



Лаборатория
Информационных
Технологий

ВВЕДЕНИЕ В ПРАКТИКУ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

А. Айриян, Д. Беляков, Я. Буша, Т. Заикина, М. Матвеев

ЛАБОРАТОРИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ОИЯИ

XXIV Летняя школа молодых ученых

и специалистов ОИЯИ

Липня, 24-26.07.2020

ВЫЧИСЛЕНИЕ ЧИСЛА π

Объем шара ограниченного сферой

$$S: (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = R^2$$

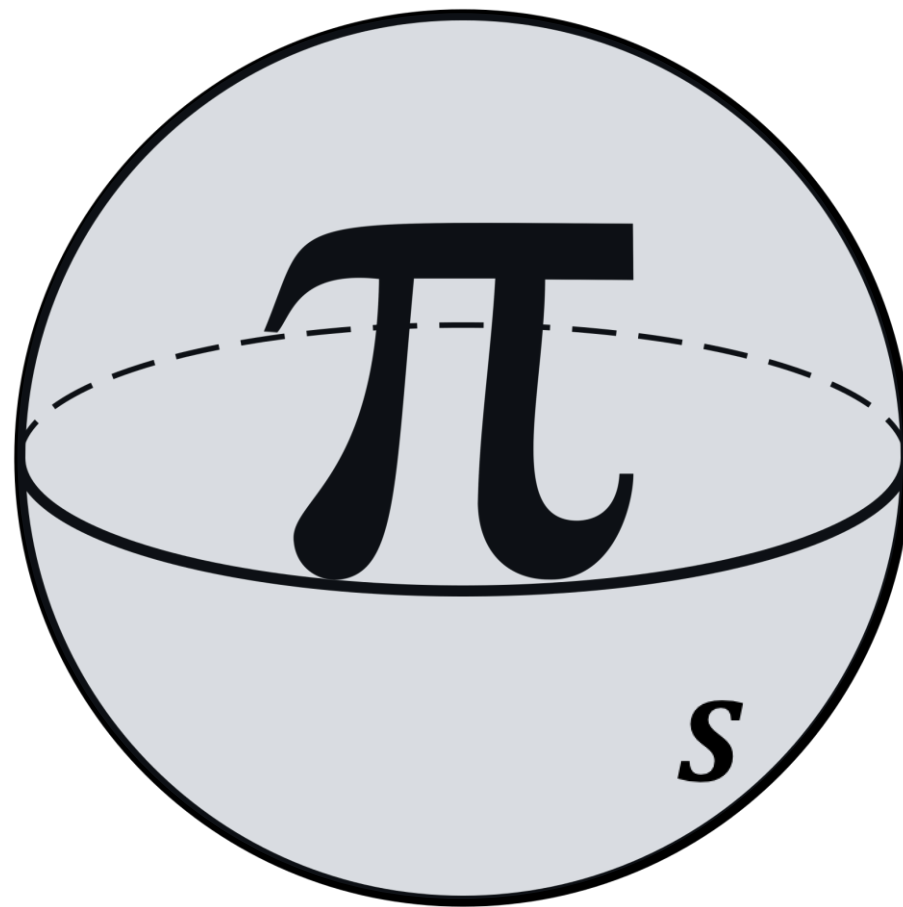
вычисляется по формуле:

$$V = \frac{4}{3} \pi r^3.$$

Зная объем шара можно вычислить

число π :

