様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

# 科学研究費助成事業　　研究成果報告書

令和　６ 年　 ５ 月 １４ 日現在

機関番号：１２６１２
研究種目：基盤研究(B)（一般）
研究期間：2020～2023
課題番号：２０Ｈ０４１９３

研究課題名（和文）ポストペタスケールのための革新的アプリケーション解析基盤技術の開発

研究課題名（英文）Development of Innovative Frameworks for Application Analysis in Post-Peta Scale Systems

研究代表者
三輪　忍（Miwa, Shinobu）

電気通信大学・大学院情報理工学研究科・准教授

研究者番号：９０４０２９４０

交付決定額（研究期間全体）：（直接経費）　　13,300,000円

研究成果の概要（和文）：2020年4月から2024年3月まで研究を実施し，並列アプリケーションのプロファイル予測とトレース予測に関する数々の研究成果を得た．具体的には，プロファイル予測として関数コール回数とキャッシュミス回数それぞれの予測手法，ならびに，MPI通信トレースとメモリアクセストレースの各トレースの予測手法の開発を行った．さらに，開発したプロファイル予測手法の他のプラットフォームへの応用を目的とした調査研究も実施した．本研究課題では3件の国際共同研究を実施し，本研究成果の一部は著名な国際会議や英文論文誌にて発表した．

研究成果の学術的意義や社会的意義
本研究成果の一部は著名な国際会議や英文論文誌にて発表したことから，並列アプリケーションの性能解析に大きなインパクトを与えたと言える．本研究課題によって達成した並列アプリケーションの性能解析コストの削減は，今後の並列アプリケーション開発の速度向上とアプリケーションそのものの速度向上に繋がる成果であり，計算科学分野のさらなる発展に資するものである．

研究成果の概要（英文）：This research project was conducted from April 2020 to March 2024 and produced various notable results related to predicting the profiles and traces of parallel applications. More specifically, we developed two methods for predicting the function call and cache miss counts as profile prediction, while we developed two methods for predicting the MPI communication and memory access traces as trace prediction. In addition, we surveyed many profilers used in various platforms such as GPUs and Intel SGX to extend the proposed methods to these platforms. We performed three collaborations with four oversea researchers in this research project. The results of this research project were partially presented in an authorized international workshop and a top journal in the field of high performance computing.

研究分野：高性能計算

キーワード：高性能計算　プロファイル　トレース　予測

※科研費による研究は、研究者の自覚と責任において実施するものです。そのため、研究の実施や研究成果の公表等については、国の要請等に基づくものではなく、その研究成果に関する見解や責任は、研究者個人に帰属します。

## １．研究開始当初の背景

並列アプリケーションの解析は，性能分析や性能チューニングなどを目的として，高性能計算分野において広く行われている．並列アプリケーション解析は，アプリケーション全体，あるいは，関数単位の実行時間，演算回数，メモリアクセス回数，通信関数の呼び出し履歴などの情報をもとに行われるが，これらの情報は，通常，解析対象のアプリケーションを解析対象のシステム上で実行することによって得られる実行時情報である．この実行時情報を取得する手段として，本研究課題の開始当初はプロファイリング/トレーシング（PR/TR）とモデリングの 2 つが存在した．

PR/TR は，コンパイラや外部ライブラリの支援によって計測用コードを解析対象のアプリケーションに挿入し，解析対象のシステム上で上記アプリケーションを実行することによって実行時情報を取得する．TAU を始めとするさまざまなツールがこれまでに開発されている．アプリケーションや関数単位の実行時間のようなプロファイルから，通信関数の呼び出し履歴（トレース）のような詳細情報までさまざまなレベルの情報が取得可能である．PR/TR はアプリケーションの正確な実行時情報を取得できる一方，上述のようにアプリケーション（全体，あるいは，一部）の実行を必要とすることから，情報の取得に要する時間はアプリケーションの実行時間と計測用コードの実行時間の合計によって制限される．

これに対してモデリングは，少数のノードにおけるプロファイリング結果からアプリケーションのスケーラビリティモデルを構築し，構築したモデルを用いて上記アプリケーションを多数のノード上で実行した場合のアプリケーションや関数単位の振る舞いを予測する．代表的なツールに Extra-P がある．モデリングは，解析対象の並列度によるアプリケーション実行が不要なことから，数千から数万ノードを必要とする超大規模アプリケーションの振る舞いを高速に予測できる．しかしながら本研究課題開始当初のモデリングツールが予測可能な情報は限定的であり，PR/TR によって取得可能なその他の実行時情報は得ることができなかった．

