter the first atom (e.g. 'Th'), and it asks for its coordinates. You enter the coordinates separated by spaces and then press enter. Typing '/' accepts the default position, which is the origin. So I could type '0 0 0' or '/'.

   v) Now it asks for the next atom. Keep entering new atoms until you account for all the inequivalent atoms. The program knows how to deal with special sites and symmetry operations, so it will automatically generate the extra atoms later.

   vi) When you have entered all your atoms, you press 'q' to tell the program you are done. DO NOT press 'a', as this discards everything and you need to restart. After you press 'q' the lminit.run makes your INIT file, and then it generates your control file: ./CTRL.

This is the file you'll be dealing with the most.

3. If you feel like it, you can now use your favorite text editor to look at your CTRL and INIT files to make sure everything is correct. (If you don't know any text editors, gedit is easy to use) It's not necessary, but is an opportunity to learn how things look if you feel like spending the time. There is not that much information in the files yet.

4. Now run lmhart.run and then lmes.run.

```
$lmhart.run
$lmes.run
```

The output is directed to a file called 'LM' and errors are directed to 'ERR'. lmhart.run deals with constructing the spherical potentials around your atomic sites, and figuring out how much space is filled and whether any atoms overlap too much. lmes.run then adds empty spheres to the remaining space so that your volume is 100% filled. To make sure this went ok check your output. You can either open LM and search for a line about VOL = 100%,

```
$gedit LM&
```

or just grep the line:

```
$grep 'VOL' LM
```

If everything went well, you will see the happy VOL= 100% and you can continue. What do you do if something else happened?

i) Look at the rest of your LM and ERR files. There might be some indication that some atoms are too close together or something. This means you might need to look at your crystal data and make sure it's right. If you made a mistake, you can go into your INIT file and change the appropriate value. Run lminit.run again, and the new INIT file will be used to rewrite your CTRL file.

5. Ok, so now that stuff is all out of the way. You could start your refinement right now if you wanted, but that would require accepting all sorts of defaults. BORING! Instead you should go ahead and open up your CTRL file and start tweaking the options.

```
$gedit CTRL&
```

i) Notice that there still is not much information in this, so we need to change that. Near the top there is a line that says: VERBOS = 30. Change it to VERBOS = 70. The number is a token, and all you need to know is that 70 tells the program called 'lmctl.run' that we want more information in CTRL. Save the file and close it.

ii) Now run lmctl.run.

```
$lmctl.run
```

This program gathers information from CTRL and other files and rewrites CTRL. It sees that VERBOS=70, and does its magic. Open CTRL again. Note that there is *a lot* more information. The first thing to look at is the new OPTIONS section. A few of the more used options are NONLOC, (which tells lm.run to use gradient corrections) FATBAND (which tells it to make the EIGN file so that fatband structures can be calculated), and FS (which prepares the necessary information for looking at the Fermi surface). NSPIN has to do with the number of spins per unit cell. For now, only change FATBAND and NONLOC to T.

iii) Near the bottom there are a few other things to look at. Computers are more powerful than they once were, so you can get away with using more data points than default. The 3 numbers after NKABC = can all be multiplied by 3 or 4. Something as high as 6 is probably worth it, actually. Make sure you multiply all of them by the same amount. 6 6 6 becomes 24 24 24. 4 4 8 becomes 24 24 32.

iv) Change DOS from 801 to a number between 2500 and 3000.

v) Eventually you will plot your band structure. The instructions for which points to use are found at the end of the CTRL file. You can change these to suit your needs. It's worth it to multiply the number of data points between each special point (the first number on each line) by 1.5 or 2. This helps make the actual plot prettier.

6. Ok. Now try running your first calculation. Type lm.run, and the refinement will start.
   **$lm.run**
   Depending on how high you made NKABC and the complexity of your compound it may take awhile. Once it is done, check your LM file for convergence. The best way to do this is "grep 'RMS' LM". Unless something went very wrong, you'll see two columns, one called dq and the other detot. If the refinement succeeded, dq should continue to get smaller and smaller until it is close enough to zero (usually .00005ish).
   **$grep 'RMS'**

   Also, the program has a special line that it outputs in case it "thinks" that it converged. Try typing
   **$grep 'Jolly'**
   and see what happens…

7. Well that's nice, but what if the compound is magnetic? So far we haven't done anything to take that into account. Let's try that now.

