氏　　名	河村 知記			
学位の種類	博士(情報科学)			
学位記番号	博情第 441 号			
学位授与年月日	令和 2 年 9 月 24 日			
論文題目	GPGPU による超大規模連立一次方程式の求解高速化に向けた省メモリ指向疎行列格納方式に関する研究			
論文審査委員	主査	井口 寧	北陸先端科学技術大学院大学	教授
		金子 峰雄	同	教授
		田中 清史	同	准教授
		本郷 研太	同	准教授
		古山 彰一	富山高等専門学校	教授

論文の内容の要旨

Recent year, large scale numerical simulations on On-site environment is widely exploited in many fields as General-purpose computing on graphics processing units (GPGPU) technology advances. GPGPU is a technology that apply GPU for general application in order to acceleration, not for image processing. Since GPU has huge calculation cores compared to CPU, applications can be calculated by high parallelism on GPU. Solving a large linear system is necessary for calculating Finite Different Method (FDM) and Finite Element Method (FEM) which are often used for numerical simulation. Therefore, the acceleration of solving a large linear system by using GPGPU leads to reducing the calculation time of the numerical simulation. Huge memory space is required to store a sparse matrix which represent a linear system during the solving linear system. The sparse matrix must be transferred to GPU memory from CPU memory before calculation in order to solve the linear system on GPGPU. However, if a sparse matrix does not fit GPU memory, GPU spends much time on the frequency of data transfer between CPU memory and GPU memory. To solve this problem, we propose two memory-saving sparse matrix storage formats, Pattern Compression (PatComp) and Row Block Packing (RBP), that take into account the regularity of sparse matrices generated by FDM and FEM. The PatComp method focuses on the fact that the order of non-zero elements in several rows of a sparse matrix is the same. By using patterns to represent the positions of non-zero elements, memory usage can be reduced. By patterning the sequence of non-zero elements in each row of a sparse matrix and registering the pattern in a table, it is possible to decode the positions of all the non-zero elements in a row from a few patterns. Therefore, the memory usage of PatComp is much lower than that of existing methods. In order to solve the problem that the usable problems of PatComp are limited due to its long conversion time, the RBP method is designed to speed up the conversion process. Assuming that some parts of the column indices in the

sparse matrix are consecutive and such parts can be described with its minimum and maximum column number. RBP method is able to conversion at high speed because finding minimum and maximum columns number is easy. In our experiments, the PatComp method reduces the memory usage of the general sparse matrix up to 31.1% in 13 out of 15 matrices compared with conventional storage format. Also, the RBP method reduces the memory usage of the general sparse matrix up to 28.0% in 13 out of 15 matrices. in the evaluation of Sparse Matrix-Vector multiplication (SpMV) execution time, the PatComp method and RBP method reduce the execution time of SpMV in several sparse matrices. In the evaluation of the execution time of GMRES which is a kind of iterative method, we confirmed that conversion time of RBP method is shorter than PatComp method. From this result, the RBP method is able to adopt a lot of problems. In conclusion, the PatComp method and RBP method are able to drastically reduce the memory usage of the sparse matrix. Also, the execution time of SpMV using PatComp and RBP are comparable to conventional sparse matrix formats. We are able to execute larger numerical simulations because of reducing memory usage of the sparse matrix by the PatComp method and RBP method.

Keywords: GPGPU, SpMV, Sparse matrix format, Data compression, FEM

論文審査の結果の要旨

PC のグラフィックス処理ユニット(GPU)は高速なグラフィックスを実現するため、非常に高い演算性能を持っている。これを汎用の計算に用いるべく、汎用 GPU(GPGPU)による高速演算が注目され、分野によっては CPU に比べて 1〜2 桁以上高い演算性能を達成している。しかしながら、GPGPU は元来グラフィックス向けに開発されているため、汎用の CPU に比べて (1) 利用可能なメモリ容量が少ない、(2) 性能が発揮されるのは多数の同一演算であり、メモリアクセスが連続する場合、などの大きな制約がある。特にメモリ容量の制約に関しては、汎用 CPU の処理系が数 TB の物理メモリが実装可能なのに対し、GPGPU では多くても 64GB 程度とかなり小さく、処理の分割やそれに伴う CPU-GPGPU 間データ転送がボトルネックになる問題がある。

そこで本研究ではメモリ使用量に注目し、有限要素法などのシミュレーションで多用される連立一次方程式の疎行列を効率良く圧縮し、少ないメモリ容量の GPU で大規模な疎行列の計算を可能とすることを試みた。

最初に、行列内のパターン性に注目し、パターンごとに符号とその数値内容を記録する PatComp 法を提案している。PatComp 法は既存の疎行列向けの圧縮手法の中で最も高い圧縮率を持つ BCCOO よりもさらに 7.7%ほど高い圧縮率を達成している。さらに GPGPU での高速演算に必要なメモリアクセスも連続化しており、従来手法に比べて最速ではないもののかなり高速な演算速度も達成している。PatComp 法の問題点は圧縮に時間が掛かることであるが、これを OpenMP による並列化で実用的な時間内での圧縮が可能であることも確認している。

一方で非定常なシミュレーションなどでは、シミュレーションの進行に応じて行列が刻々と変化し、その都度圧縮を行う必要がある。これに毎回 PatComp 法を適用するのでは、処理全体の速度が低下してしまうので、圧縮処理の高速化を目指した RBP 法も提案した。RBP 法は行列中の非ゼロ要素の数値とその場所の始めと終わりのみを記憶する手法であり、従来の高速圧縮手法である ELL や CSR と同等の圧縮時間でありながら、これらよりも高い圧縮率を達成した。

以上、本論文は GPGPU 向けに、定常問題向けには PatComp、非定常問題向けには RBP と、用途に応じて従来手法よりも高い疎行列圧縮を提供する手法を開発した。今後の GPGPU による高速計算の適用範囲を広げるものであり、現実の数値計算において非常に有用な成果である。よって博士(情報科学) の学位論文として十分価値あるものと認めた。