このように，上記 2 つの手法は得られる情報量（＋精度）と情報の取得コストとの間にトレードオフが存在する．そのため，本研究課題の開始当初は，簡易的な性能見積もり（1 次解析）で十分な場合にはモデリングが，より詳細な解析（2 次解析）が必要な場合には PR/TR が利用されている状況であった（図 1 左）．

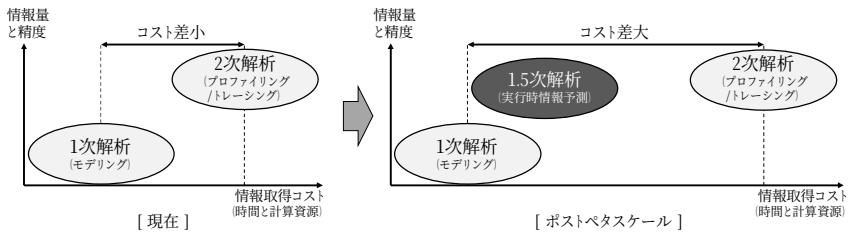


**図１．本研究課題で解決しようとする問題**

スーパーコンピューティングシステムは，計算科学分野アプリケーションの大規模化・複雑化と呼応してシステム規模を増大することにより発展してきた．例えば，我が国において本研究課題開始当初に実施されていたフラッグシップ 2020 プロジェクトでは，膨大な計算時間を必要とする大規模シミュレーションやビッグデータ解析などのアプリケーションを実行する場合を想定し，京コンピュータを上回る規模（CPU 数にして約 2 倍の規模）のシステムである「富岳」の開発を行っていた．そのため，「富岳」の運用開始後は，多くのエンドユーザが，京コンピュータ上で実行していたアプリケーションよりもコード量の多いアプリケーションを，より高い並列度で実行するようになることはほぼ確実な状況であった．

この傾向は，並列アプリケーション解析における PR/TR とモデリングのギャップを拡大する（図 1 右）と予想された．並列アプリケーションの詳細解析には PR/TR が用いられるが，PR/TR にはアプリケーションそのものの実行が必要なため，アプリケーション実行に必要な時間・計算資源の増加に比例して PR/TR に必要な時間・計算資源も増加する．また，PR/TR の結果は一般にファイルへ出力されるが，アプリケーションの並列度が増加するとファイル I/O も増加するため，PR/TR のオーバヘッドは並列度の増加にともなって増大する．一方，モデリングに必要な時間・計算資源は，元々，PR/TR と比較して極めて小さいことから，システム規模やアプリケーション規模の増大の影響も（絶対値としては）軽微である．

PR/TR は広く利用されている技術であるが，その情報取得コストの大きさから，ポストペタスケールでは利用可能な場面が限定的にならざるを得ないと予想された．一方，モデリングによって得られる実行時情報は限定的であり，アプリケーションの細かなチューニングなどに必要な詳細情報は得ることができなかった．既存技術のこれらの制約によって，ポストペタスケールでは，エンドユーザによる並列アプリケーション開発が阻害されると予想された．

## ２．研究の目的

本研究課題の目的は，上述のギャップを埋める新たな並列アプリケーション解析基盤技術を

開発することであった．具体的には，少数のノード上で対象アプリケーションを静的・動的に解析することにより，同アプリケーションを多数のノード上で実行した場合の実行時情報を予測する手法（実行時情報予測）を開発することを目指した．実行時情報予測を利用することにより，エンドユーザは，数千から数万ノードの超並列環境において PR/TR を行った際に得られる情報量に匹敵するアプリケーションの実行時情報を，モデリングに近い速度で取得できる（表 1）．実行時情報予測が提供する実行時情報は，モデリングと同様，近似的であるが，並列アプリケーションのプロセス配置最適化などの用途においては，正確な情報ではなく近似的な情報で十分である．実行時情報予測は，現在の 1 次解析と 2 次解析との間に位置する，言わば 1.5 次解析を可能にする（図 1）．並列アプリケーションの PR/TR が資源的・時間的制約により難しい場合においても，実行時情報予測が近似的な実行時情報を提供することにより，ポストペタスケールにおけるエンドユーザの生産性の向上に資する．

表 1．並列アプリケーションの実行時情報の取得方法

| | PR/TR | モデリング | 実行時情報予測 |
|---|---|---|---|
| 解析速度 | 低速 | 高速 | 高速 |
| 情報の種類 | 豊富 | 性能のみ | 豊富 |
| 情報の粒度 | プロファイル/トレース | プロファイル | プロファイル/トレース |
| 情報の精度 | 正確 | 近似 | 近似 |

