

Title	大規模なメッセージ通信並列プログラムに対する実行履歴の生成およびデバック支援
Author(s)	置田, 真生
Citation	大阪大学, 2006, 博士論文
Version Type	
URL	<a href="https://hdl.handle.net/11094/46672">https://hdl.handle.net/11094/46672</a>
rights	
Note	著者からインターネット公開の許諾が得られていないため、論文の要旨のみを公開しています。全文のご利用をご希望の場合は、 <a href="https://www.library.osaka-u.ac.jp/thesis/#closed">〈a href="https://www.library.osaka-u.ac.jp/thesis/#closed"〉</a> 大阪大学の博士論文について <a>〉</a> をご参照ください。

***Osaka University Knowledge Archive : OUKA***

<https://ir.library.osaka-u.ac.jp/>

Osaka University

氏名	替 置 田 眞 生
博士の専攻分野の名称	博 士 (情報科学)
学位記番号	第 20477 号
学位授与年月日	平成 18 年 3 月 24 日
学位授与の要件	学位規則第 4 条第 1 項該当 情報科学研究科コンピュータサイエンス専攻
学位論文名	大規模なメッセージ通信並列プログラムに対する実行履歴の生成および デバッグ支援
論文審査委員	(主査) 教授 萩原 兼一  (副査) 教授 増澤 利光 教授 楠本 真二

#### 論 文 内 容 の 要 旨

メッセージ通信パラダイムは、クラスタおよびグリッドなどの分散メモリ型並列計算環境に適したプログラミング手法である。例えば、メッセージ通信仕様 MPI (Message Passing Interface) を用いればこれらの環境において高性能かつ移植性の高い並列プログラムを記述できる。このパラダイムでは、プログラムの並列性をソースコードに明記する必要があるため、その開発において開発者がプログラムの動作を正確に把握することが重要である。そこで、並列プログラムの開発を支援するための多くのツールが存在する。しかし、多数のプロセッサで長時間に渡って実行する大規模な並列プログラムを対象とした場合、既存のツールによる支援は十分でない。本論文では、大規模な並列プログラムの開発支援を目的とし、(1)デバッグ作業の煩雑さの軽減、および(2)性能改善のための大規模な実行履歴の生成に取り組んだ。

まず(1)では、異常停止するバグを持つ大規模な並列プログラムを対象に、デバッグ支援ツール MPI-PreDebugger (MPI-PD) を開発、実装した。並列プログラムでは、あるプロセッサがバグによって異常停止すると、プロセッサ間の依存によって異常が伝播し、本来バグを含まないプロセッサも連鎖的に異常停止する。このため、大規模なプログラムの異常停止時の実行状況は複雑となる。MPI-PD は、プログラム実行時に通信ごとに異常の発生を確認し、異常発生時に異常停止の連鎖を解析することで、異常停止の原因となったプロセッサを自動的に検索する。MPI-PD を用いてバグを含むと考えられる範囲を絞り込むことで、バグ特定におけるプログラムの負担を軽減できる。今回の評価実験では、MPI-PD を用いることでデバッグ時間を最大で約 6 分の 1 に削減でき、デバッグの煩雑さを軽減できた。

次に(2)では、大規模な並列プログラムの実行情報を正確に記録する手法を提案、実装した。多くの開発支援ツールは、プログラムの実行情報を実行履歴に記録する。しかし、実メモリの容量を超えて実行履歴を生成する場合、性能計測の障害となる性能摂動の発生が顕著となり、正確な性能計測が困難となる。提案手法では、性能摂動を小さく抑えつつ大規模な実行履歴を生成する手法を実現した。今回の実験では、提案手法を用いない場合と比較して性能摂動による実行時間の増大を約 4 分の 1 以下に削減できた。また、性能摂動を含まない本来の実行を 1.8% の誤差で再現できた。

## 論文審査の結果の要旨

メッセージ通信パラダイムは、クラスタおよびグリッドなどの分散メモリ型並列計算環境に適したプログラミング手法である。例えば、メッセージ通信仕様 MPI (Message Passing Interface) を用いればこれらの環境において高性能かつ移植性の高い並列プログラムを記述できる。しかし、このパラダイムでは、プログラムの並列性をソースコードに明記する必要があるため、その開発において開発者がプログラムの動作を正確に把握することが重要である。そこで、並列プログラムの開発を支援するための多くのツールが存在する。しかし、多数のプロセッサで長時間に渡って実行する大規模な並列プログラムを対象とした場合、既存のツールによる支援は十分でない。本論文では、大規模な並列プログラムの開発支援を目的とし、(1)デバッグ作業の煩雑さの軽減、および(2)性能改善のための大規模な実行履歴の生成に取り組んだ。

まず(1)では、異常停止するバグを持つ大規模な並列プログラムを対象に、デバッグ支援ツール MPI-PreDebugger (MPI-PD) を開発、実装した。並列プログラムでは、あるプロセッサがバグによって異常停止すると、プロセッサ間の依存によって異常が伝播し、本来バグを含まないプロセッサも連鎖的に異常停止する。このため、大規模なプログラムの異常停止時の実行状況は複雑となる。MPI-PD は、プログラム実行時に通信ごとに異常の発生を確認し、異常発生時に異常停止の連鎖を解析することで、異常停止の原因となったプロセッサを自動的に検索する。MPI-PD を用いてバグを含むと考えられる範囲を絞り込むことで、バグ特定におけるプログラムの負担を軽減できる。評価実験では、MPI-PD を用いることでデバッグ時間を最大で約6分の1に削減でき、デバッグの煩雑さを軽減できた。

次に(2)では、大規模な並列プログラムの実行情報を正確に記録する手法を提案、実装した。多くの開発支援ツールは、プログラムの実行情報を実行履歴に記録する。しかし、実メモリの容量を超えて実行履歴を生成する場合、性能計測の障害となる性能摂動の発生が顕著となり、正確な性能計測が困難となる。提案手法では、性能摂動を小さく抑えつつ大規模な実行履歴を生成する手法を実現した。提案手法を用いた実験の結果、提案手法を用いない場合と比較して性能摂動による実行時間の増大を約4分の1以下に削減できた。また、性能摂動を含まない本来の実行を1.8%の誤差で再現できた。

以上の研究成果は、大規模化が進む近年の分散メモリ環境において、並列プログラムの開発者に対して大規模な並列プログラムのデバッグの煩雑さを軽減し、また、多数存在する有用な開発支援ツールの大規模なプログラムに対する適用可能性を高めた点で有用であり、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。