$$\pi = \frac{3V}{4r^3}$$



ЧИСЛЕННОЕ ВЫЧИСЛЕНИЕ ОБЪЕМА ШАРА

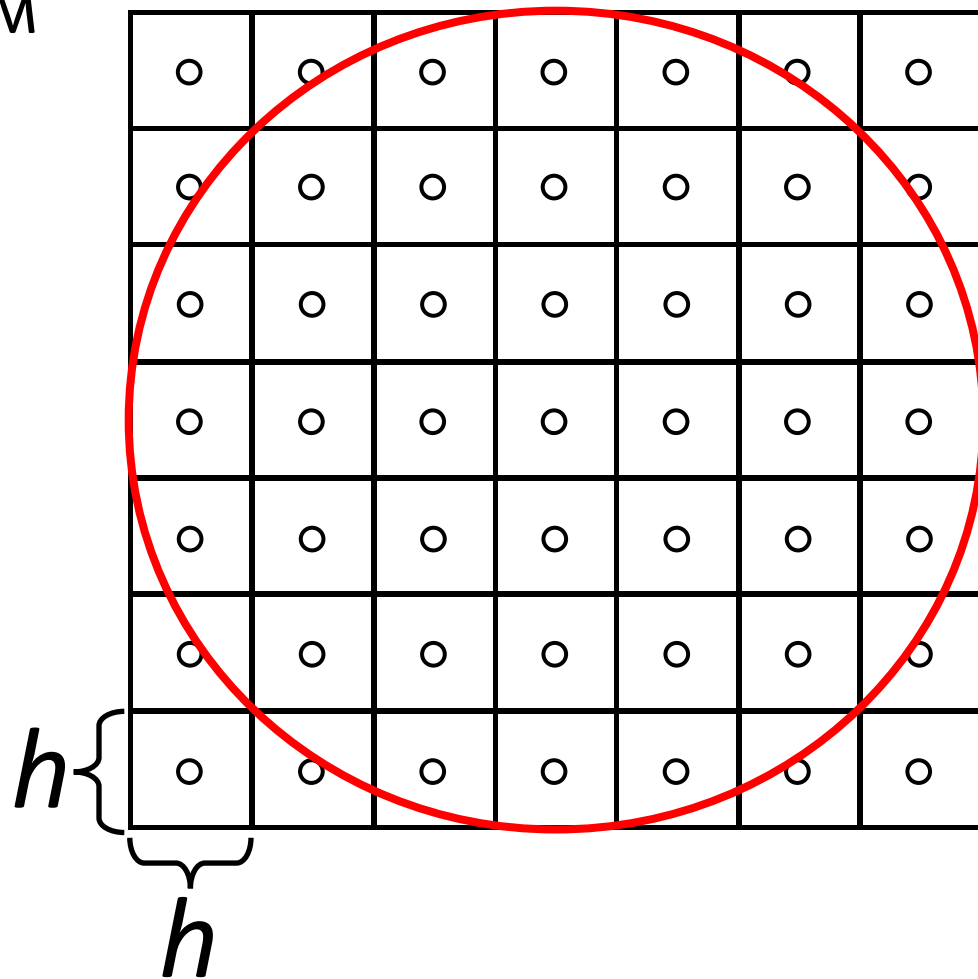
Вписываем сферу в куб и дискретизируем область ограниченную кубом.

Вычисляем координаты центров ячеек:

