氏名（本籍）	福原淳司（北海道）
学位の種類	博士（理学）
学位記番号	甲第1299号
学位授与の日付	2023年3月19日
学位授与の要件	学位規則第4条第1項該当
学位論文題目	**Compiler Optimization for GPU Based on Code Motion** **（コード移動に基づくGPU向けコンパイラ最適化）**
論文審査委員	（主査）教授　滝本　宗宏 教授　桂田　浩一　　教授　前田　譲治 教授　竹村　　裕　　教授　宮崎　　智

論文内容の要旨

CPUの性能が頭打ちになってくるにつれて，Graphics Processing Unit（以降，GPU）が汎用的な高性能計算に活用されるようになってきている．CPUは多くても数十の計算コアしか持たないが，GPUは数百から数千のコアを持ち，数千のスレッドを並列に実行することができる．この並列実行によって，GPUはCPUよりも高い計算能力を有しており，画像処理や機械学習，科学シミュレーションなどの多くの分野で活用されている．さらに，GPUを活用するためのプログラミング環境は，ここ十数年で急速に発展しており，GPU向けプログラムを簡単に記述できるようになった．このプログラミング環境の発展は，GPUの応用分野をさらに広める役割を果たしている．

GPU向けプログラミング環境として最も広く使われているものの一つにNVIDIA社製のGPUに用いられるCompute Unified Device Architecture（以降，CUDA）がある．CUDAは，Single Instruction Multiple Thread（以降，SIMT）実行形式を採用しており，ウォープと呼ばれるスレッドグループに単一の命令を発行し並列実行を行うことで，高速演算を可能にしている．しかし，プログラム中の分岐命令でウォープ内のスレッドが異なる方向に分岐する場合に，全ての分岐先の命令を順に実行する分岐発散という問題が生じる．分岐発散が生じると，全ての分岐先に実行コストが生じるのに加えて，並列性が低下するので，プログラムの実行効率が低下する．

GPU が広く活用されるようになり，GPU 向けコンパイラは，実行効率をより向上させるために，GPU 向けプログラムを最適化する必要性が高まっている．一般的な GPU 向けコンパイラは CPU 上で実行されるホストコードと GPU 上で実行されるデバイスコードを分けてコンパイルする．このとき，デバイスコード内の関数はカーネルと呼ばれる．GPU 向けの最適化は，その適用範囲の違いから，主にカーネル内最適化とカーネル間最適化に分類される．カーネル内最適化は単一のカーネルを解析し，最適化するもので，各カーネルの実行効率を改善する．一方で，カーネル間最適化は複数のカーネルとホストコードを解析し，最適化するもので，GPU 向けプログラム全体の実行効率を改善する．

従来のコンパイラが行う強力なコード最適化手法に，コード移動に基づいたものがある．コード移動に基づいた手法の一つである部分冗長除去法 (Partial Redundancy Elimination. 以降，PRE) は，全ての実行経路で冗長である全冗長な式を除去するだけでなく，いくつかの実行経路だけで冗長である部分冗長な式を除去することができる．さらに，一部のループ不変式は，部分冗長な式と見なせるので，ループ不変式をループの外に移動するループ不変コード移動も行うことができる．PRE は任意の実行経路上に新しい計算を導入しない性質を持つので，いくつかの実行経路に新しく計算を導入する投機的コード移動は行わないという特徴を持つ．投機的コード移動は，新しく計算を導入するので，実行効率を低減する可能性があるが，特定の条件で積極的に投機的コード移動を行うことによって，全冗長な式を増やし，より多くの式を除去できる場合もある．また，PRE がループの繰返し間で冗長になる式を解析しないのに対し，繰返しを跨いで冗長になるメモリアクセスを除去するようにした手法として，スカラ置換が提案されている．一方で，PRE やスカラ置換をカーネルに適用すると，式を移動する際に，分岐発散を生じる分岐の先に式を挿入することによって，分岐発散を増大させ，実行効率を減じる可能性があり，カーネルへの適用は難しかった．

本稿では，従来のコード移動に基づく手法を拡張することによって，カーネル内とカーネル間のそれぞれにおいて有効な最適化手法を提案する．まず，カーネル内最適化として，冗長な式を除去するだけでなく，カーネルの実行効率の低下を引き起こす分岐発散を低減する手法を，PRE とスカラ置換のそれぞれを拡張することによって提案する．次に，カーネル間最適化として，複数のカーネルを一つにまとめるカーネル融合の手法をコード移動に基づいて拡張することで，より多くのカーネルを融合できるようにする手法を提案する．

