第一次试验报告

王世奇 181870207

一、环境配置

1-1 docker下载

下载完成之后，确认是否安装成功：

```
wangsky@wangsky-virtual-machine:~$ docker --version
Docker version 19.03.13, build 4484c46d9d
```

显示安装成功

1-2 镜像拉取与默认设置

通过

```
docker pull ubuntu:14.04
```

直接拉取镜像

拉取镜像前，如果本机没有换源，可能导致下载过慢，导致出现timeout错误，如下

```
root@wangsky-virtual-machine:/home# docker pull ubuntu:14.04

Error response from daemon: Get https://registry-1.docker.io/v2/library/ubuntu/m
anifests/14.04: net/http: TLS handshake timeout
```

换源之后，即可解决这个问题

```
root@wangsky-virtual-machine:/home# vim /etc/docker/daemon.json
root@wangsky-virtual-machine:/home# service docker restart
root@wangsky-virtual-machine:/home# docker pull ubuntu:14.04
14.04: Pulling from library/ubuntu
2e6e20c8e2e6: Pull complete
95201152d9ff: Pull complete
5f63a3b65493: Pull complete
Digest: sha256:63fce984528cec8714c365919882f8fb64c8a3edf23fdfa0b218a2756125456f
Status: Downloaded newer image for ubuntu:14.04
docker.io/library/ubuntu:14.04
```

拉取之后，开启第一个容器：origin，来完成默认设置

```
docker run -dit --hostname origin --name origin ubuntu:14.04
```

进入该容器进行换源，换源之前先备份，按照助教的list文件出错，推测可能是版本问题，因此在网上重新找了一个

```
root@origin:/# sudo vi /etc/apt/sources.list
root@origin:/# sudo apt-get install libopenmpi-dev
Reading package lists... Error!
E: Type 'multiverse' is not known on line 2 in source list /etc/apt/sources.list
E: The list of sources could not be read.
E: The package lists or status file could not be parsed or opened.
```

阿里源

```
deb http://mirrors.aliyun.com/ubuntu/ xenial main
deb-src http://mirrors.aliyun.com/ubuntu/ xenial main

deb http://mirrors.aliyun.com/ubuntu/ xenial-updates main
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-updates main

deb http://mirrors.aliyun.com/ubuntu/ xenial universe
deb-src http://mirrors.aliyun.com/ubuntu/ xenial universe
deb http://mirrors.aliyun.com/ubuntu/ xenial-updates universe
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-updates universe

deb http://mirrors.aliyun.com/ubuntu/ xenial-security main
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-security main
deb http://mirrors.aliyun.com/ubuntu/ xenial-security universe
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-security universe
```

接下来配置mpi环境

```
sudo apt-get install libopenmpi-dev
sudo apt-get install openmpi-bin
```

生成公钥密钥，并将其加入自己的authorized_keys中，这样同一个镜像的不同容器之间可以互相访问

```
ssh-keygen -t rsa
```

通过origin 创建一个新镜像

```
docker commit -a werllen origin mpi-ubuntu:latest
```

查看已创建镜像如下：

```
root@wangsky-virtual-machine:/home# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED
 SIZE
mpi-ubuntu          latest              5287a9554060        About a minute ago
 491MB
ubuntu              14.04               df043b4f0cf1        7 days ago
 197MB
```

1-3 新建容器，构建集群

开两个容器，并挂载同一个共享文件夹

```
docker run -dit --hostname host1 --name mpi-host1 -v
/home/wangsky/tempbash:/root/mpidir mpiubuntu:latest
docker run -dit --hostname host2 --name mpi-host2 -v
/home/wangsky/tempbash:/root/mpidir mpiubuntu:latest
```

查看所有容器如下

```
root@wangsky-virtual-machine:/home/wangsky# docker ps -a
CONTAINER ID        IMAGE                COMMAND             CREATED
STATUS              PORTS                NAMES
78cf7ac56b11        mpi-ubuntu:latest    "/bin/bash"         24 hours ago
Up 24 hours                              mpi-host2
cf9a98bf577e        mpi-ubuntu:latest    "/bin/bash"         24 hours ago
Up 24 hours                              mpi-host1
462ecbffcd54        ubuntu:14.04         "/bin/bash"         46 hours ago
Up 46 hours                              origin
```

