# LINPACK性能测试，实验报告

## 一. 实验目的

* 掌握 Linpack 和 HPL 的背景知识
* 完成 HPL 的安装和配置
* 运行 HPL，并调整相关参数分析集群性能

## 二.实验环境

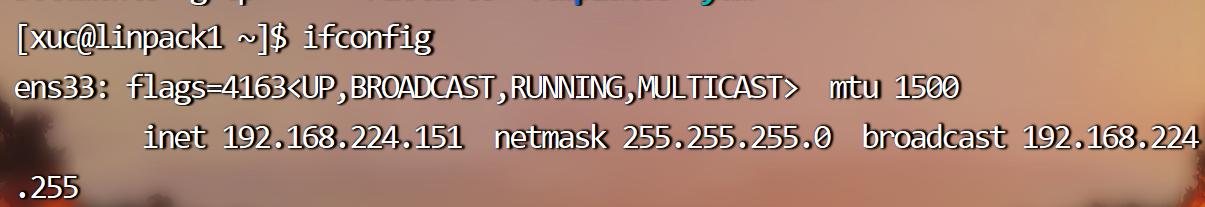
* CentOS7 64位操作系统（CentOS Linux release 7.6.1810）（运行于虚拟机）
* 虚拟机使用了3台，为每台虚拟机设置为1块双核CPU
* VMware® Workstation 16 Pro-16.2.4 build-20089737 虚拟机软件
* FinalShell 3.9.8.2 终端软件，通过SSH与Linux虚拟机连接
* 硬件环境：
* 设备名称 martinred  
  处理器 AMD Ryzen 7 6800H with Radeon Graphics 3.20 GHz  
  机带 RAM 16.0 GB (15.2 GB 可用)  
  设备 ID 5FBC4275-DA53-4E4A-890E-0B2D7B2246AD  
  产品 ID 00342-30700-55566-AAOEM  
  系统类型 64 位操作系统, 基于 x64 的处理器

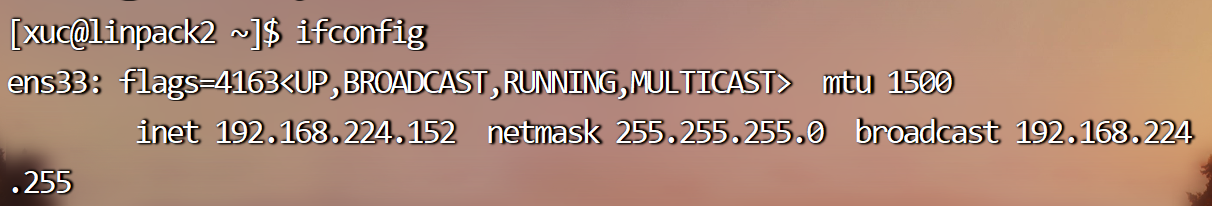
## 三.实验内容

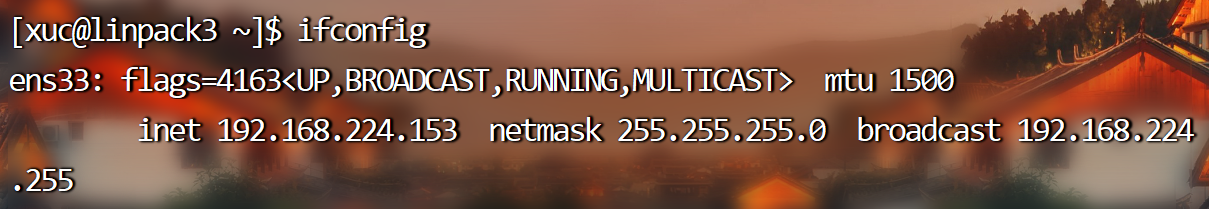
### 1. 配置3台虚拟机的ip地址和hostname

| hostname | ip address |
| --- | --- |
| linpack1 | 192.168.224.151 |
| linpack2 | 192.168.224.152 |
| linpack3 | 192.168.224.153 |

