INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING, *SERIES B*
Volume 3, Number 2, Pages 185–191

SPARSE MATRIX-VECTOR MULTIPLICATION ON NVIDIA GPU

HUI LIU, SONG YU, ZHANGXIN CHEN, BEN HSIEH AND LEI SHAO

Abstract. In this paper, we present our work on developing a new matrix format and a new sparse matrix-vector multiplication algorithm. The matrix format is HEC, which is a hybrid format. This matrix format is efficient for sparse matrix-vector multiplication and is friendly to preconditioner. Numerical experiments show that our sparse matrix-vector multiplication algorithm is efficient on GPU.

Key words. sparse matrix-vector multiplication, GPU, HEC, parallel algorithm

1. Introduction

Sparse matrix-vector multiplication (SPMV) arises in numerous computational areas, such as eigenvalue problems and the solution of large-scale sparse linear system. Krylov subspace solvers [3, 4] and algebraic multigrid solvers [1, 2, 4] are general methods for these linear systems. For these linear solvers, the sparse matrix-vector multiplication operations control the total running time. It's critical to design efficient sparse matrix-vector multiplication algorithms.

NVIDIA Tesla GPUs are powerful in floating point calculation [13, 14]. Take the NVIDIA Telsa C2050 as an example. This GPU has a peak performance of 1030GFlops on single precision calculation, while the latest CPUs have about 100GFlops [11]. GPUs are much faster than CPUs. Nowadays, GPU computing has been popular in various scientific computing applications due to its superiority over conventional CPU. GPUs have been applied to FFT [13], Krylov subspace solvers [11, 12, 6], algebraic multigrid solvers [1] and reservoir simulation [11, 6].

In this paper, we focus on the sparse matrix-vector multiplication on GPU. Baskaran et al. developed a SPMV algorithm for CSR (Compressed Sparse Row) format matrix, where each row was calculated by a half warp [5]. Li et al. designed a SPMV algorithm for the JAD (Jagged Diagonal) format matrix [12, 11]. Bell and Garland investigated different kinds of matrix formats and SPMV algorithms in [9, 10], in which the HYB format matrix and the corresponding SPMV algorithm was also designed. Here a similar idea to HYB is applied, and we develop a new matrix format, HEC, which is hybrid of ELL format matrix [8] and CSR format matrix. A new sparse matrix-vector multiplication algorithm is also developed. This SPMV algorithm is efficient and the new matrix format is friendly to preconditioners, such as ILU(k), ILUT and domain decomposition preconditioners [7]. Authors investigated vector algorithm and serial algorithm for CSR format in [9, 10]. The memory access pattern for serial algorithm isn't coalesced, so this SPMV kernel is alway the worst. For the vector version, when the matrix is very dense, each warp has sufficient tasks to complete and the memory access is regular, its performance can be better than others. Memory access for ELL format is always coalesced, but it may consume too much memory even if one row is too long. HYB and HEC

Received by the editors January 1, 2012 and, in revised form, March 22, 2012.
2000 *Mathematics Subject Classification.* 65F30, 65F50, 65Y05, 65Y10, 68W10.
This research was supported by NSERC/AIEE/Foundation CMG and AITF Chairs.

are hybrid formats, and most non-zeros values are stored in ELL part. Memory access for these two formats are coalesced and memory usage is moderate. The HYB format and HEC format are general formats for numerical linear algebra. The algorithm difference between HYB and HEC formats is how to calculate the COO part and CSR part. Authors in [9, 10] used reduction operations, and in this case the communication may be complicated. In this paper, we use one thread for each row in CSR part and therefore there isn't any communication. From the numerical experiments we can see that our algorithm is better in most cases.

The layout is as follows. In §2, the new matrix format and the SPMV algorithm are introduced. In §3, numerical experiments are performed to test our SPMV algorithm and other SPMV algorithms.

2. Sparse Matrix-Vector Multiplication

In this section, we will investigate commonly used matrix formats and our new matrix format, HEC, is introduced. Then the sparse matrix-vector multiplication algorithm for GPU is proposed.

2.1. Matrix format. The ELL format was introduced in ELLPACK [8], which is shown in Figure 1. From the figure we can find that this matrix contains two matrices. The left matrix is for the column indices and the right matrix is for the non-zero values. The row length of these two matrices is the same. It's clear that the memory access pattern for the ELL format matrix is regular, and therefore, the performance is high. The disadvantage of the ELL matrix is that even if one row has too many non-zero values, all rows should have the same length. In this case, it's a huge waste of memory space, which is limited on current GPUs.

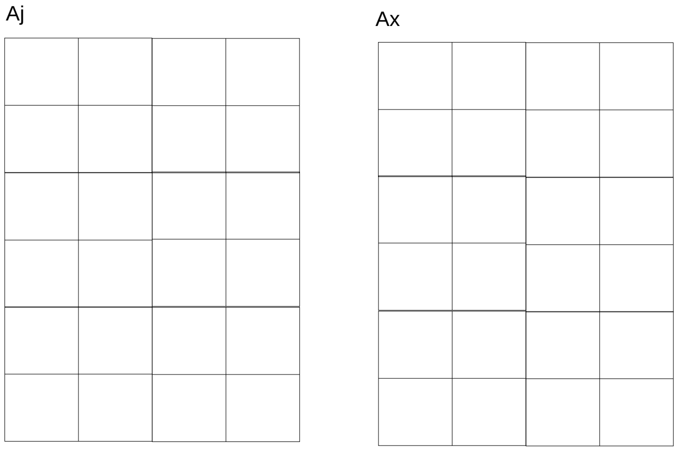

Figure 1: ELL matrix format.

The COO (Coordinate) format is shown in Figure 2. The matrix has three arrays, which are for row indices, column indices and non-zero values. The three arrays has the same length, which is the number of non-zero values. Data access pattern for this matrix is regular. The shortcoming is that when one row is split into different threads on GPU, we need to apply reduction operations to obtain final result. This matrix has good average performance, but may not have the best peak performance.