分别为host1和host2配置host文件，修改容器内host文件如下，在末尾加上另一个host的声明：

```
127.0.0.1       localhost
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
172.17.0.3      host1
172.17.0.4      host2
~
~
```

检测环境，分别编译一下hello.c，需要加上--allow-run-as-root

```
root@host2:~/mpidir# mpicc -g -Wall -o  hello hello.c
root@host2:~/mpidir# mpiexec --allow-run-as-root -n 5 ./hello
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var
/lib/docker/overlay2/l/3DPH4YWCKIPGRYDJKNWH6A4K46:/var/lib/docker/overlay2/l/MW5
E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/v
ar/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUGF:/var/lib/docker/overlay2/l/P
24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/2d66cc40299952640f2e
667abe36e557bd687ccc21fd818f6044ad512eba8ebc/diff,workdir=/var/lib/docker/overla
y2/2d66cc40299952640f2e667abe36e557bd687ccc21fd818f6044ad512eba8ebc/'
I am process 0. I recv string 'Hello World!' from process 1.
I am process 0. I recv string 'Hello World!' from process 2.
I am process 0. I recv string 'Hello World!' from process 3.
I am process 0. I recv string 'Hello World!' from process 4.
root@host2:~/mpidir# exit
```

```
root@host1:~/mpidir#  mpiexec --allow-run-as-root -n 5 ./hello
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var
/lib/docker/overlay2/l/GKTDI4J6TOL2CWT5JG5YM24D24:/var/lib/docker/overlay2/l/MW5
E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/v
ar/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUGF:/var/lib/docker/overlay2/l/P
24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/59d4b3da3f5aa3561681
3de03fbe35e9a145e8598303182003f711a961f506bc/diff,workdir=/var/lib/docker/overla
y2/59d4b3da3f5aa35616813de03fbe35e9a145e8598303182003f711a961f506bc/'
I am process 0. I recv string 'Hello World!' from process 1.
I am process 0. I recv string 'Hello World!' from process 2.
I am process 0. I recv string 'Hello World!' from process 3.
I am process 0. I recv string 'Hello World!' from process 4.
```

在多机并行计算是还需要加上--oversubscribe，防止报错进程数不够

```
root@host1:~/mpidir# mpirun --allow-run-as-root -oversubscribe -np 10 -host host1,host2 hello
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var/lib/docker/overlay2/l/GKTDI4J6TOL2CWT5JG5YM24D24:/var/lib/docker
/overlay2/l/MW5E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/var/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUG
F:/var/lib/docker/overlay2/l/P24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/59d4b3da3f5aa35616813de03fbe35e9a145e8598303182003f711a
961f506bc/diff,workdir=/var/lib/docker/overlay2/59d4b3da3f5aa35616813de03fbe35e9a145e8598303182003f711a961f506bc/'
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var/lib/docker/overlay2/l/3DPH4YWCKIPGRYDJKNWH6A4K46:/var/lib/docker
/overlay2/l/MW5E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/var/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUG
F:/var/lib/docker/overlay2/l/P24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/2d66cc40299952640f2e667abe36e557bd687ccc21fd818f6044ad5
12eba8ebc/diff,workdir=/var/lib/docker/overlay2/2d66cc40299952640f2e667abe36e557bd687ccc21fd818f6044ad512eba8ebc/'
I am process 0. I recv string 'Hello World!' from process 1.
I am process 0. I recv string 'Hello World!' from process 2.
I am process 0. I recv string 'Hello World!' from process 3.
I am process 0. I recv string 'Hello World!' from process 4.
I am process 0. I recv string 'Hello World!' from process 5.
I am process 0. I recv string 'Hello World!' from process 6.
I am process 0. I recv string 'Hello World!' from process 7.
I am process 0. I recv string 'Hello World!' from process 8.
I am process 0. I recv string 'Hello World!' from process 9.
```

