

研究報告書

「自動チューニング可能な一般化 N 体問題解法枠組みの開発」

研究期間：2019年4月～2021年3月
研究者番号：50224
研究者：佐藤 重幸

1. 研究のねらい

効率的な並列計算を実現するプログラムを正しく作ることは、一般に難しい。計算対象に関する高水準の要求仕様から、計算機に関する低水準の要求仕様まで、幅広い専門知識を動員し、多大な工数を掛けて慎重にプログラミングする必要がある。その上、そのように苦労して開発した高性能なソフトウェアは、しばしば再利用し難い形となり、本質的に類似する計算であっても、具体的な応用問題毎に、別々にハードコーディングされることが一般的である。特に本研究で注目する N 体問題は、様々な応用領域に見られるような汎用性がありつつ、それを解くアルゴリズムの実装(ソルバ)は、用途毎にハードコーディングされている。そこで本研究では、N 体ソルバの為にドメイン特化言語(DSL)を設計し、その DSL 処理系によって効率的な並列ソルバの開発を助ける。提案 DSL は、様々な N 体アルゴリズムが記述できるような表現能力を持ちつつ、並列計算機を隠蔽する高水準抽象化を提供する。一方、DSL 処理系は、並列計算機に適応した効率的で正しいコードを生成する。これにより、高性能 N 体ソルバに関する車輪の再発明を防止し、多分野に渡る N 体ソルバプログラミングの無駄を削減することを目指す。

2. 研究成果

(1) 概要

大きく分けて次の 4 つを実施した。

- A) 様々な N 体ソルバに対応可能なドメイン特化言語(DSL)を開発した。
- B) N 体ソルバの効率化手法をコンパイラ最適化として定式化し、DSL コンパイラに実装した。
- C) 複数の複雑な N 体ソルバに対して提案コンパイラ最適化を適用し、性能向上を実験的に評価した。
- D) 多分野の応用における提案手法の正しさと妥当性を実証するための N 体ベンチマーク集を開発した。

(2) 詳細

A) 様々な N 体ソルバに対応可能な DSL

Python ベースの言語内 DSL として、N 体 DSL を設計し、C++を生成するコンパイラを実装した。言語設計として特徴的なのは、次の 2 点である。

- N 体カーネルとそれを適用するソルバに記述を分離させたこと。
- 入力データセットをネストした有限集合と述語に限定したこと。

第 1 の点により、カーネルの代数的性質とソルバの計算構造が明確になり、プロ

グラムが読み易く、それぞれに特化した効率化手法を適用しやすくした。第2の点は、多様なデータ構造の詳細を隠蔽し、アルゴリズム記述を高水準化させる。実際のN体ソルバで用いられる様々なデータ構造の実装の詳細を意識させるアルゴリズム記述は、人にとって仕様が分かりにくい上に、機械的な変換を難しくする。高速化のためのデータ構造レベルのトリックは、集合及び述語の実装として、実行時ライブラリの責任として分離させた。