配置完成后展示：



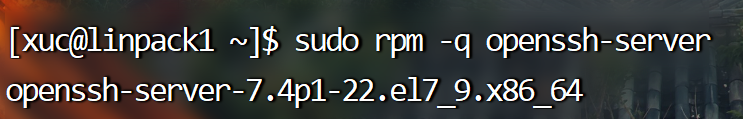


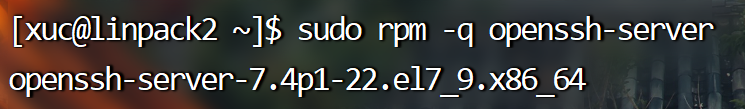


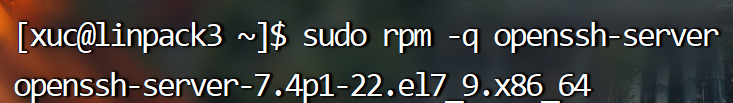
### 2. 安装SSH

实验1已经完成。

验证虚拟机的SSH是否安装好：



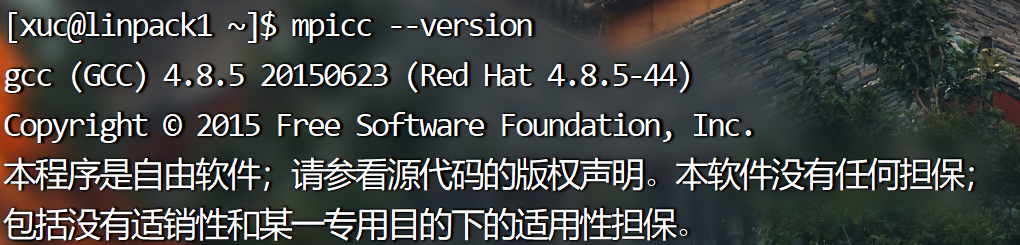


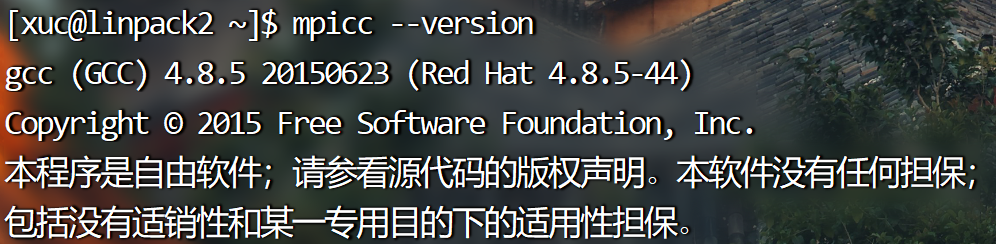


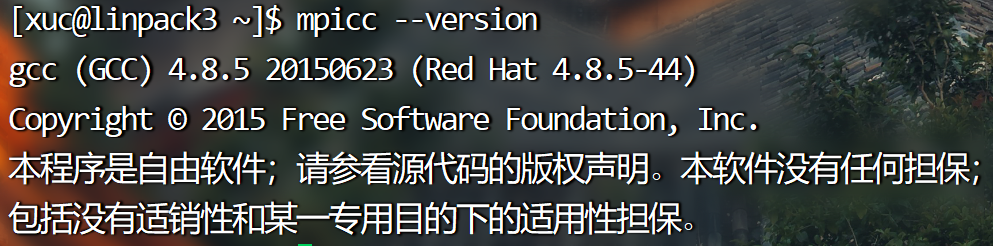
### 3. 安装openmpi

实验3已经完成。

验证虚拟机的OpenMPI是否安装好：





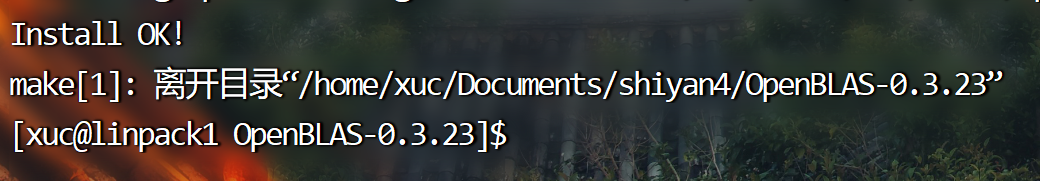


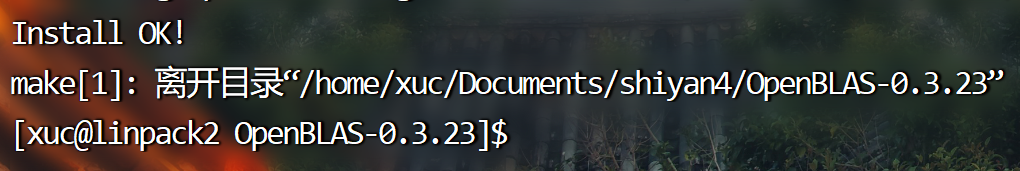
### 4. 安装OpenBLAS

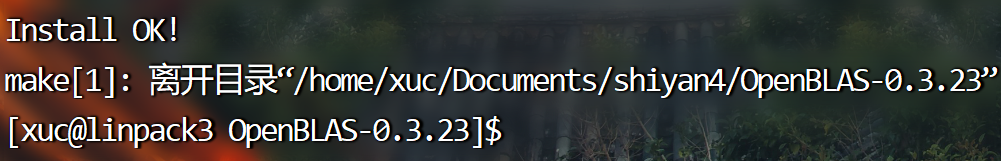
安装操作的指令为：

tar -xf OpenBLAS-0.3.23.tar.gz # 解压缩OpenBLAS源代码包
  
cd OpenBLAS-0.3.23 # 进入解压后的OpenBLAS目录
  
sudo make # 执行配置命令
  