二、改写代码

2-1单机运行

在本机编译修改后的代码integral.c，计算定积分

```
wangsky@wangsky-virtual-machine:~/mpich-3.3.2/examples$ mpicc -o integral integr
al.c
```

运行：

```
wangsky@wangsky-virtual-machine:~/mpich-3.3.2/examples$ mpiexec -n 5 ./integral
The integal of x*x in region [0,10] =333.333333333341216
wangsky@wangsky-virtual-machine:~/mpich-3.3.2/examples$
```

结果正确

在本机编译修改后的sqrtsum.c，计算开方和

```
wangsky@wangsky-virtual-machine:~/mpich-3.3.2/examples$ mpicc -o sqrtsum sqrtsum
.c
/tmp/ccjyYg4Y.o：在函数‘main’中：
sqrtsum.c:(.text+0x100)：对‘sqrt’未定义的引用
collect2: error: ld returned 1 exit status
```

需要手动加库，在末尾加一个-lm参数，再编译运行如下

```
wangsky@wangsky-virtual-machine:~/mpich-3.3.2/examples$ mpicc -o sqrtsum sqrtsum
.c -lm
wangsky@wangsky-virtual-machine:~/mpich-3.3.2/examples$ mpiexec -n 5 ./sqrtsum
The total SqrtSum=21129.094472.
```

两个文件源码，在目录下，这里附上：

```c
//计算大数组开方和
#include <stdio.h>
#include <mpi.h>
#include <math.h>
#define N 1002 //N采用固定的全局参数

int main(int argc, char** argv) {
    int myid, P, source, numprocs, C=0;
    double  data[N], SqrtSum=0.0;
    for (int i=0;i<N;i++)
        data[i]=i;
    MPI_Status status;
    char message[100];
    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &myid);
    MPI_Comm_size(MPI_COMM_WORLD,&numprocs);
    - -numprocs;
    double sum=0,inte;
    for (int k=0;k<N;k++)
        if(myid!=0){
            if(k%(numprocs-1)==myid-1)
                sum+=sqrt(k);
        }
    MPI_Reduce(&sum,&inte,1,MPI_DOUBLE,MPI_SUM, 0, MPI_COMM_WORLD);
    if(myid==0){
```

```
        printf("The total SqrtSum=%f.\n",inte);
    }
    MPI_Finalize();
    }
```

```
//计算定积分
#define N 100000000
#define  a 0
#define  b 10
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include "mpi.h"
int main(int argc, char** argv)
{
    int myid,numprocs;
    int i;
    double local=0.0, dx=(double)(b-a)/N;
    double inte, x;
    MPI_Status status;
    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &myid);
    MPI_Comm_size(MPI_COMM_WORLD,&numprocs);
    for(i=myid;i<N;i=i+numprocs) {
        x = a + i*dx +dx/2;
        local +=x*x*dx;
    }
    double p=local;
    if(myid!=0){
        MPI_Send(&local,1,MPI_DOUBLE,0,1,MPI_COMM_WORLD);
    }
    else if(myid==0)
    {
        for(int source=1;source<numprocs;source++)
        {
        double local1;
        MPI_Recv(&local1,1,MPI_DOUBLE,source,1,MPI_COMM_WORLD,&status);
        p=p+local1;
        }
        printf("The integal of x*x in region [%d,%d] =%16.15f\n", a, b, p);
    }
    MPI_Finalize();
}
```

2-2在模拟集群上运行程序

将修改后的代码放入挂载的文件夹中

分别在host1和host2编译修改后的代码

```
hello  hello.c  integral.c  sqrtsum.c
root@host1:~/mpidir# mpicc -o sqrtsum sqrtsum.c -lm
root@host1:~/mpidir# mpicc -o integral integral.c
root@host1:~/mpidir#
```

```
root@wangsky-virtual-machine:/home/wangsky# docker exec -it mpi-host2 bash
root@host2:/# cd /root/mpidir
root@host2:~/mpidir#  mpicc -o sqrtsum sqrtsum.c -lm
root@host2:~/mpidir# mpicc -o integral integral.c
```