まず，PRE を拡張したカーネル内最適化として，PRE を分岐発散の有無に応じて投機的コード移動を行うように拡張した投機的疎なコード移動 (Speculative Sparse Code Motion. 以降，SSCM) を提案する．SSCM は，分岐発散が生じている分岐の両方の分岐先が実行されるという性質を利用して，実行効率を低減することなく投機的コード移動を行う．この投機的コード移動によって，分岐発散を生じている分岐先の式を減らすことができるので，分岐発散を低減し，実行効率を改善する．さらに，投機的コード移動によって分岐の上に巻き上げられた式は，より多くの式を冗長にするので，従来の PRE よりも多くの冗長な式を除去することができる．しかし，分岐発散を生じていない分岐に対して投機的コード移動を行うと，いくつかの実行経路に新しく式を導入し，実行効率を低減する可能性がある．その実行効率の低減を避けるために，SSCM では分岐発散を生じている分岐だけに投機的コード移動を行

う選択的な適用を可能にした．SSCMの効果を確認するために，従来手法とSSCMをGPU向けコンパイラに実装し，いくつかのベンチマークプログラムに適用した結果を比較した．実験の結果，従来手法よりも最大で8%の実行効率の改善が得られることを示した．さらに，SSCMは冗長除去を行う際にレジスタ圧力を高める点と，カーネルのスレッド数を多くすると1つのスレッドが使えるレジスタが少なくなる点から，カーネルのスレッド数を変更した実験も行った．その結果，実行効率を減じる場合もあったが，60%以上の実行効率の改善が得られることも示した．

次に，スカラ置換を拡張したカーネル内最適化として，質問伝播に基づく投機的スカラ置換（Speculative Scalar Replacement Based on Question Propagation．以降，SSRQP）を提案する．質問伝播とは，式eの出現ごとに「eは冗長であるか」というクエリをプログラム上に伝播させることで冗長性を解析する要求駆動型の手法である．SSRQPは，質問伝播を用いてスカラ置換を実現した質問伝播に基づくスカラ置換を，分岐発散の有無に応じて投機的コード移動を行うように拡張した手法である．SSRQPでは，SSCMと同様に分岐発散を考慮した選択的な投機的コード移動を行い，分岐発散の低減と冗長除去を実現する．SSCMが従来のPREのデータフロー解析を拡張して投機的コード移動を行っているのに対し，SSRQPは式eの出現ごとに「eは投機的コード移動が可能か」というクエリをプログラム上に伝播させ，その結果に基づいて投機的コード移動を行う．この解析法によって，SSRQPはSSCMよりも多くの式を投機的に巻き上げるコード移動を実現している．また，SSCMが，従来のPREに基づく性質から，依然として分岐発散を増大させる可能性を持つのに対し，SSRQPでは分岐発散の有無に応じた式の挿入を行う．すなわち，式の挿入先が分岐発散を生じている分岐先でない場合にだけ式を挿入することによって，分岐発散の増大を回避する．　SSRQPの効果を確認するために，従来手法とSSRQPをGPU向けコンパイラに実装し，様々なベンチマークプログラムに適用した結果を比較した．実験の結果，SSRQPは従来手法と比べて最大で40%以上の実行効率の改善が得られることを示した．

最後に，カーネル間最適化として，コード移動に基づくカーネル融合手法（Kernel Fusion based on Code Motion．以降，KFCM）を提案する．カーネル融合は複数のカーネルを一つに合成する手法である．CUDAの実行モデルでは，カーネルの入力と出力は最も遅いグローバルメモリに格納する必要があり，多くのGPUアプリケーションにおいて，その遅いメモリ通信がボトルネックになっている．また，一般的なGPUアプリケーションは，複数のカーネルを持ち，それらのカーネルがデータを共有していることが多い．カーネル融合手法はデータを共有しているカーネルを融合することで，GPUの遅いメモリ通信を削減し，実行効率を改善することができる．また，カーネル融合手法は，カーネルを融合することでカーネルの数を減らすので，カーネルを起動するオーバーヘッドを削減する．さらに，カーネルを融合すると融合前よりも大きなカーネルができるので，カーネル内最適化の適用範囲を大きくし，最適化機会を増加させる効果がある．しかしながら，従来のカーネル融合手法は，プログラム上で連続して実行されるカーネルしか融合できず，多くの融合の機会を損なっているという問題があった．KFCMでは，データフロー解析に基づいたコード移動をカーネルに適用することによって，制御フローを考慮したカーネルの融合可能性を増大し，従来手法よりも

