Building the High Performance Linpack (HPL) Benchmark with GotoBLAS

Note: Things that you should type are in the **`computer boldface`** font.

1. Before doing this, you **MUST** have already installed GotoBLAS and know the directory that it's located in.
2. Open up a web browser.
3. Go to your favorite search engine (e.g., Google).
4. Search for:

   "High Performance Linpack"
5. On the HPL benchmark webpage, scroll down and download the source code for HPL (for example, `hpl-2.0.tar.gz`).
6. Copy the location of that link.
7. Log in to the computer that you want to build HPL on.
8. Create an HPL directory:

   `mkdir HPL`
9. Go into that directory:

   `cd HPL`
10. Create a directory named for the implementation of BLAS you're using (in this case, GotoBLAS):

    `mkdir GotoBLAS`
11. Go into that directory:

    `cd GotoBLAS`
12. Create a directory named for the implementation of MPI that you're going to use (in this case, OpenMPI):

    `mkdir OpenMPI`
13. Go into that directory:

    `cd OpenMPI`
14. Download the HPL source code using this command; for example:

    `wget http://www.netlib.org/benchmark/hpl/hpl-2.0.tar.gz`
15. "Untar" the compressed tar file:

    `tar zxvf http://www.netlib.org/benchmark/hpl/hpl-2.0.tar.gz`
16. Go into the newly created HPL directory:

    `cd hpl-2.0`
17. Determine your current working directory:

    `pwd`
18. Read the file named **`INSTALL`**, which has instructions for how to install HPL:

    `more INSTALL`

    If you're unfamiliar with the `more` command, ask someone for help.

19. Copy an appropriate `Make.something` from the subdirectory named `setup` into the current working directory, naming the new copy with the name of the machine you're on; for example:

    `cp setup/Make.Linux_PII_CBLAS Make.Sooner`

20. Using your preferred text editor (for example, `vi`, `emacs`, `nano`), edit the new copy (for example, `Make.Sooner_GotoBLAS_OpenMPI`) as follows:

    a. Change the value of `ARCH` to the part of this file's name after the dot (for example, `Sooner`).

    b. Change the value of `TOPdir` to the current working directory.

    c. Change the values of `MPdir`, `MPinc` and `MPlib` to be blank (after the equals sign).

    d. Change the value of `LAdir` to the directory containing the GotoBLAS libraries.

    e. Change the value of `LAlib` to:

    `-L$(LAdir) -lgoto2`

    f. Change the value of `CC` to `mpicc`.

    g. Change the value of `LINKER` to `mpif77`.

21. If you're on Sooner, then set the environment variables for the compiler and interconnect driver software.

    If the Unix shell that you're using is tcsh, that'd be:

    `setenv MPI_COMPILER gnu`

    `setenv MPI_INTERCONNECT ib`

    `setenv MPI_VENDOR openmpi`

    If the Unix shell you're using is bash, that'd be:

    `export MPI_COMPILER=gnu`

    `export MPI_INTERCONNECT=ib`

    `export MPI_VENDOR=openmpi`

22. Do the build using the make command; for example:

    `make arch=Sooner_GotoBLAS_OpenMPI`

    This will take several minutes.

    NOTE: If anything goes wrong, do the following:

    a. Copy your `Make.something` file to somewhere else; for example:
    `cp Make.Sooner_GotoBLAS_OpenMPI /tmp`

    b. Go up to the parent directory:
    `cd ..`
    NOTE: The two periods ("dotdot") at the end of the `cd` command mean "to the parent of this directory" and are **VERY IMPORTANT**.

    c. Delete the entire HPL directory:
    `rm -rf hpl-2.0`

    d. Untar the tar file again (step 15, above).

    e. Go into the newly untarred directory (step 16, above).

f. Copy the `Make.something` file back into that directory; for example:

```
cp /tmp/Make.Sooner_GotoBLAS_OpenMPI .
```

NOTE: The period ("dot") at the end of the cp command means "to the current working directory" and is **VERY IJMPORTANT**.

g. Repeat steps 20-22 as needed.

23. Check that the output from the `make` command shows that you built HPL properly.