在host2上运行多机并行程序

```
--------------------------------------------------------------------------
root@host2:~/mpidir# mpiexec --allow-run-as-root -oversubscribe -np 8 -host host
1,host2 sqrtsum
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var
/lib/docker/overlay2/l/3DPH4YWCKIPGRYDJKNWH6A4K46:/var/lib/docker/overlay2/l/MW5
E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/v
ar/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUGF:/var/lib/docker/overlay2/l/P
24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/2d66cc40299952640f2e
667abe36e557bd687ccc21fd818f6044ad512eba8ebc/diff,workdir=/var/lib/docker/overla
y2/2d66cc40299952640f2e667abe36e557bd687ccc21fd818f6044ad512eba8ebc/'
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var
/lib/docker/overlay2/l/GKTDI4J6TOL2CWT5JG5YM24D24:/var/lib/docker/overlay2/l/MW5
E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/v
ar/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUGF:/var/lib/docker/overlay2/l/P
24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/59d4b3da3f5aa3561681
3de03fbe35e9a145e8598303182003f711a961f506bc/diff,workdir=/var/lib/docker/overla
y2/59d4b3da3f5aa35616813de03fbe35e9a145e8598303182003f711a961f506bc/'
The total SqrtSum=21129.094472.
root@host2:~/mpidir#
```

```
root@host2:~/mpidir# mpiexec --allow-run-as-root -oversubscribe -np 8 -host host
1,host2 integral
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var
/lib/docker/overlay2/l/3DPH4YWCKIPGRYDJKNWH6A4K46:/var/lib/docker/overlay2/l/MW5
E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/v
ar/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUGF:/var/lib/docker/overlay2/l/P
24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/2d66cc40299952640f2e
667abe36e557bd687ccc21fd818f6044ad512eba8ebc/diff,workdir=/var/lib/docker/overla
y2/2d66cc40299952640f2e667abe36e557bd687ccc21fd818f6044ad512eba8ebc/'
Unexpected end of /proc/mounts line `overlay / overlay rw,relatime,lowerdir=/var
/lib/docker/overlay2/l/GKTDI4J6TOL2CWT5JG5YM24D24:/var/lib/docker/overlay2/l/MW5
E3YXPVVYZ4LPGTHRUBV5VKU:/var/lib/docker/overlay2/l/VH2XEYBVPKIY2RIZ3SVXN7LLT6:/v
ar/lib/docker/overlay2/l/IUCVUA3Q5MWZFNFUY67DMJDUGF:/var/lib/docker/overlay2/l/P
24EWM5WFWEV55H6JU5TMFUKU5,upperdir=/var/lib/docker/overlay2/59d4b3da3f5aa3561681
3de03fbe35e9a145e8598303182003f711a961f506bc/diff,workdir=/var/lib/docker/overla
y2/59d4b3da3f5aa35616813de03fbe35e9a145e8598303182003f711a961f506bc/'
The integal of x*x in region [0,10] =333.333333333345479
```

