Parallel strategy for Optimal Power Efficiency

Zhoue He

Linyi University, Shandong, China

Keywords: Graphics Processing Units, Parallel Strategy, Power Efficiency, Power Model

Abstract. Large-scale data streams processing is import to data processing application. So we need to investigate the parallel strategy for the Large-scale data streams processing. Here we propose two parallel strategies to handle data streams in real time, and consider the power efficiency as an important factor to the parallel strategies. We present a method to quantify the power efficiency for data streams during the computing. Finally, we compare the two parallel strategies on a large quantity of real stream data. The experiments prove the accuracy of analysis on power efficiency.

Introduction

Large-scale data streams processing is import to data processing application. So we need to investigate the parallel strategy for the Large-scale data streams processing. The existing research focus on parallelism for improving the data stream processing. As Graphics Processing Units are becoming powerful, researcher have begun to handle the Large-scale data streams on Graphics Processing Units.

But there has been little study to investigate the issue of Large-scale data streams computing from the perspective of task. The Large-scale data streams processes have the task feature that is the communication between the CPUs and the computation on GPUs. Hence, we look into the level of task model and then investigate its impact on computation. We propose two parallel strategies to at the task level Large-scale data streams processing. The Large-scale data streams processing consumes much energy because it copes with Large-scale data in real time. To address this issue, we analyze the power efficiency of the two parallel strategies.

MODELS

Tasks Model

Given a task sequence S with n tasks, and each task T has partial ordering relation. S = $\{T_1,T_2,...,T_n\}$, $<T_i,T_j>$ ($i \neq j$). Consider the case when the n tasks execute sequentially on the GPUs and each task has the same size, $|T_i| = |T_j| (i \neq j)$. The task is highly intensive computing task that is work well on data-parallel systems and has different kinds of subtasks denoted as R, namely $T_i = \{ R_1,R_2,...,R_k\}$, $1< i < n$, $|T_i| = \sum_{j=1}^{k} |R_j|$.

Program Model

The GPU can be programmed in C language, such as the Compute Unified Device Architecture (CUDA) or OpenCL of the NVIDIA [7]. The abstractions are kernels, thread blocks, threads, and thread grids. So there are two program models. We may use a kernel to complete the tasks. The thread blocks deal with different subtasks. Then the program just has one kernel, P = {K}. The other program model deals with the different subtasks, $P = \{K_1,K_2,...,K_m\}$. Assume the program P to compute the task T_i, and T_i has m subtasks R_1, R_2,...,R_m. The kernel K_i is to compute the subtasks R_i.

Power Model

The power consumption model estimates the energy efficiency of computing for the task model from the perspective of software. Then the GPU power consumption is dominated by P_d, which is given by $P_d = C_{ef} \times V_{dd}^2 \times f$, where C_{ef} is the effective switch capacitance, V_{dd} is the supply voltage and f is the processor clock frequency. The energy consumed can be given as

$$E(t) = \int_{t1}^{t2} P_d(t)dt \quad (1)$$

P(t) is difficult to obtain due to the variation of this function. We use average power $\overline{P}$ to substitute P(t) and estimate the energy consumed. So E is a sum of product of average power , and T obtained by the performance counter, namely $E = \overline{P} \times T$ [9].

Parallel Strategies

Multi-Stage parallel strategy

The main idea of Multi-stage parallel strategy is to divide the task into multiple stages. As shown in Fig.1

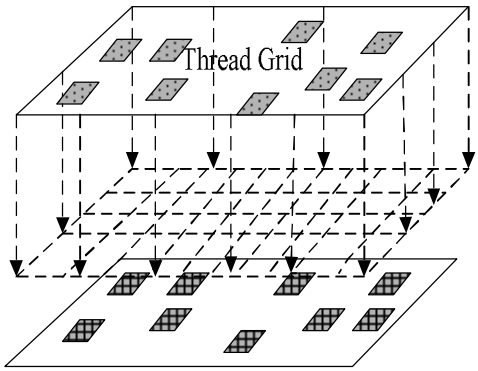

Figure 1 Multi-Stage Parallel Strategy

The rectangle on the top represents the thread blocks set, and the bottom one represents the task to be handled by the thread blocks.

Multi-Pass Parallel Strategy

The same subtasks map to the same kernel and the whole task is mapped to multiple kernels. The mainly idea of MPPS is shown in Fig. 2.

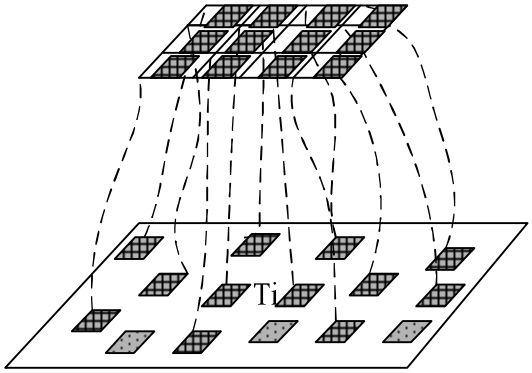

Figure 2 Multi-Pass Parallel Strategy

The rectangle on the top represents a kernel and the bottom one are subtasks.