$$x_i = x_{min} + h(i + 0.5)$$

$$y_j = y_{min} + h(j + 0.5)$$

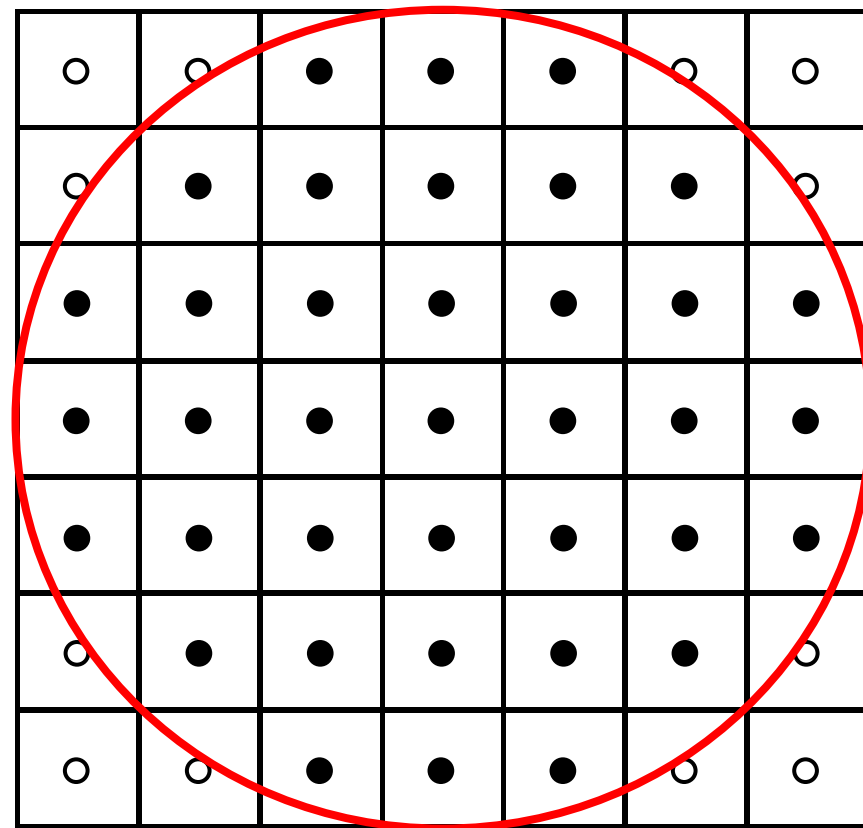
$$z_k = z_{min} + h(k + 0.5)$$



ЧИСЛЕННОЕ ВЫЧИСЛЕНИЕ ОБЪЕМА ШАРА

Если центр ячейки попал в сферу,
то присваиваем ему единичное значение,
иначе нулевое:

$$v(x_i, y_j, z_k) = \begin{cases} 1 & \text{if } \sqrt{x_i^2 + y_j^2 + z_k^2} \leq 1 \\ 0 & \text{if } \sqrt{x_i^2 + y_j^2 + z_k^2} > 1 \end{cases}$$



ЧИСЛЕННОЕ ВЫЧИСЛЕНИЕ ОБЪЕМА ШАРА

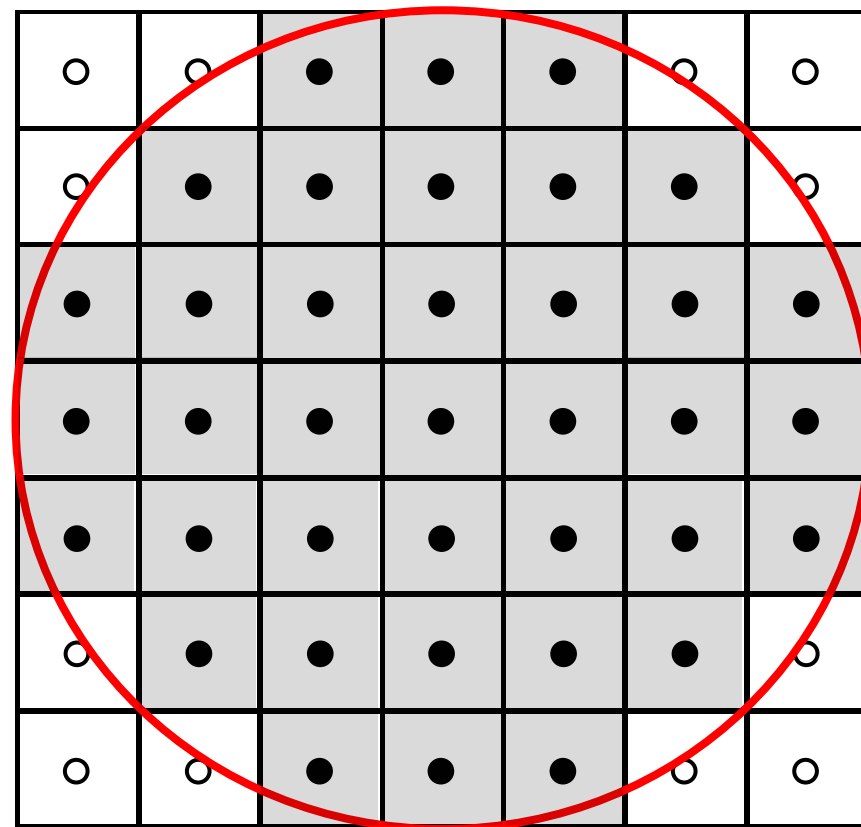
Суммируем элементарные объемы h^3 :

$$V \approx h^3 \sum_{k=0}^{n-1} v_{i,j,k},$$

где n кол-во точек дискретизации.