结果正常输出

2-3 探究进程数与运行时间的关系

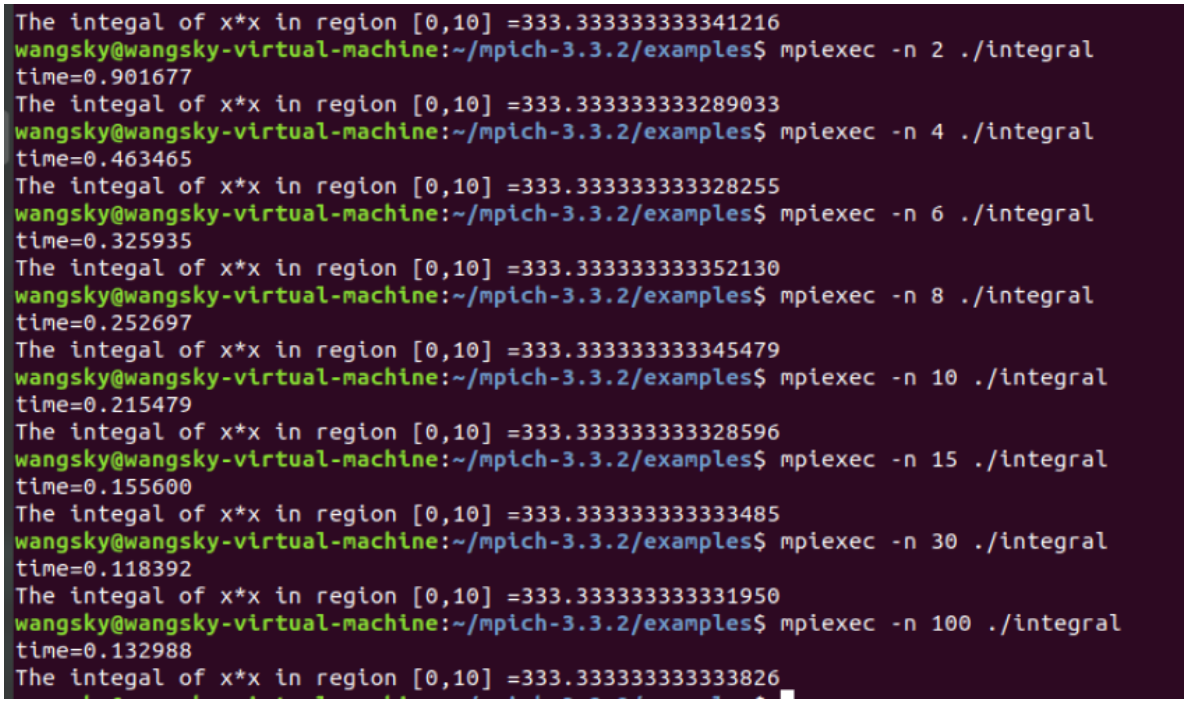
这里选用integral.c程序，即计算定积分的程序

首先**在程序里加上输出运行时间的代码**，命名为integral_time.c

```
#define N 100000000
#define  a 0
#define  b 10
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include "mpi.h"
```

```
int main(int argc, char** argv)
{
    int myid,numprocs;
    int i;
    double time;
    clock_t start,finish;
    start=clock();
    double local=0.0, dx=(double)(b-a)/N;
    double inte, x;
    MPI_Status status;
    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &myid);
    MPI_Comm_size(MPI_COMM_WORLD,&numprocs);
    for(i=myid;i<N;i=i+numprocs) {
        x = a + i*dx +dx/2;
        local +=x*x*dx;
    }
    double p=local;
    if(myid!=0){
        MPI_Send(&local,1,MPI_DOUBLE,0,1,MPI_COMM_WORLD);
    }
    else if(myid==0)
    {
        for(int source=1;source<numprocs;source++)
        {
    double local1;
    MPI_Recv(&local1,1,MPI_DOUBLE,source,1,MPI_COMM_WORLD,&status);
        p=p+local1;
        }
    finish=clock();
    time=(double)(finish-start)/CLOCKS_PER_SEC;//计算运行时间
    printf("time=%lf\n",time);//输出运行时间
        printf("The integal of x*x in region [%d,%d] =%16.15f\n", a, b, p);
    }
    MPI_Finalize();
}
```

在单机上按不同进程数运行程序：



我们可以看到，在进程数由2到30的过程中，**运行速度都在加快，但是加速度在逐渐下降。**

进程数从2到4，时间从0.90变为0.46，减少了将近一半

但是从15到30，时间的变化幅度已经很小，这一点我们可以在下表关于进程数与运行时间的表格中看到

进程数	运行时间	运行加速度
2	0.901677	*
4	0.463465	0.219
6	0.325935	0.069
8	0.252697	0.037
10	0.215479	0.019
15	0.155600	0.012
30	0.118392	0.0025

而在进程数达到100时发现，运行时间变为0.133，**反倒比进程数为30时还慢**

这说明，**进程数也不是越多越好，因为过多的进程数会加大MapReduce的通信成本，反而是程序运行速度下降**