sudo make PREFIX=/usr/local install # 执行安装命令

安装完成后显示：







### 5. 安装HPL并进行基准测试

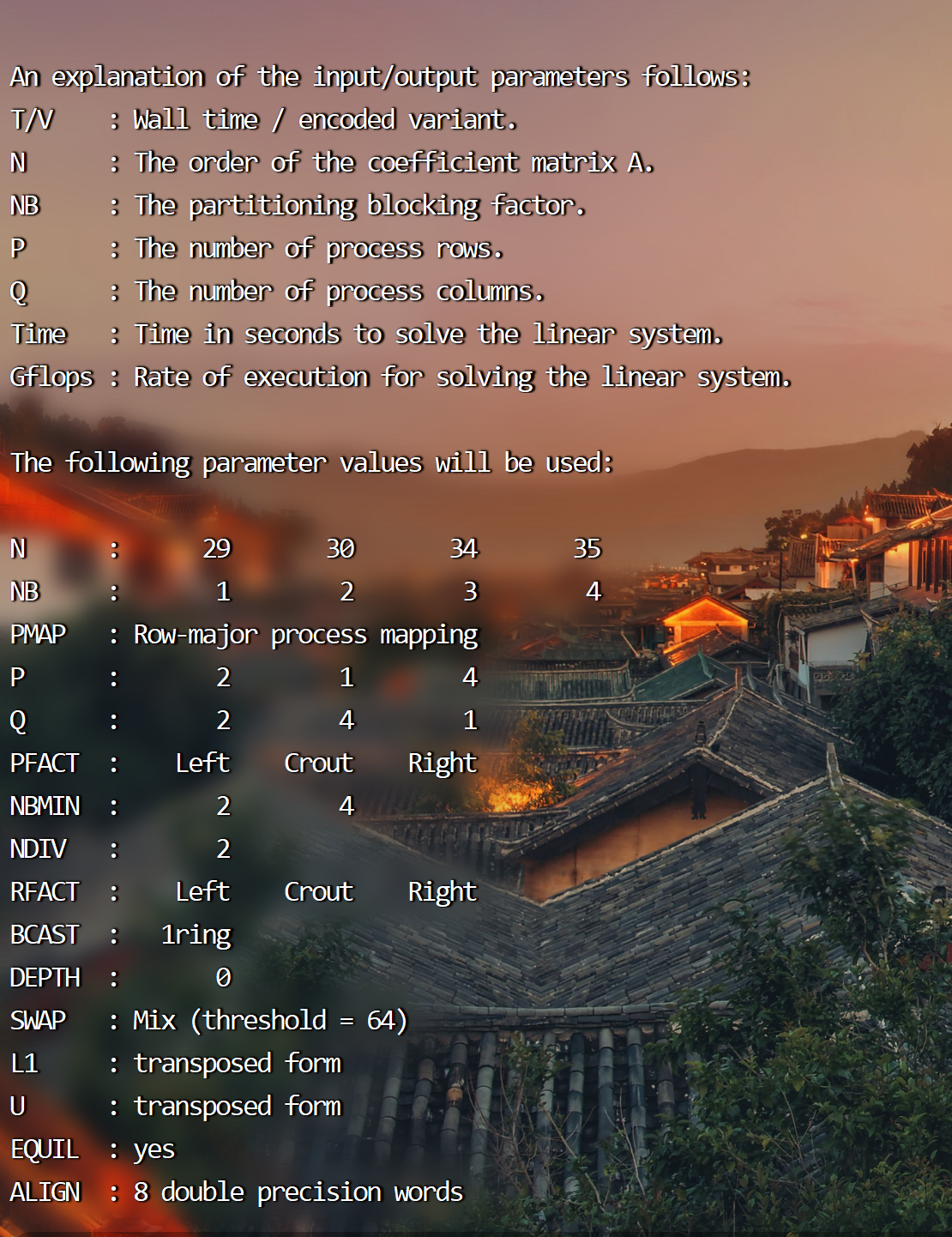
使用如下命令：

tar -xf hpl-2.3.tar.gz # 解压缩HPL源代码包
  
cd hpl-2.3 # 进入解压后的HPL目录
  
cp ./setup/Make.Linux\_PII\_FBLAS Make.Linux\_PII\_FBLAS # 选择模板文件，将其复制为Make.Linux\_PII\_FBLAS
  
vim Make.Linux\_PII\_FBLAS # 使用文本编辑器打开Make.Linux\_PII\_FBLAS文件，进行配置
  
make arch=Linux\_PII\_FBLAS # 运行make命令进行编译，使用Linux\_PII\_FBLAS作为架构选项
  
cd ./bin/Linux\_PII\_FBLAS/ # 进入生成的可执行文件所在目录Linux\_PII\_FBLAS
  
mpirun --allow-run-as-root -np 4 ./xhpl > 1.txt # 使用mpirun命令运行HPL程序，使用4个进程，并将输出重定向到1.txt文件

Make.Linux\_PII\_FBLAS文件修改的地方如下：

70 TOPdir = $(HOME)/hpl-2.3 # HPL顶级目录，存放HPL文件的位置
  
 84 MPdir = /usr/lib64/openmpi # OpenMPI安装目录
  
 86 MPlib = $(MPdir)/lib/libmpi.so # OpenMPI库文件路径
  
 95 LAdir = /usr/local/lib # OpenBLAS安装目录
  
 97 LAlib = $(LAdir)/libopenblas.a # OpenBLAS库文件路径
  
169 CC = /usr/lib64/openmpi/bin/mpicc # C编译器路径
  
171 CCFLAGS = $(HPL\_DEFS) -fomit-frame-pointer -O3 -funroll-loops -W -Wall -pthread # C编译器标志
  
176 LINKER = /usr/lib64/openmpi/bin/mpif77 # 链接器路径

执行完成后的1.txt文件内容为：



### 6. 集群系统性能调优以及分析

实验过程所用命令展示：

cd /root/hpl-2.3/bin/Linux\_PII\_FBLAS
  
vim nodes
  
vim HPL.dat
  
mpirun --allow-run-as-root -np 4 --machinefile nodes --mca btl\_tcp\_if\_include ens33 ./xhpl > data.txt