В итоге получаем оценку числа π :

$$\pi \approx \frac{3}{4} V$$



GETTING STARTED

Network: ASUS

Password: HybriLIT

ssh zabavnyXX@192.168.1.50

Password: tut2020

```
cp /zfs/hybrilit.jinr.ru/projects/lipnya-2020.zip .
```

```
unzip lipnya-2020.zip
```

```
cd lipnya-2020
```

```
module avail, module list,
```

```
module add xx, module rm xx
```



ПОСЛЕДОВАТЕЛЬНЫЙ КОД НА C/C++

```
cd cpp
```

```
make all
```

```
sbatch pi_cpp.sh
```

```
squeue
```

```
cat slurm-XX.out
```





OPEN MPI

MESSAGE PASSING INTERFACE (MPI)

Фактический стандарт для организации параллельных вычислений на системах с распределенной памятью.

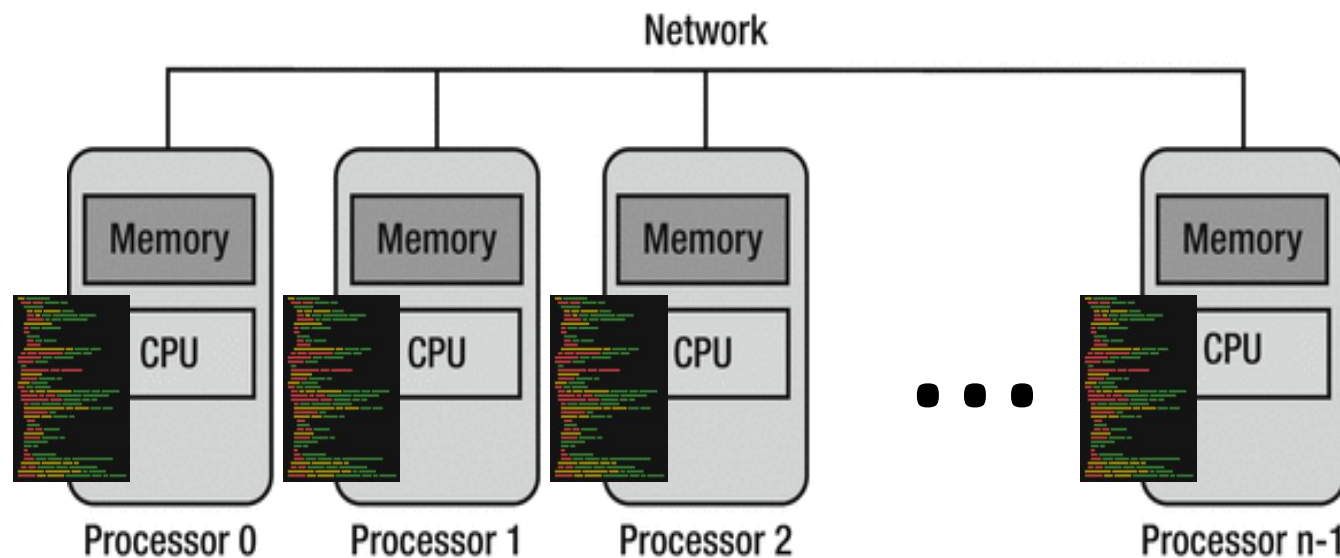
```
cd mpi
```

```
module add openmpi/v4.0.4
```

```
make all
```

```
sbatch pi_mpi.sh
```

```
cat slurm-XX.out
```



OPEN MULTI-PROCESSING (OPENMP)