## ３．研究の方法

前述のように，PR/TR によって得られる実行時情報にはプロファイルとトレースの 2 種類がある．これらは次元の異なる情報であるため予測には異なる戦略が必要なことから，本研究課題では以下の方法によりプロファイルとトレースそれぞれに対して最適な予測手法を開発する予定であった．

### ３．１．最適なプロファイル予測の開発

並列アプリケーションの性能予測を性能以外の情報の予測へ拡張する．すなわち，小規模なプロファイリングによって得られた情報をもとに並列度や問題サイズとその情報との関係性を表すモデルを推定し，このモデルを用いて，対象アプリケーションに対して大規模なプロファイリングを行った場合の当該情報を予測する．ただし，性能以外の情報と並列度・問題サイズとの詳しい関係はまだ明らかでないことから，さまざまな並列度・問題サイズにおけるアプリケーションのプロファイリング結果を分析し，各情報と並列度・問題サイズとの関係を最もよく表現可能な関数を推定する．また，並列アプリケーションの性能以外の情報のモデルには，性能モデルとは異なる入力変数が必要な可能性が高い．例えば，キャッシュミス率は，アプリケーションのメモリフットプリントサイズ，各キャッシュメモリのサイズにも依存するため，これらの情報もモデルの入力変数として必要なことが予想される．そこで，アーキテクチャの観点から当該情報に関係するパラメータを列挙し，それらの中から当該情報の表現に必要な入力変数を絞りこむことにより，当該情報と並列度・問題サイズとの関係を表す最適なモデルを構築する．

### ３．２．最適なトレース予測の開発

対象アプリケーションのソースコードを分析することによって得た情報と小規模実行によって得たトレースをもとに，当該アプリケーションを大規模実行した場合のトレースを予測する．MPI 通信関数の呼び出し履歴のような並列アプリケーションの詳細な挙動を正確に予測するためには，トレースだけでなくアプリケーション内の変数やループ構造などのコードレベルの情報が必要と考えられる．そこで，並列アプリケーションのソースコードに対して静的解析を行うことによってトレース生成に関与する変数とループを抽出し，各変数に対して並列度・問題サイズとの関係性を表すモデルを構築する．モデルのパラメータ推定には小規模実行によって得たトレースを利用する．大規模実行時のトレースは，上記の方法によって構築したモデル集合を用いて，アプリケーションの変数レベルの挙動を推定することにより予測する．

## ４．研究成果

2020 年 4 月から 2024 年 3 月まで本研究課題を実施し，並列アプリケーションのプロファイル予測とトレース予測に関する数々の研究成果を得た．以下ではそれぞれの研究成果について述べる．

### ４．１．プロファイル予測に関する研究成果

プロファイルに含まれる実行時情報の中で関数コール回数とキャッシュミス回数に着目し，それぞれの実行時情報を予測する手法を開発した．具体的には，少ないコア数，小さな問題サイズで取得したプロファイルを用いて当該実行時情報を予測するモデルをフィッティングし，フィッティングにより得られたモデルを用いて多いコア数，大きな問題サイズで当該プログラムを実行した際の当該実行時情報を予測する手法を開発した．予測に使用するモデルとして，線形，反比例など 4 種類のモデルを開発した．NPB を用いて評価を行ったところ，小さな問題サイズ（クラス A, B, C）の実行結果から大きな問題サイズ（クラス D）の実行結果を予測する場合において，関数コール回数については誤差率 19.32%，L1 データキャッシュミス数については誤差

率約 40%で予測できた．また，提案手法により，プロファイル取得に要するコストをどちらの場合も約 5%に削減できた．上記の実験には TSUBAME3.0 の 64 コアを使用した．以上の研究成果は情報処理学会第 178 回 HPC 研究会にて発表を行った．

さらにモデルに改良を加えたことにより，並列数とコア数の両方を同時にスケールさせた場合においても高い精度で関数コール回数と L1 データキャッシュミス数を予測できるようになった．これにより，モデルのパラメータフィッティングに必要なプロファイルの取得コストを大幅に軽減することができ，プロファイル予測が有用なケースを増加させることができた．

その後，既存の性能モデリングツール（Extra-P）に関する複数の資料（一部の資料は本研究課題の採択後に公開）を調査・分析したところ，この手法は実行時間以外の性能メトリクスの予測にも応用できることがわかった．そこで，Extra-P と本研究課題で開発した手法を用いた場合で関数コール回数予測の精度比較を行ったところ，Extra-P の方が予測精度が高いことが判明した．そのため，関数コール回数予測手法の開発を中止し，予測された関数コール回数の応用について検討を行った．応用の 1 つとして，予測された関数コール回数を用いた場合に関数の実行時間予測精度が向上することを実験的に確認した．以上の研究成果は情報処理学会第 187 回 HPC 研究会にて発表した．