在HPL测试中，使用的参数选择与测试的结果有很大的关系。HPL中参数的设定是通过从一个配置文件HPL.dat中读取的，所以在测试前要改写HPL.dat文件，设置需要使用的各种参数，然后再开始运行测试程序。配置文件HPL.dat内容的结构如下：

HPLinpack benchmark input file
  
Innovative Computing Laboratory, University of Tennessee
  
HPL.out output file name (if any) # 输出文件名（如果有）
  
6 device out (6=stdout,7=stderr,file)
  
4 # 问题规模(N)
  
29 30 34 35 Ns # 问题大小的具体数值（Ns）
  
4 # NB的数量（块大小）
  
1 2 3 4 NBs # NB的具体数值（NBs）
  
0 PMAP process mapping (0=Row-,1=Column-major) # PMAP进程映射（0=行主序，1=列主序）
  
3 # 进程网格的数量（P x Q）
  
2 1 4 Ps # 进程网格的行号（Ps）
  
2 4 1 Qs # 进程网格的列号（Qs）
  
16.0 threshold # 阈值
  
3 # 面板因子化的数量
  
0 1 2 PFACTs (0=left, 1=Crout, 2=Right) # 面板因子化的具体数值（PFACTs）
  
2 # 递归停止准则的数量
  
2 4 NBMINs (>= 1) # 递归停止准则的具体数值（NBMINs）
  
1 # 递归中面板的数量
  
2 NDIVs # NDIV的具体数值
  
3 # 递归面板因子化的数量
  
0 1 2 RFACTs (0=left, 1=Crout, 2=Right) # 递归面板因子化的具体数值（RFACTs）
  
1 # 广播的数量
  
0 BCASTs (0=1rg,1=1rM,2=2rg,3=2rM,4=Lng,5=LnM) # 广播的具体数值（BCASTs）
  
1 # 向前看的深度的数量
  
0 DEPTHs (>=0) # 向前看的深度的具体数值（DEPTHs）
  
2 SWAP (0=bin-exch,1=long,2=mix) # SWAP类型（0=bin-exch,1=long,2=mix）
  
64 swapping threshold # 交换阈值
  
0 L1 in (0=transposed,1=no-transposed) form # L1是否以转置形式存储（0=转置，1=非转置）
  
0 U in (0=transposed,1=no-transposed) form # U是否以转

要得到调试出高的性能，必须考虑内存大小，网络类型以及拓扑结构，调试上面的参数，直到得出最高性能。本次实验需要对以下三组参数进行设置：

2 # of problems sizes (N)
  
1960 2048 Ns # 指出要计算的矩阵规格有2种，规格是1960，2048
  
2 # of NBs
  
60 80 NBs # 指出使用2种不同的分块大小，大小为60，80
  
2 # of process grids (P x Q)
  
2 4 Ps # 指出用2种进程组合方式
  
2 1 Qs # 分别为（p＝2，q=2） 和（p＝4，q=1）

注：p＝2，q=2时需要的进程数是p×q＝2×2＝4，运行时mpirun命令行中指定的进程数必须大于等于4。

以上3组每组有两种情况，组合后一共有8种情况，将得到8个性能测试值，经过不断的调试将会得出一个最大的性能值，这就是得到的最高性能值。

以下是将会测试的参数：

The following parameter values will be used:
  
  
N : 1960 2048
  
NB : 60 80
  
PMAP : Row-major process mapping
  
P : 2 4
  
Q : 2 1
  
PFACT : Left Crout Right
  
NBMIN : 2 4
  
NDIV : 2
  
RFACT : Left Crout Right
  
BCAST : 1ring
  
DEPTH : 0
  
SWAP : Mix (threshold = 64)
  