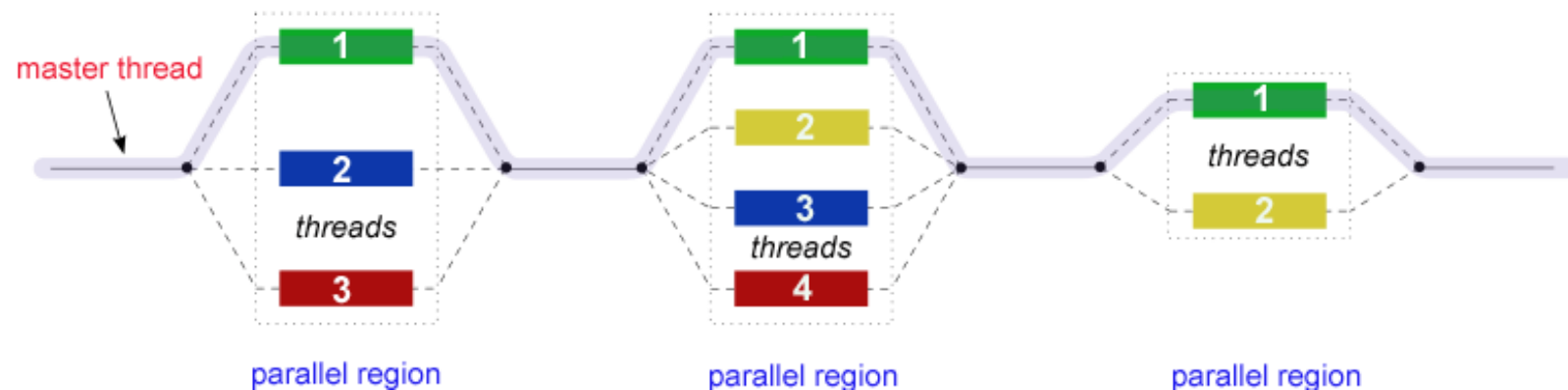
Набор директив компилятора для многопоточных приложений на вычислительных системах с разделяемой (общей) памятью.

```
cd ../omp
```

```
make all
```

```
sbatch pi_omp.sh
```

```
cat slurm-XX.out
```



OPEN COMPUTING LANGUAGE (OPENCL)

Фреймворк параллельных вычислений на различных вычислительных системах с разделяемой (общей) памятью, таких, например, как ГПУ, ЦПУ, и FPGA.

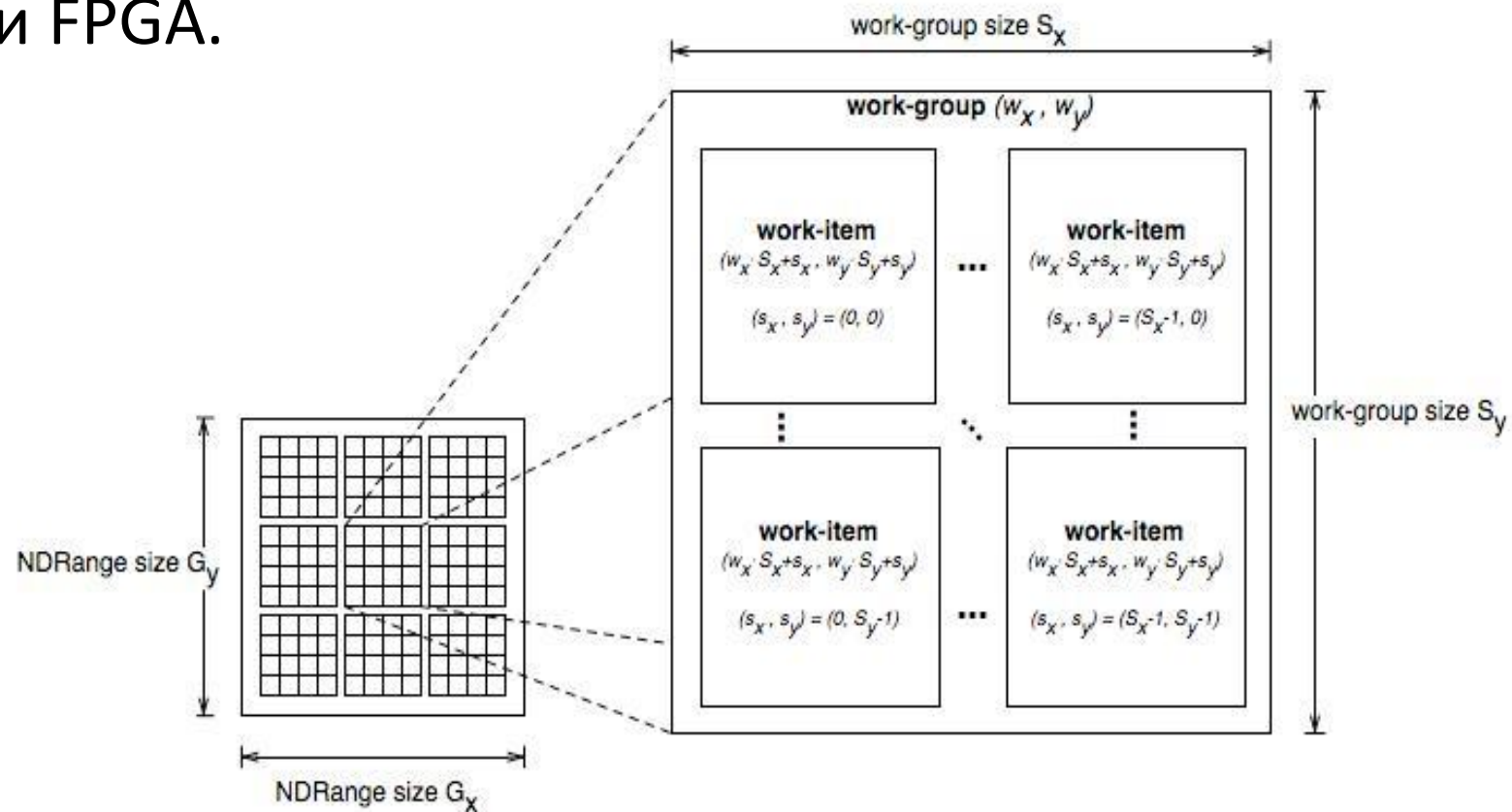
```
cd ../ocl
```

```
module add cuda/v11.0
```

```
make all
```

```
sbatch pi_ocl.sh
```

```
cat slurm-XX.out
```