一方，キャッシュミス回数予測に関しては，Extra-P と本研究課題で開発した手法の予測精度を比較したところ，本研究課題で開発した手法の方が高い予測精度を示すことが確認できた．そこで，上記の予測手法の改良に引き続き取り組むとともに，この手法の適用範囲を L1D キャッシュから L2 キャッシュと L3 キャッシュに拡大した場合の評価実験を行った．その結果，提案手法は L2 キャッシュのミス回数予測においては十分高い精度を示すことが確認できたものの，L3 キャッシュのミス回数予測においては予測精度が著しく悪化することを確認した．以上の研究成果は情報処理学会第 187 回 HPC 研究会にて発表した．

また，本研究課題で開発したプロファイル予測技術の他のプラットフォームへの応用を想定し，GPU と Intel SGX におけるプロファイラの調査・分析等を行った．特に Intel SGX に関しては，オープンソースの SGX ライブラリ OS（例えば Gramine）と組み合わせることで既存の性能プロファイラ（例えば perf）が利用できるケースもあり，本研究課題で開発した手法をほぼそのまま流用できることがわかった．なお，本研究は米国ジョージタウン大学の研究者と共同で実施したものであり，その成果の一部は高性能計算分野のトップカンファレンスの 1 つである SC23 の併設ワークショップにて発表を行った．

４．２．トレース予測に関する研究成果

MPI アプリケーションの通信トレースを予測する技術，および，任意の並列アプリケーションのメモリトレースを予測する技術の開発を行った．

通信トレース内の通信時刻以外の情報に関しては，MPI 並列アプリケーションのソースコードを解析し，各通信関数に渡される引数の値（通信先のランク番号や通信量）などを予測する変数モデルを自動で構築し，構築した変数モデルを用いて元のソースコードを通信トレース予測プログラムへ自動変換するツールを開発した．上記のツールは，LLVM の最適化パスとして実装したため，MPI+C/C++や MPI+Fortran などの任意の MPI 並列アプリケーションの解析に利用できる上，x86, POWER, ARM などの任意の ISA のトレース予測プログラムを生成できる．研究代表者らが行った実験結果によると，開発したツールは，平均 0.11%の誤差と引き換えに既存のツールと比較して最大 1,612 倍高速に通信トレースを収集できることがわかった．また，開発したツールによって収集された近似的な通信トレースが，MPI プロセスのノード配置最適化に利用できることを示した．なお，本研究は Lawrence Livermore National Laboratory（米国）の研究者，および，ミュンヘン工科大（ドイツ）の研究者と共同で実施した．以上の研究成果をまとめた論文は，高性能計算分野のトップジャーナルの 1 つである IEEE Transactions on Parallel and Distributed Systems にて発表した．

通信トレース内の通信時刻に関しては，先行研究で提案された通信時刻予測手法の問題点を明らかにするとともにその解決法を考案し，その有効性を示した．具体的には，先行研究（ScalaExtrap）の通信時刻予測手法を評価した結果，同手法には 1) ランク毎のばらつきを正確に予測することができない，2) 通信関数の呼び出し回数が並列数以外にも依存するようなアプリケーションでは予測精度が特に悪化する，という 2 つの問題点があることがわかった．そこで本研究課題では，通信時刻予測のためのモデルとして並列数と問題サイズを入力変数とするモデルを新たに提案し，提案モデルの評価を行った．その結果，通信関数の呼び出し回数が並列数以外にも依存するアプリケーションに対して，提案モデルは先行研究のモデルよりも高い予測精度を示すことを確認した．以上の研究成果は情報処理学会第 182 回 HPC 研究会にて発表した．

並列アプリケーションのメモリアクセストレース予測に関しては．具体的には，小規模実行時のメモリアクセストレースから対象アプリケーションのメモリアクセストレースを予測するモデルを生成し，生成したモデルを用いて大規模実行時のメモリアクセストレースを予測する手法の開発を行った．モデルはロード/ストア命令単位でフィッティングを行うことで生成する．

ただし，間接参照等により不規則なメモリアクセスパターンを示す命令に関してはモデルによる予測が難しいと考えられることから，当該アドレスの計算に必要な命令のみを実行することでアドレスを生成する．アドレス計算のみを行うプログラムはアプリケーションコードから LLVM を用いて自動生成する．具体的には，アドレス計算に必要な命令をプログラムスライシングによって抽出し，抽出された命令のみからなるコードを生成する LLVM パスを実装する．本研究で対象とするアプリケーションは LLVM によってコンパイル可能な任意の並列アプリケーションであり，したがって MPI アプリケーションだけでなく Charm++等の非 MPI アプリケーションも含まれる．