L1 : transposed form
  
U : transposed form
  
EQUIL : yes
  
ALIGN : 8 double precision words

以下是其中一个性能测试值，规格为1960，分块是60，p＝2，q=2时，运行时间为：2.96，运算速度为1.6971Gflops。PASSED代表结果符合要求。

T/V N NB P Q Time Gflops
  
--------------------------------------------------------------------------------
  
WR00L2L2 1960 60 2 2 2.96 1.6971e+00
  
HPL\_pdgesv() start time Thu May 11 23:23:10 2023
  
  
HPL\_pdgesv() end time Thu May 11 23:23:13 2023
  
  
--------------------------------------------------------------------------------
  
||Ax-b||\_oo/(eps\*(||A||\_oo\*||x||\_oo+||b||\_oo)\*N)= 2.60744810e-03 ...... PASSED

以下是各参数组合得到的平均测试结果，可以看出，在N=2048,NB=80,P=2,Q=2时有最优秀的性能，每秒钟可以执行的浮点运算次数最高。

|  | N | NB | P | Q | Gflops |
| --- | --- | --- | --- | --- | --- |
| 0 | 1960 | 60 | 2 | 2 | 1.367705 |
| 1 | 1960 | 60 | 4 | 1 | 0.763215 |
| 2 | 1960 | 80 | 2 | 2 | 1.482946 |
| 3 | 1960 | 80 | 4 | 1 | 0.897765 |
| 4 | 2048 | 60 | 2 | 2 | 1.576211 |
| 5 | 2048 | 60 | 4 | 1 | 1.332594 |
| 6 | 2048 | 80 | 2 | 2 | 1.562788 |
| 7 | 2048 | 80 | 4 | 1 | 0.975348 |

本次实验主要分析的是参数调优，可以通过调整 HPL.dat 测试中的参数配置，如网格分块的维度（P）和每个分块内的处理器数量（Q），进行测试，从而找到最佳的参数组合。

### 7. 性能测试

#### 7.1 线程性能测试

以下为N=2048,NB=64,P=2,Q=2时不同线程数的平均测试结果：

|  | Threads | Gflops |
| --- | --- | --- |
| 0 | 4 | 1.949234 |
| 1 | 5 | 2.093354 |
| 2 | 6 | 2.060867 |
| 3 | 7 | 6.401161 |
| 4 | 8 | 5.916823 |
| 5 | 9 | 6.152068 |
| 6 | 10 | 16.847124 |
| 7 | 11 | 16.800333 |
| 8 | 12 | 16.805681 |

#### 7.2 矩阵规模测试

以下为4线程,NB=64,P=2,Q=2时不同规模的平均测试结果：

|  | N | Gflops |
| --- | --- | --- |
| 0 | 64 | 0.006532 |
| 1 | 128 | 0.014667 |
| 2 | 256 | 0.053258 |
| 3 | 512 | 0.152053 |
| 4 | 1024 | 0.592636 |
| 5 | 2048 | 2.056832 |
| 6 | 4096 | 7.256893 |
| 7 | 8192 | 16.258892 |

### 7.3 分块测试

|  | N | NB | P | Q | Gflops |
| --- | --- | --- | --- | --- | --- |
| 0 | 1960 | 60 | 2 | 2 | 1.367705 |
| 1 | 1960 | 60 | 4 | 1 | 0.763215 |
| 2 | 1960 | 80 | 2 | 2 | 1.482946 |
| 3 | 1960 | 80 | 4 | 1 | 0.897765 |
| 4 | 2048 | 60 | 2 | 2 | 1.576211 |
| 5 | 2048 | 60 | 4 | 1 | 1.332594 |
| 6 | 2048 | 80 | 2 | 2 | 1.562788 |
| 7 | 2048 | 80 | 4 | 1 | 0.975348 |