COMPUTE UNIFIED DEVICE ARCHITECTURE (CUDA)

Программно-аппаратная архитектура параллельных вычислений на графических процессорах Nvidia.

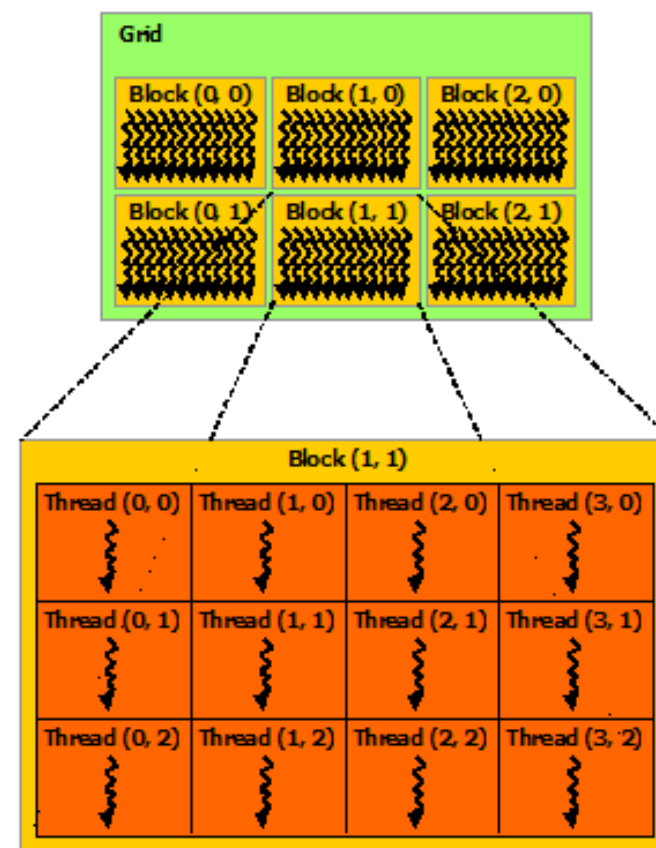
```
cd ../cuda
```

```
module add cuda/v11.0
```

```
make all
```

```
sbatch pi_cuda.sh
```

```
cat slurm-XX.out
```



НА ЭТОМ ВСЕ! МОЖНО РАССЛАБИТЬСЯ.



СПАСИБО ЗА
ВНИМАНИЕ!