   i) Create a subdirectory in this directory. I usually call it 'ferro'.
   **$mkdir ferro**
   ii) Copy your CTRL file into the new directory. You may need to copy over POT and STR over, too.
   **$cp CTRL POT STR ./ferro**
   iii) Inside CTRL, change it so that NSPIN = 2 and you should probably make sure that FATBANDS = F (saves hard drive space…).
   **$cd ferro**
   **$gedit CTRL&**

   iv) Now save, but before you exit, go down to the atom you think is magnetic and look at the state distribution underneath it. These are s,p,d,f in that order, although some elements don't have f, or even d sometimes. If this is a transition element, then the d row number is likely much larger than the others.

   v) Run lmctl.run. This doubles the number of states for each atom. Open CTRL, and go down to the atom(s) that could possibly be magnetic and notice that it literally copied the lines from before over again so that they appear twice. If you think it is the d electrons that could be magnetic, than increase one number by one, and decrease the other by one.

```
$lmctl.run
$gedit CTRL&
```

vi) Now run lm.run again. This time it will also calculate the overall magnetic moment.

```
$lm.run
```

vii)When it is done, look at the LM file to find out what happened. First look for convergence: grep “RMS” LM. Then check the magnetic moment.

```
$grep “MAG “ LM
or
$grep “Jolly” LM
```

Note the space after MAG. If the number converges to zero than the compound is *probably* not magnetic. Sometimes this program underestimates magnetic activity, and even more so if you don’t make NKABC large enough. If MAG sticks around then you have more work to do. For now let’s assume it is nonmagnetic.

8. Go back to your original directory for this calculation and run lmbnd.run and lmdos.run. This does the majority of the work for viewing the DOS and band plots.

```
$cd ..
$lmbnd.run
$lmdos.run
```

9. To create a postscript view of your DOS plot, type ‘gnudos.exec’

```
$gnudos.exec
```

Choose ‘1’ for postscript file. Now you select your options. Choose your energy scale, range, DOS range, electron state you want to plot over, etc. If you want to accept a default choise, type ‘/’. The output is always dos.ps, so you should rename it so that it can be saved later. You can use a postscript viewer like ghostview to view and print the file. If you want to just print without viewing, the print from terminal command is ‘lpr dos.ps’. You may also want to change the name of the file so that it doesn’t get saved over if you make a new plot.

```
$ghostview dos.ps
$lpr dos.ps
$cp dos.ps new_dosplot.ps
```

**10.** The band structure is similar. Type ‘gnubnd.exec’

```
$gnubnd.exec
```

Choose ‘1’ again. The first two options are your choice, but I recommend you make the next two ‘f’. The 5th question about plotting band character appears only if you selected FATBANDS=T in your CTRL file. If you select ‘t’ here you have to decide which orbitals you are using, which requires selecting each atom and its orbitals. Select what you want, and finally choose the energy range you want to plot over. This file is saved has bnds.dos.

```
$ghostview bnds.ps
$lpr bnds.ps
$cp bnds.ps new_bndplot.ps
```

**Linux for beginners.**
**File structure.**
**When you log onto the computer, your home directory is**

**/home/[login_name]/**

**However, in the terminal it abbreviates your home as ~/**
**So if you are in /home/[login_name]/mergh/, it will just say ~/mergh/**
**The current directory is abbreviated as ./**
**The parent directory is abbreviated as ../**

**Note: the directory divider / is different than the windows/DOS divider \**

**Linux commands:**

**cd foo**

**Changes directory to ./foo/ ; cd .. goes back one directory (because .. is the parent directory)**

```
~/ $ cd mergh
~/mergh/ $ cd ..
~/ $
```

**To go back two directories, use ../../ (and so forth)**

**cp foo bar**

**Copies foo. If the last argument (bar) is an existing folder, foo keeps the same name but is now in ./bar/, but if bar is not an existing folder and foo and bar are the only arguments than bar will be a new file that is a copy of foo. Also, foo can be a list of files.**

**emacs**

**Advanced text editor.**

**grep foo bar**

**Returns lines containing ‘foo’ in file bar**

**gedit**

**Basic text editor utility**

**mkdir foo**

**Makes a directory called foo**

**rm foo**

**Deletes a file called foo**

**rmdir foo**

Removes a directory called foo

Meaningful symbols:

&

This command will not disrupt usage of the terminal (more or less).