多くのカーネル融合を実現する．KFCM は，このより多くのカーネル融合を通して，より多くのメモリアクセスや起動オーバーヘッドの削減を行い，実行効率を改善する．KFCM の効果を確認するために，従来手法と KFCM をコンパイラ基盤上に実装し，様々なベンチマークプログラムに適用した結果を比較した．実験の結果，KFCM は従来手法と比べて最大で 50%以上のカーネル起動数の削減を行い，最大で 35%の実行効率の改善が得られることを示した．

本稿では，コード移動手法に基づいて，カーネル内最適化とカーネル間最適化の両方の手法を提案した．カーネル内最適化では分岐発散の低減，カーネル間最適化ではカーネル融合の新しい手法をそれぞれ提案し，従来手法よりも実行効率が向上することを実験で示した．今後の研究として，まず，SSRQP は，分岐発散の有無に応じた式の挿入によって，冗長除去の機会を損なっている場合があるので，プログラム全体のコストを考えて式の挿入を行うように拡張することを検討している．また，KFCM には，融合可能なカーネルの全てを融合し，融合する際のコストを考慮しないという問題があるので，それぞれの融合のコストと利益を考慮する解析を加えることを検討している．

論文審査の結果の要旨

本学位論文は、プログラムの一部を他の場所に移動するコード移動という手法を用いて、並列演算装置である GPU 向けのプログラムを効率化する３つの手法について述べている。本論文について、内容の学術的な新規性、独創性および有用性について審査を行なった。

GPU は、高性能計算を行うほとんどのコンピュータに備わっており、最近では、深層学習を含む多くの機械学習における成果を支える基盤である。その GPU 上で動作するプログラムの高速化は、IT 社会の今後の発展にとって必要不可欠な技術である。

本論文は、次の７章から構成されている。

第１章では、本研究における背景および目的を示すとともに、本論文の構成を述べている。この中で、本研究の動機の一つである分岐発散を紹介し、本研究で扱う３つのコード最適化手法が、２種類の粒度から成ることを述べている。

第 2 章では、本研究を理解するための基礎知識を与えている。特に、GPU の詳細と GPU 向けのプログラムを記述する CUDA 言語および、GPU のプログラムの構成が GPU 上で実行されるカーネルと、ホスト側でカーネルの実行を制御するホストプログラムから成ることを説明している。また、GPU プログラムの最適化には、プログラム構成に基づいて、カーネル内最適化とカーネル間最適化の２つの粒度があることを説明している。

第３章では、本研究の基礎を与えるコード移動について説明している。特に、データフロー解析に基づいてコード移動を行う部分冗長除去の実行命令数を増やさない性質を解説し、実行命令数を増やす可能性があっても、多くの実行でコストを低減する場合に

コード移動を行う投機的なコード移動について説明している。

第４章では、本論文で提案するカーネル内最適化として、疎な投機的コード移動（以下 SSCM と呼ぶ）と質問伝播に基づく投機的スカラー置換（以下 SSRQP と呼ぶ）を提案している。SSCM は、冗長な命令を削除するとともに、分岐発散を低減できることを述べている。また、SSRQP は、メモリアクセスについて、ループの反復を跨いで冗長性を低減するとともに分岐発散を低減できることを述べている。さらに、SSCM と SSRQP をベンチマークで評価した結果、従来法と比較して、それぞれ、最大で 60%以上、40%以上の実行効率の改善が見られたことを述べている。

第５章では、カーネル間最適化として、ホスト側のカーネル呼出しにコード移動を適用することによって、能動的に複数のカーネル呼出しとカーネル本体をそれぞれ融合し、カーネルの実行コストを低減するコード移動に基づくカーネル融合（以下 KFCM と呼ぶ）を提案している。本章では、後向きのコード移動と前向きのコード移動を組み合わせることによって、多くのカーネル呼出しを連続させ、融合を可能にするアルゴリズムの詳細を述べている。また、KFCM をベンチマークに適用して評価した結果、従来手法と比較して、最大で 50%以上のカーネル起動数の削減を実現し、35%以上の実行効率の改善が見られたことを述べている。

第６章では、分岐発散の低減とカーネル融合の側面から、関連研究を説明した後、他の GPU プログラムの最適化手法についても触れている。

最後に第７章で、本論文のまとめを述べ、今後の研究の発展性について議論している。

本論文は、現在の計算基盤として欠かせない GPU 上のプログラムを高速化する研究として大変高く評価できるものであり、この分野に大きな貢献があるばかりでなく、実践的な利用が期待できると考えられる。以上から、理学的に価値ある知見と成果を得たものと判定し、博士（理学）の学位論文として十分価値あるものと認めた。