上記の LLVM パスは，MPI 通信トレース予測の研究で研究代表者が開発した LLVM パスを元に開発する．ただし，通信トレース予測の研究で開発した LLVM パスは LLVM-3.6.0 を対象としており，最新の LLVM-19.0.0 とは複数のビルトイン関数のインターフェースが異なるため，最新の環境では動作しない．そこで，以前の研究で開発した LLVM パスを LLVM-19.0.0 に移植する作業を行った．移植作業は概ね完了しており，現在動作確認中である．なお，本研究は米国メリーランド州立大学の研究者と共同で実施した．

５．主な発表論文等

〔雑誌論文〕　計2件（うち査読付論文　2件／うち国際共著　2件／うちオープンアクセス　1件）

| | |
|---|---|
| １．著者名<br>Miwa Shinobu、Laguna Ignacio、Schulz Martin | ４．巻<br>32 |
| ２．論文標題<br>PredCom: A Predictive Approach to Collecting Approximated Communication Traces | ５．発行年<br>2021年 |
| ３．雑誌名<br>IEEE Transactions on Parallel and Distributed Systems | ６．最初と最後の頁<br>45～58 |
| 掲載論文のDOI（デジタルオブジェクト識別子）<br>10.1109/TPDS.2020.3011121 | 査読の有無<br>有 |
| オープンアクセス<br>オープンアクセスではない、又はオープンアクセスが困難 | 国際共著<br>該当する |

| | |
|---|---|
| １．著者名<br>Miwa Shinobu、Matsuo Shin'Ichiro | ４．巻<br>none |
| ２．論文標題<br>Analyzing the Performance Impact of HPC Workloads with Gramine+SGX on 3rd Generation Xeon Scalable Processors | ５．発行年<br>2023年 |
| ３．雑誌名<br>SC-W '23: Proceedings of the SC '23 Workshops of The International Conference on High Performance Computing, Network, Storage, and Analysis | ６．最初と最後の頁<br>1850～1858 |
| 掲載論文のDOI（デジタルオブジェクト識別子）<br>10.1145/3624062.3624267 | 査読の有無<br>有 |
| オープンアクセス<br>オープンアクセスとしている（また、その予定である） | 国際共著<br>該当する |

〔学会発表〕　計5件（うち招待講演　0件／うち国際学会　0件）

| |
|---|
| １．発表者名<br>有馬 海人, 長谷川 健人, 三輪 忍, 八巻 隼人, 本多 弘樹 |
| ２．発表標題<br>LULESHを対象とした関数コール回数予測 |
| ３．学会等名<br>情報処理学会研究報告 2022-HPC-187 |
| ４．発表年<br>2022年 |

| |
|---|
| １．発表者名<br>長谷川 健人, 有馬 海人, 三輪 忍, 八巻 隼人, 本多 弘樹 |
| ２．発表標題<br>並列アプリケーションのキャッシュミス数予測の評価 |
| ３．学会等名<br>情報処理学会研究報告 2023-HPC-188 |
| ４．発表年<br>2023年 |

１．発表者名
岡田 悠希，三輪 忍，八巻 隼人，本多 弘樹

２．発表標題
MPIにおける小規模実行時の通信トレース解析による大規模実行時の通信タイミング予測の評価

３．学会等名
情報処理学会研究報告 2021-HPC-182

４．発表年
2021年

１．発表者名
有馬 海人，長谷川 健人，三輪 忍，八巻 隼人，本多 弘樹

２．発表標題
MPIアプリケーションの関数コール回数予測

３．学会等名
第178回HPC研究会

４．発表年
2021年

１．発表者名
長谷川 健人，有馬 海人，三輪 忍，八巻 隼人，本多 弘樹

２．発表標題
MPIアプリケーションのキャッシュプロファイル予測

３．学会等名
第178回HPC研究会

４．発表年
2021年

〔図書〕　計0件

〔産業財産権〕

〔その他〕

―

６．研究組織

| | 氏名<br>（ローマ字氏名）<br>（研究者番号） | 所属研究機関・部局・職<br>（機関番号） | 備考 |
|---|---|---|---|

７．科研費を使用して開催した国際研究集会

〔国際研究集会〕　計0件

８．本研究に関連して実施した国際共同研究の実施状況

| 共同研究相手国 | 相手方研究機関 | | | |
|---|---|---|---|---|
| 米国 | Lawrence Livermore National Laboratory | Georgetown University | The University of Maryland | |
| ドイツ | Technical University of Munich | | | |