A Compiling and Running MPI Programs

This section is intended to give the *barest* outline of how to compile and run a program using each of the freely available versions of MPI. Please consult the documentation that comes with these packages for further details.

In each case, we assume that you wish to run your program on a homogeneous network of UNIX workstations, and that the executables, libraries, and header files have been installed in a public directory on the machines on which you are compiling and executing your program.

A.1 MPICH

Here, we assume that the MPICH files are stored in the following files.

- Executables: /usr/local/mpi/bin
- Libraries: /usr/local/mpi/lib
- Header files: /usr/local/mpi/include

To compile the C source program prog.c, you should type

```
% cc -o prog prog.c -I/usr/local/mpi/include\
      -L/usr/local/mpi/lib -lmpi
```

In order to run the program with, say, 4 processes, you should first copy the executable to your home directory on each machine (unless the directory is NFS mounted), and then type

```
% mpirun -np 4 prog
```

This assumes that your system has a generic configuration file that lists machines on which MPI programs can be run.

A.2 CHIMP

Before using CHIMP, you need to be sure the CHIMP home directory is in your path on all the machines on which you intend to run MPI programs. For example, if the CHIMP home directory is /home/chimp on each machine, and you use csh, you should add the following lines to your .cshrc on each machine.

```
setenv CHIMPHOME /home/chimp
set PATH $CHIMPHOME/bin:$PATH
```

After modifying your .cshrc file, you should change to your home directory on each machine and execute the following commands.

```
% cd
% source .cshrc
% ln -s $CHIMPHOME/chimprc .chimpv2rc
```

Note that these commands only need to be carried out once — when you use CHIMP again, you can skip these steps.

If your MPI source program is called prog.c, you can compile it with

```
% mpicc -o prog prog.c
```

Before executing your program, you need to create a CHIMP configuration file. This contains a list of the executables, hosts on which to run the program, and directories containing the executables. Its basic format is a list of lines having the form:

```
(<executable>): host=<hostname>, dir=<directory>
```

For example, to run prog on four machines, we might create a file called prog.config that contains the following lines.

```
(prog): host=mobydick, dir=/home/peter
(prog): host=kingkong, dir=/home/peter
(prog): host=euclid, dir=/home/peter
(prog): host=lynx, dir=/home/peter
```

In order to run the program, first copy the executable to the appropriate directory on each machine (unless the directory is NFS mounted), and then type

```
% mpirun prog.config
```

A.3 LAM

Before starting, make sure that the directory containing the LAM executables is in your path on each machine on which you intend to run your program. For example, if the LAM executables are in /usr/local/lam/bin and you use csh, you can simply add the following commands to your .cshrc file.

```
setenv LAMHOME /usr/local/lam
set PATH $LAMHOME/bin:$PATH
```

After modifying your .cshrc file, you should change to your home directory on each machine and execute the following commands.

```
% cd
% source .cshrc
```

Note that these commands only need to be carried out once — when you use LAM again, you can skip these steps.

Next create a file listing the names of the hosts on which you intend to run MPI. For example, a 4 host file might contain the following lines.

```
mobydick.usfca.edu
kingkong.math.usfca.edu
euclid.math.usfca.edu
lynx.cs.usfca.edu
```

If this file is called lamhosts, the command recon verifies that LAM can be started on each machine.

```
% recon -v lamhosts
recon:  testing n0 (mobydick.usfca.edu)
recon:  testing n1 (kingkong.math.usfca.edu)
recon:  testing n2 (euclid.math.usfca.edu)
recon:  testing n3 (lynx.cs.usfca.edu)
```

To actually start up LAM on each machine, type

```
% lamboot -v lamhosts
LAM - Ohio Supercomputer Center
hboot n0 (mobydick.usfca.edu)...
hboot n1 (kingkong.math.usfca.edu)...
hboot n2 (euclid.math.usfca.edu)...
hboot n3 (lynx.cs.usfca.edu)...
```

In order to compile your program, type

```
% hcc -o prog prog.c -lmpi
```

In order to run the program, first copy the executable to your home directory on each machine (unless the directory is NFS mounted), and then type

```
% mpirun -v n0-3 prog
1362 prog running on n0 (o)
14445 prog running on n1
12687 prog running on n2
1433 prog running on n3
```

To shut down LAM, type

```
% wipe -v lamhosts
tkill n0 (mobydick.usfca.edu)...
tkill n1 (kingkong.math.usfca.edu)...
tkill n2 (euclid.math.usfca.edu)...
tkill n3 (lynx.cs.usfca.edu)...
```

References

[1] Geoffrey Fox, et al., *Solving Problems on Concurrent Processors,* Englewood Cliffs, NJ, Prentice–Hall, 1988.

[2] William Gropp, Ewing Lusk, and Anthony Skjellum, *Using MPI: Portable Parallel Programming with the Message-Passing Interface,* Cambridge, MA, MIT Press, 1994.

[3] Brian W. Kernighan and Dennis M. Ritchie, *The C Programming Language,* 2nd ed., Englewood Cliffs, NJ, Prentice–Hall, 1988.

[4] Message Passing Interface Forum, *MPI: A Message-Passing Interface Standard* , International Journal of Supercomputer Applications, vol 8, nos 3/4, 1994. Also available as Technical Report CS-94-230, Computer Science Dept., University of Tennessee, Knoxville, TN, 1994.

[5] Steve Otto, et al., *MPI Annotated Reference Manual,* Cambridge, MA, MIT Press, to appear.

[6] Peter S. Pacheco, *Parallel Programming with MPI,* San Francisco, CA, Morgan Kaufmann, 1997.