Appendix 1. How to determine $FLOPS$ and $time_{schedule}$ in the evaluation of backend cost

Both CPU and GPU use $FLOPS$ to measure the capability of the processors. Only GPU has the $time_{schedule}$ term. Their values are determined as follows.

- $FLOPS$. For CPU, if the OS is Linux or Android, we can access the maximal frequency of each CPU core. Then choose the largest $k$ frequencies and add them together as the $FLOPS$ term, where $k$ is the pre-specified number of threads (such as two threads or four threads). For the other CPU systems, set $FLOPS = 2 \cdot 10^9$. For GPU, we estimate the $FLOPS$ through practical running. Specifically, we run the MobileNet-v1 network for 100 times and obtain the $FLOPS$ values for a bunch of common mobile GPUs. The results are shown in the list below. For those GPUs not in this list, we set the $FLOPS$ as $4 \times 10^9$, namely, faster than CPU ($FLOPS = 2 \cdot 10^9$), as is the normal case.

The list of GPU $FLOPS$ ($10^9$): Mali-T860: 6.83; Mali-T880: 6.83; Mali-G51: 6.83; Mali-G52: 6.83; Mali-G71: 31.61; Mali-G72: 31.61; Mali-G76: 31.61; Adreno (TM) 505: 3.19; Adreno (TM) 506: 4.74; Adreno (TM) 512: 14.23; Adreno (TM) 530: 25.40; Adreno (TM) 540: 42.74; Adreno (TM) 615: 16.77; Adreno (TM) 616: 18.77; Adreno (TM) 618: 18.77; Adreno (TM) 630: 42.74; Adreno (TM) 640: 42.74.

- $time_{schedule}$. This value depends on the adopted graphical API. For OpenCL and OpenGL, it is empirically set to 0.05 (ms), which is the normal average time of calling API like `clEnqueueNDRangeKernel`. For Vulkan, since it only needs to summit `commandBuffer`, which is less time-consuming, thus $time_{schedule}$ can be estimated as 0.01 (ms).