User Manual for Running KMC1D to Calculate Main Effect Kinship Matrix

1. KMC1D Overview

A typical GWAS study may deal with several millions of SNP markers. The first step of GWAS analysis is computing a marker assisted kinship matrix, which describes the relationships among individuals. Conventional kinship matrix are usually used for 1D main effect GWAS analysis, and its calculation mathematically has a complexity of $O(mn^2)$, where m is the marker number and n is the number of individuals. When the sample and the number of markers are very large, calculation of kinship matrix presents a huge computation burden. Recently, the GPU-empowered parallel system has become a standard for HPC platform. Kinship matrix calculation involves pure matrix operations and thus is suitable for GPU-employed parallel structure.

We thoroughly investigated the principles of kinship matrices and proposed a method to divide the high-dimensional markers into successive blocks. We then calculate the kinship matrix for each block and merge together the block-wise kinship matrices to form the genome-wide kinship matrix. The matrix operations thus can be parallelized by GPU kernels. Based on these, we developed a GPU-empowered pipeline: KMC1D for main effect kinship matrix calculation. The user friendly web version can be accessed at http://bioinfo.noble.org/KMC1D/. However, the user also can download the CUDA source code from the GitHub, then compile and run it using Command Line. KMC1D can be configured with the different option parameters to run at the mode of GPU parallel, CPU Serial or Read Loop Through. The following sections will describe the technical parts about the application of running KMC1D to calculate your main effect kinship matrix.

2. Running KMC1D to Calculate Main Effect Kinship Matrix

Essentially, KMC1D only can run on a CUDA-capable GPU Linux Server. You can run it at our GPU sever, or download the source code, compile and run it at your CUDA-capable GPU server. Once you download the KMC2D repository, please unzip the two zipped files: Run_Test.zip and Simulated_Data.zip.

2.1 Application of KMC1D using our user-friendly web portfolio.

If you do not have CUDA-capable GPU Linux, no problem, you still can use KMC1D through our developed web portfolio at https://bioinfo.noble.org/KMC1D/ .

2.2 Application of KMC1D using bash shell script

If you know well the Linux commands and bash script, and also have a CUDA-capable computation environment, but are not familiar with CUDA-NVIDIA Environment and want to tentatively test your GPU hardware and CUDA software environment, and simply run KMC1D in your local Linux server. You can get the KMC1D repository from

https://bioinfo.noble.org/KMC1D/ or GitHub at https://github.com/noble-research-institute/KMC1D. At the KMC1D repository, unzip the **Run_Test.zip** and **Simulated_Data.zip** at first, and then go to sub-directory **Run_Test**, and use the script run_test.sh and the following instruction to calculate the epistatic kinship matrix.

```
$ chmod a+x ./run_test.sh

$ chmod a+x ./KMC1D

$ ./run_test.sh ../Simulated_Data/Gen_Data_16190_1390.CSV ../Simulated_Data/Result/kinship_matrix.csv
```

2.3 Application of KMC2D using Command Line

If you know well the Linux command, and also are very familiar with CUDA GPU programming, you want to learn the KMC1D's source code, you can refer to INSTALL.md to set your CUDA-capable environment at first and then go to the sub-directory **Source_Code**, and use make command to compile the KMC1D.cu into an execute KMC1D.

2.3.1 Display the help usage message

```
$ ./KMC1D –h
```

```
Welcome to use this program to do Kinship Matrix Calculation
The usage of input parameter arguments are listed as followings:
-h or -H: Output this Help usage message
-g or -G: The full name of Genotype file
-k or -K: The full name of Kinship Matrix file
-i or -I: The Individual number
-b or -B: The Block size for dividing the Genotype markers
-m or -M: Run_Mode(0, 1, 2 for the GPU/CPU/Read Through)
```

2.3.2 Run KMC1D at the mode of GPU_Parallel (-m 0)

```
./KMC1D -g ../Simulated_Data/Gen_Data_16190_1390.CSV -k ../Simulated_Data/Result/KM1D_GPU.txt
-i 1390 -b 2000 -m 0

Hello, the elapse time is 8.4 seconds
```

2.3.3 Run KMC1D at the mode of CPU_Serial (-m 1)

```
$ ./KMC1D -g ../Simulated_Data/Gen_Data_16190_1390.CSV -k
../Simulated_Data/Result/KM1D_CPU.txt -i 1390 -b 2000 -m 1
Hello, the elapse time is 101 seconds
```

2.3.4 Run KMC1D at the mode of Read_Loop_Through (-m 2)

```
$ ./KMC1D -g ../Simulated_Data/Gen_Data_16190_1390.CSV -k
../Simulated_Data/Result/KM1D_None.txt -i 1390 -b 2000 -m 2

Hello, the elapse time is 4.43 seconds
```