Weekly Report

LIU Honghao

Feb 28 2021

1 DFT

1.1 1D-DFT

Forward discrete fourier transform

$$F(u) = \sum_{x=0}^{M-1} f(x) \exp\left[-2\pi i \frac{xu}{M}\right] \tag{1}$$

Inverse discrete fourier transform

$$f(x) = \frac{1}{M} \sum_{u=0}^{M-1} f(x) \exp\left[2\pi i \frac{xu}{M}\right] \tag{2}$$

1.2 2D-DFT

Forward discrete fourier transform

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \exp\left[-2\pi i \left(\frac{xu}{M} + \frac{yv}{N}\right)\right] \tag{3}$$

Inverse discrete fourier transform

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) \exp\left[2\pi i \left(\frac{xu}{M} + \frac{yv}{N}\right)\right] \tag{4}$$

Both one dimensional DFT and two dimensional DFT have matrix-vector multiplication form. The 2D-DFT could be calculated as two 1D-DFT and rewritten as Matrix-Matrix multiplication, when the u and v are integrals.

However, in radio astronomical application, the values of them are decimals. In the matrix-vector form, the column number is extremely large which indicated the number of u and v.

2 Matrix Vector Multiplication

2.1 Simple case

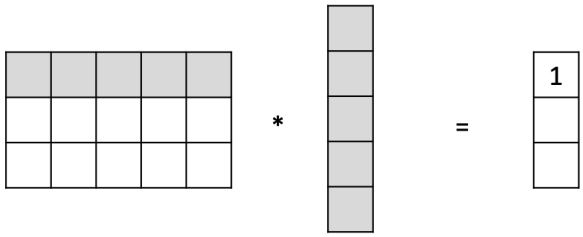

Figure 1: Simple case

Each element in output corresponding to a thread handles the multiplication of one row in matrix and the vector. Nonetheless, the workload is relatively heavy for each thread due to the large size of vector.

2.2 Tiled case with shared memory

Since the vector will participate into the calculation multiple times, loading a tile of vector into shared memory can access this tile faster.

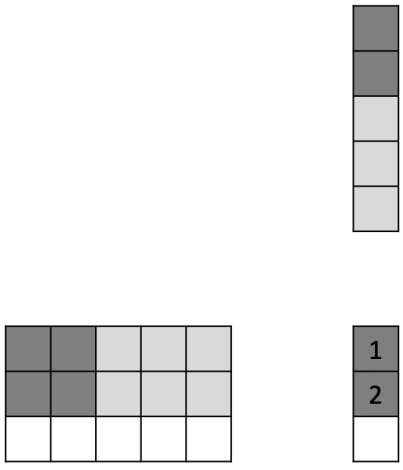

Figure 2: Tiled case

Each tile will load a tile of elements in vector in to shared memory and calculate the values after multiplication.
In Fig2, 1 and 2 are a tile corresponding to a block. They will load the vector piece by piece and add the calculated value in some iteration to get the results.
However, the time of iterations is still large due to the vector is large.

2.3 Atomic Add

In order to get a value of output, several threads will participate in.

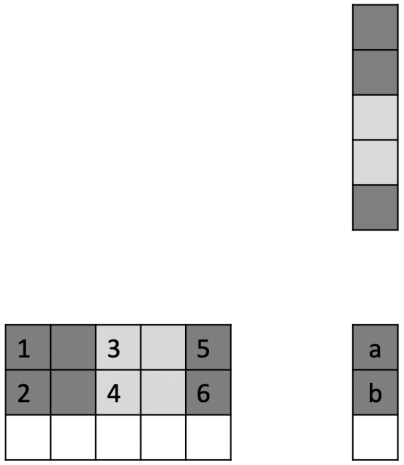

Figure 3: MVM with atomic addition

In this example, 1 and 2 are two threads in one block in order to utilize the shared memory. Threads 1, 3 and 5 will be involved to calculate the value of a and the calculation needs atomic operation.