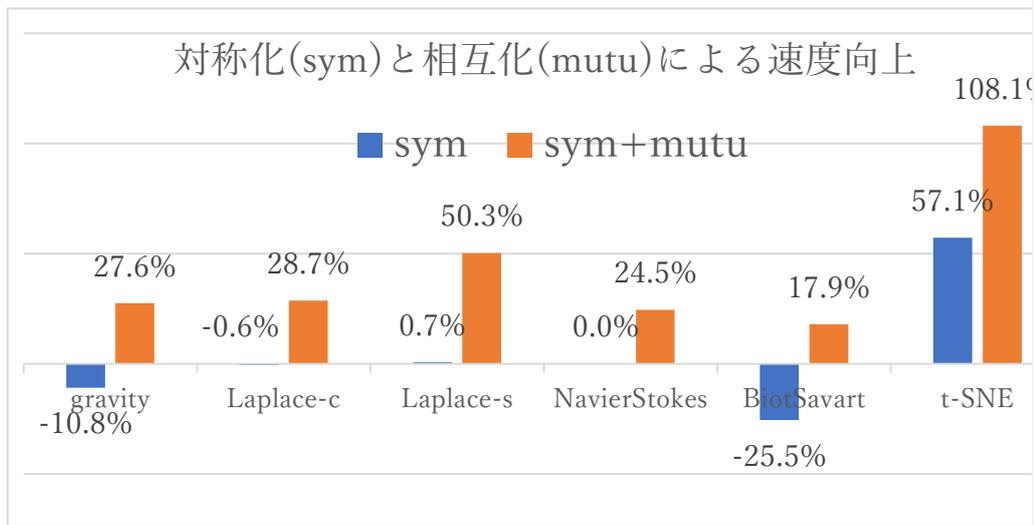
B) N体DSLのコンパイラ最適化

N体カーネルは、重力作用の作用と反作用のように、2つのデータ対に対して、部分的な対称性を持った形で適用されることが多い。一般的には、手作業で対称性を考慮した実装を与えるが、これには専門家の知識が要る。これを自動化するように、プログラムの代数的性質に関するプログラム解析を設計し、対称性が基づく冗長性を発見・除去する**相互化**というコンパイラ最適化を実装した。同時に、ソルバの計算構造を集合論と述語論理で定式化し、対称性のある部分計算を特定して特殊化させる**対称化**というコンパイラ最適化を実装した。

一方、並列化の観点では、N体ソルバの直積構造に着目した並列化技法を開発した。SIMD命令と呼ばれるデータ並列性を持つ命令の利用(SIMD化)は、通常のコンパイラでは、最内ループに対して限定される。しかし、それでは複雑なカーネルを含んだソルバを、効率的にデータ並列処理することが叶わない。そこで、カーネル全体をパラメトリシティに基づいてSIMD化し、ソルバ部分の最内の直積構造を変換して、そのSIMD化カーネルを呼び出す手法を開発した。これにより、カーネルがどれだけ複雑であっても、N体ソルバが共通してもつ直積構造に沿って、効率的なデータ並列処理ができるようになる。

C) 提案DSLとコンパイラ最適化の実験的評価

複数の異なるカーネルとソルバを用いて、コンパイラ最適化の効果を測定した。次の図は、特に対称化と相互化の速度向上である。



この実験で、100行以上もあり、5-6重のループネストを持つような複雑なカーネルコードを用いたソルバについての、性能向上にも成功した。このような複雑なループに対して、既存の汎用コンパイラは、対称化や相互化のような高水準の大域的な最適化は適用できない。その上、N体ソルバに特化したSIMD化手法によって、複雑なカーネルコードに含まれる数学関数などの重い計算をも含めてSIMD化され、性能が向上した。

D) 近似N体ベンチマーク集の開発

対称化や相互化のような代数的性質を利用したプログラム変換において問題になるのは、実計算機上での結果の正しさと妥当性である。数学的な数字(例えば有理数)として考えれば、プログラム解析によって正しさは保証されているが、実計算機では浮動小数点数演算を用いるので、演算順序に依存した丸め誤差が生じる。したがって、提案するコンパイラ最適化が、丸め誤差に有意な差を生まないことが実用上重要である。更に難しいことに、誤差は、丸め誤差からのみ生じる訳ではない。実用的なN体ソルバは、しばしば近似アルゴリズムを採用しており、モデルレベルの誤差が生じる。提案処理系を、実際のアプリケーションで用いるには、モデルレベルの誤差の許容範囲に収まることを検証する必要がある。それを検証するためのベンチマークとして、近似N体ソルバアプリケーションを4つ開発した。

- H-matrix(境界要素法, 電界解析)
- t-SNE(クラスタリング, 可視化)
- gravity(重力N体シミュレーション, 球状星団)
- SPH(流体シミュレーション, 表面波)

これらのベンチマークで重要なのは、合理的な入力データセットの生成と共に、モデルレベルの誤差パラメータを持ちつつ、出力に関して応用上の許容される誤差基準を、それぞれに設計した点である。これによって、提案するコンパイラ最適化に対する評価基盤というだけでなく、実用的な近似計算のベンチマーク集として、精度と速度のトレードオフを調べる基盤としても利用できるようになっている。