Power Consumption Analysis

Multi-stage parallel strategy is a strategy that lauching threads aims at the whole task rather than a subtasks set. The energy consumption of the two parallel strategies are as follows.

$$E^1(N)=P^1_s\times t^1_s + P^1_c\times t^1_c \quad (2)$$

$$E^2(N)=P^2_s\times t^2_s + P^2_c\times t^2_c \quad (3)$$

Where P_s and P_c are the power of data tranfering and kernel executing. And t denotes the time. The following equation assumes that $P^1_s \approx P^2_s$ and $t^1_c \approx t^2_c$.

$$\Delta E(N) = E^2 - E^1 = \Delta E_s + \Delta E_c$$
$$\Delta E_s = P_s(t^2_s - t^1_s), \Delta E_c = (P^2_c - P^1_c)t_c \quad (4)$$

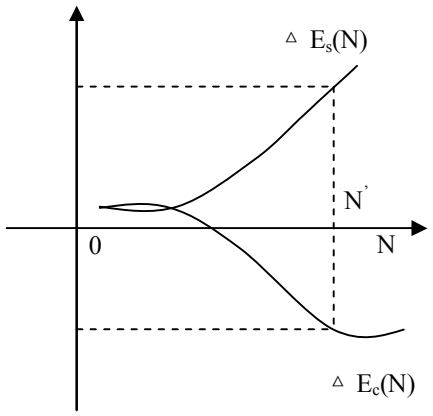

Figure 3 Threshold of N in Equation 4

Simulation Experiment

To compare the two parallel strategies, we conduct simulation experiments based on the task model. In our experiments, assuming the task contains 3 subtasks and each subtask has the same intensity. We investigate the impact of intensity to the two parallel strategies. While the intensity is less than 1.09, multi-stage parallel strategy is prior to multi-pass parallel strategy. But when the intensity is greater than 1.09, the delta of S^e is increasing. When the intensity is less than 1.09, the subtask is the memory-intensive task. multi-stage parallel strategy obtains better communication performance than multi-pass parallel strategy. Hence the performance power is better. The subtasks will spend more time on the communication with the arithmetic intensity increasing. multi-pass parallel strategy achieves better performance power because this strategy launches less active SMs than multi-stage parallel strategy and reduces the energy consumed.

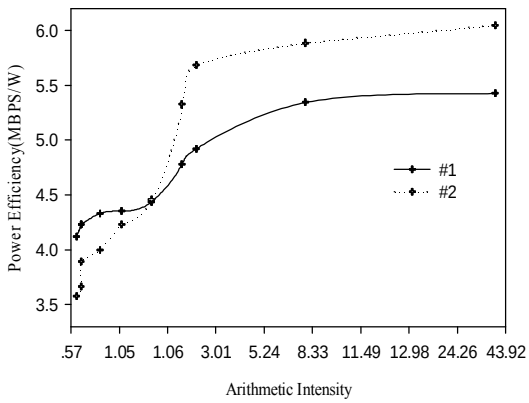

Figure 4 Comparison of the Parallel strategies in Power Efficiency

References

[1]G. Ghinea, G.-M. Muntean, P. Frossard, M. Etoh, F. Speranza, and H. Wu. Special issue on quality issues on mobile multimedia broadcasting[J]. IEEE Trans. Broadcast., (2008) vol.54, no, 3, pp.424-727, Sep..

[2]G. Vasiliadis, S. Antonatos, M. Polychronakis, E.P. Markatos, and S. Ioannidis, "Gnort: High Performance Network Intrusion Detection Using Graphics Processors", in Proc. RAID, (2008), pp.116-134.

[3]Giorgos Vasiliadis and Sotiris Ioannidis. GrAVity: a massively parallel antivirus engine. Recent Advances in Intrusion Detection,79-96. Springer. (2010).

[4]Naga Govindaraju, Jim Gray, Ritesh and Dinesh Manocha. GPUTeraSort: High Performance Graphics Coprocessor Sorting for Large Database Management. ACM SIGMOD (2006).

[5]Patrick Kurp, Green Computing, Commons. Of the Association for Computing Machinery, (2008),51(10):11-13.

[6]Pharr M, Fernando R. GPU Gems2. Boston: Addison Wesley, (2005),493-495.

[7] R. Ge, X. Feng, S. Song, H. Chang, D.Li, K.Cameron, PowerPack: energy profiling and analysis of high-performance systems and applications. IEEE Transactions on Parallel and Distributed Systems,(2010), Vol. 21, No.5, pp. 658-671.

[8]Y.Jiao, H. Lin, P. Balarji.et al. Power and Performance Characterization of Computational Kernel on the GPU[C]. IEEE/ACM Int'l Conference on Green Computing and Communications& Int'l Conference on Cyber, Physical and Social Computing.(2010), pp:221-228,

[9]Sunpyo Hong, Hyesoon Kim, "An Integrated GPU Power and Performance Model", In Proceeding of the 37 th annual international Symposium on Computer Architecture, (2010),pp.280~289.

[10]M.Z.Shaikh, M.Gregoire, W.Li, M. Wroblewski, S.Simon, "In situ Power Analysis of General Purpose Graphical Processing Unit", In 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing. (2011).

[11]Moorthy,A.K. Seshadrinathan,K. et al. Wireless Video Quality Assessment: A Study of Subjective Scores and Objective Algorithms[J] IEEE Transaction on Circuits and Systems for Video Technology. (2010) Vol.20(4),pp:587-599.

[12]Paul,Manoranjan; Weisi Lin, Chiew Tong Lau, Bu-sung Lee. Direct Intermode Selection for H.264 Video Coding Using Phase Correlation. [J] IEEE Transaction on Image Processing. (2011), Vol.20(2), pp:461-473