様式3

論 文 内 容 の 要 旨

氏 名 （ 沈靖程 ）	
論文題名	Data-Centric Computing Techniques for Out-of-Core GPU Applications （アウトオブコアGPU応用のためのデータ中心計算技術）

論文内容の要旨

Graphics processing units (GPUs) are highly efficient devices for high-performance computing (HPC) and are widely adopted to construct heterogeneous (i.e., CPU-GPU) supercomputers. However, the small GPU memory capacity necessitates the use of out-of-core computation to process excess data, involving the time-consuming CPU-GPU data transfer prone to limit the overall performance.

In this thesis, for two out-of-core GPU applications, we propose data-centric computing techniques to address the problem that the CPU-GPU data transfer limits the overall performance. The first application is a branch-and-bound (B&B) method to solve the 0-1 knapsack problem, which is a widely used combinatorial optimizer. The second application is stencil computation that occurs in various scientific fields. The proposed techniques significantly reduce the CPU-GPU data transfer, improving the performance of the studied applications. Moreover, the two applications have very different runtime characteristics. Most notably, the B&B 0-1 knapsack solver changes memory footprint at runtime, whereas the stencil computation uses a fixed amount of memory at runtime. Despite the difference between applications, the proposed techniques follow the same mindset of data-centric computing: making as much use as possible of the data residing on the GPU and reducing as much as possible of the CPU-GPU data transfer.

The proposed data-centric computing techniques are in detail explained in three parts of the thesis. In the first part, an out-of-core B&B method to solve large 0-1 knapsack problems on a GPU is proposed. Given a large problem that produces many subproblems, the proposed method dynamically swaps subproblems to CPU memory. Because such a CPU-centric subproblem management scheme increases CPU-GPU data transfer, we adopt three data-centric techniques to eliminate this side effect, including (1) an out-of-order search (O3S) technique that reduces the data transfer overhead by adaptively transferring subproblems between the CPU and GPU, (2) a GPU stream compaction technique that reduces the sparseness of arrays to be transferred, and (3) an explicitly-managed pipelining technique that hides the data transfer overhead by overlapping data transfer with GPU B&B operations.

Experimental results demonstrate that the out-of-core solver with proposed techniques stored 41× as many subproblems as a previous in-core solver that manages subproblems in GPU memory, solving approximately twice as many problem instances on the GPU. In addition, compared to a previous breadth-first search (BFS) technique, the proposed O3S technique achieved an average speedup of 7.5×.

In the second part, we extend a stencil framework to address data transfer problems. We propose two data-centric computing techniques: (1) a direct-mapping technique that eliminates host (i.e., CPU) buffers, which intermediate between the original data and device buffers, and (2) a region-sharing technique that significantly reduces CPU-GPU data transfer by allowing contiguous data chunks to share common regions on the GPU. The extended framework was applied to an acoustic wave propagator, automatically extending the length of the original serial code 2.3-fold to generate the out-of-core code. Experimental results reveal that the generated code ran 41.0×, 22.1×, and 3.6× as fast as implementations based on Open Multi-Processing (OpenMP), Unified Memory, and the previous framework, respectively. The generated code also demonstrates usefulness with small datasets that fit in the device capacity, running 1.3× as fast as an in-core implementation.

In the third part, we propose a data-centric technique that further accelerates out-of-core GPU stencil computation with on-the-fly GPU compression, introducing a novel data compression scheme that solves the data dependency between contiguous decomposed data chunks. We also modify a widely used GPU compression library to support pipelining that overlaps data transfer with GPU computation. Experimental results show that the stencil code with the proposed technique achieved a speedup of 1.13× with a tolerable fidelity loss, compared with a code that involves no compression.

Our work demonstrates that data transfer should be weighed more than computation in optimizing large GPU codes. The two studied applications show that the proposed data-centric computing techniques effectively improve overall performance and thus emphasize the importance of data-centric computing techniques to present and future large-scale GPU applications.

様式 7

論文審査の結果の要旨及び担当者

氏　名　（　沈　靖程　）		
	（職）	氏　名
論文審査担当者	主 査　教 授 副 査　教 授 副 査　教 授	伊野　文彦 井上　克郎 増澤　利光

論文審査の結果の要旨

本学位論文は，Graphics Processing Unit（GPU）を装備する計算機において，GPUメモリ容量よりも大きなデータを対象とするアウトオブコア計算を加速するためのデータ中心計算技術をまとめたものである．GPUシステムにおいて性能を律速する，CPU・GPU間のデータ転送量を削減することを目的として，以下の3つの研究成果を得ている．

１．計算順序の入れ替えによる削減手法

学位論文の2章では，単純にはGPUメモリが不足してしまう規模の0-1ナップザック問題を対象として，CPU・GPU間のデータ転送量を削減するためのアウトオブオーダ探索手法を提案した．提案手法は，探索木に対する分枝操作時にGPUメモリが枯渇しないように，CPUへデータを退避するとともに，探索の進行とともに疎になるデータを密データに圧縮し，GPUメモリの使用量を節約する．GPUメモリ上で処理が完結するインコア手法に対し，提案手法は41倍の部分問題を扱うことができ，幅優先探索と比べて7.5倍の高速化を得た．

２．GPU上のデータ再利用による削減手法

3章では，離散方程式を解くソルバにおいて頻出する大規模ステンシル計算を対象として，GPU上の計算結果を再利用することでCPU・GPU間のデータ転送量を削減する手法を提案した．提案手法は，同一のGPUに割り当てる計算のうち，GPUに転送済みのデータで再利用できるものに対しては再送を避ける．また，提案手法を簡潔に記述するためのコンパイラ指示文をトランスレータとともに開発し，プログラム記述のための労力を軽減する．提案手法をC言語で記述された音響シミュレータに適用した結果，2.3倍の行数からなるGPUプログラムを自動生成でき，マルチコアCPU版に対して41倍の速度向上を得た．また，データ再利用により3.6倍の高速化を得た．

３．GPU上のオンザフライデータ圧縮による削減手法

4章では，3章と同様の大規模ステンシル計算を対象として，GPU上で並列動作するオンザフライ圧縮手法を提案した．提案手法は，GPUへ転送する必要のあるデータを一度に圧縮するのではなく，隣接するデータ間でデータ依存のある部分とそうでない部分に区別して圧縮することにより，パイプライン処理時の実行効率を高める．さらに，GPU上で動作する既存の非可逆圧縮手法cuZFPを基に，ソフトウェアパイプラインを構築することで，CPU・GPU間のデータ転送を処理しながら，GPU上で圧縮ならびに展開を並行処理する．圧縮なしの場合と比較して，提案手法は1.13倍の高速化を達成した．また，圧縮に伴う計算誤差は10^{-12}未満であり，音響シミュレータの計算精度に関して問題がないことを確認した．

以上のように，本学位論文で得られた研究成果は，GPUメモリが枯渇してしまう大規模科学技術計算を高速化する際に役立つだけでなく，その実現のための労力の軽減に寄与する．よって，本論文は博士（情報科学）の学位論文として価値のあるものと認める．