3. 今後の展開

プログラミング言語の実装研究という観点では、DSL処理系の基盤を作れたという面が大きい。開発したDSLコンパイラは、提案するN体DSLにしか使えない枠組みではなく、他の用途のDSLにもほぼそのまま転用できる仕組みとなっている。例えば、テンソル計算のDSLなどに発展させ、機械学習分野のプログラミングに貢献できるような実装技術の開発などの展開が考えられる。そして、本研究で扱った近似計算に対する高水準最適化というのは、未だに発展途上であり、今後も発展させられる余地が大いにある。例えば、同じ部分計算の結果を保存して再利用することで計算を効率化する漸増化を、近似計算に対して拡張する展開などが考えられる。

本プロジェクトで開発した近似N体ベンチマークは、近似計算を支援するプログラミング言語技術の評価基盤として即座に利用可能だが、プログラミング言語分野に限らず有用なものである。近似計算は、ハードウェア分野でも研究が進んでおり、低信頼だが低電消費のCPUや、

低信頼だが高速なメモリなどが研究されている。そのようなハードウェアの評価基盤としても利用できる。現段階では公開していないが、速やかに permissive license でオープンソース化する予定である。

より大きな観点で言えば、本研究は、N 体ソルバを通して、用途毎にコンパクトな DSL を利用することの意義を実証できたと考えている。実社会では、既存のソフトウェア資産と同じ言語で、新しいソフトウェアを作りがちである。DSL に基づくプログラミング言語技術によって、既存ソフトウェアを上回るような高品質なソフトウェアが手軽に作れるようになることの認知度が上がれば、社会全体のソフトウェア開発の生産性向上に寄与する、

4. 自己評価

研究目的の達成状況

ソフトウェア開発を重点的に進め、2 年間という期間内に目途がやっと立ったが、結局、トップ国際会議に論文出版には至らなかった。研究の性質上、ある程度仕方がない面はあるが、2 年間においてポスター発表のみで論文出版が間に合わなかったという出版の少なさは大きな課題である。

研究の進め方(研究実施体制及び研究費執行状況)

研究費執行の計画は当初の予定から変わり、結果としてフィクスターズとの共同体制に資金の大部分を投じることになったが、結果としては大成功だった。フィクスターズが持つ産業側のノウハウと、自身が持つアカデミア側のノウハウを協調させる研究マネジメントを実施できたことは、自身の研究経験における大きな財産となった。

修士学生に研究補助を依頼しつつ、それに関連した、その学生の修士研究になるようなテーマを割り当てるという学生マネジメントの経験が積めたことも良かった。

研究成果の科学技術及び学術・産業・社会・文化への波及効果

近似 N 体ベンチマーク集は、計算を扱う学術分野において、大きなインパクトをもたらす可能性があると考えている。近似計算は、技術的に扱うことは可能だが、それを評価することが難しい。単にプログラムだけがあるのではなく、そのモデル式やパラメータの意味や制約、そして出力検証まで、きちんとドキュメント化されたベンチマーク集は、大変貴重である。

既存の汎用言語と違って、研究レベルの DSL は、発表段階では誰も使ったことが無く、しばしば、それ自体はすぐには役立たないものである。しかし、DSL 研究を通して、様々なプログラミング言語を使い分けることの意義が広がることは、プログラミングという活動を、より豊かで生産的なものにすると考えている。

研究課題の独創性・挑戦性

コンパイラ最適化では、変換後のプログラムの正しさが至上命題である。その限界を超えるような、少し正しくない結果が生じることがあっても、応用上の常識から許容される範囲のより効率の良いプログラムを得ることは、これまで人が、経験的に手作業で行ってきたことである。本研究で、その 1 つの形を形式化し自動化できたということは、プログラミングの自動化という観点で大変意義深いことである。

5. 主な研究成果リスト

(1) 論文(原著論文)発表

1. 著者. 発表論文タイトル. 掲載誌名. 発行年, 巻号, 始頁-終頁, その他

(2) 特許出願

研究期間累積件数:0 件

(3) その他の成果(主要な学会発表、受賞、著作物、プレスリリース等)

Sato, S. A Symmetry-Based N-Body Solver Compiler. SPLASH Companion 2019: Proceedings Companion of the 2019 ACM SIGPLAN International Conference on Systems, Programming, Languages, and Applications: Software for Humanity. Poster.