



# リンク構造を扱う共有メモリ型並列プログラムの自動最適化

鎌田, 十三郎

---

(Degree)

博士 (工学)

(Date of Degree)

2004-03-12

(Date of Publication)

2008-11-07

(Resource Type)

doctoral thesis

(Report Number)

乙2740

(URL)

<https://hdl.handle.net/20.500.14094/D2002740>

※ 当コンテンツは神戸大学の学術成果です。無断複製・不正使用等を禁じます。著作権法で認められている範囲内で、適切にご利用ください。



【 3 3 2 】

氏 名・(本 籍)	鎌田 十三郎	(奈良県)
博士の専攻分野の名称	博士(工学)	
学 位 記 番 号	博ろ第263号	
学位授与の 要 件	学位規則第4条第2項該当	
学位授与の 日 付	平成16年3月12日	

【 学位論文題目 】

リンク構造を扱う共有メモリ型並列プログラムの自動最適化

審 査 委 員

主 査	教 授	金田悠紀夫
	教 授	田村 直之
	教 授	増田 澄男

(別紙様式3)

## 論文内容の要旨

氏名 鎌田 十三郎

論文題目 (外国語の場合は、その和訳を併記すること。)

### リンク構造を扱う共有メモリ型並列プログラムの自動最適化

本論文は、リンクを有する共有データ構造を対象とした並列プログラムに関する、自動高速化/最適化技術を取り扱った論文である。具体的には、(A)プログラムの振る舞いの変化を捉えた変数アクセス解析手法と(B)その解析結果を踏まえた排他制御の緩和技法、加えて、(C)共有メモリ計算機に適したオブジェクト配置法の自動メモリ管理機構への統合と、(D)分散メモリ計算環境のためのオブジェクトキャッシュ技術を取り扱う。

並列計算は大きな計算パワーを提供できる一方で、効率的なプログラムの実現にはしばしば専門的な知識や、ユーザの詳細なコーディング能力が必要となる。リンクデータ構造などを対象とした不規則な計算においては、現状では自動並列化などによる最適化はあまり期待できない。

効率的な並列プログラムを実現するためには、プログラマは様々なことに気をつけてプログラムを記述する必要がある。まず、並列プログラムである以上、アトミックに操作すべき箇所に十分な排他制御を施した正しく並列プログラムを記述しなくては行けないが、一方で、排他制御区間を短くしないとプログラムがボトルネックを発生してしまう。加えて、各種計算資源をどのように配置するかが問題となる。分散環境においては、共有オブジェクトへのアクセスはノード間通信を伴うため、通信コストの削減のためにはオブジェクトのキャッシュを参照側に配置する必要がある。その際、プログラマはオブジェクトの性質を把握した上で、データ一貫性を保証するためのキャッシングを実現する必要がある。一方で、共有メモリ型計算機においても状況はそれほど単純ではない。分散計算環境ほどのコストではないが、プロセッサ間でデータを共有するためには、キャッシュのコヒーレンスミスというペナルティを払う必要があり、状況によってはメモリコンテンションを引き起こすことすらある。このため、プログラムを高速化するためには、しばしば多大な実装技術と職人的コーディング技術が必要とされ、並列計算が一般用途に普及するための障害になっている。

このような現状を打開するため、本研究はプログラマには当初は正しい排他制御を行うプログラムの実現に専念してもらい、その後の高速化に関しては多くを処理系最適化に任せるための方法を模索する。但し、不規則データを対象としたプログラムの解析は一般には難しいため、問題によってはユーザの知識をヒントとしてプログラム中に記述してもらい、あるいは対象プログラムをサンプル実行しプロファイル結果を取得することで、最適化を施していく。

排他制御の問題は、複数スレッドで共有されたデータ構造に対し変数のアクセスを解析し、データ競合が起らない範囲で排他制御区間を短縮することにある。但し、そのためには変数が更新されない区間を求める必要があるが、複数スレッドから共有される変数の状況変化を解析することは従来難しかった。本研究では、ユーザが重要だと考えるオブジェクトの状態を局面として捉え、局面毎のオブジェクトの振る舞いを解析することで変数への更新がない区間を解析し、排他制御の緩和を実現している。

本解析結果は、効率的分散オブジェクトの実現にも応用可能である。分散環境では通信コストの削減が重要であるが、本解析結果を用いることで、ある局面の間更新を行わないと分かった変数を参照側のプロキシの一部としてキャッシュし、遠隔メソッド呼出しの削減という積極的な最適化技法が可能となる。そのためにも、局面と変数更新に関する精度の高い解析が必要とされる。

一方で、共有メモリ型並列計算機上でのオブジェクトレイアウトの問題は、より微妙な最適化が必要である。というのは、前述のようにプロセッサ間の共有変数を介した値の伝達には、キャッシュのコヒーレンスミスを伴うが、他のキャッシュミスに比して絶対的に大きいとも言えないからである。つまり、キャッシュの無効化だけでなく、空間的局所性やキャッシュ密度を総合的に考慮したレイアウトの決定を行う必要がある。本研究では、共有メモリ並列計算機向けのオブジェクトレイアウト法を提案し、自動メモリ管理機構に統合することで、プログラマの明示的メモリ管理からの解放を目指す。本システムは、プログラムの実行プロファイルからオブジェクトの変数アクセス状況を取得し、それに基づき適切なレイアウトへの自動変換を行う。

本論文では、これら実装技術に関して実際に実アプリケーションを通じた評価を行い、その有効性を示すことができた。また、対象言語モデルや開発ターゲットとしてJava言語という一般のオブジェクト指向言語を選んでおり、今後、一般的に利用可能な技術となることを期待している。

氏名	鎌田十三郎		
論文題目	リンク構造を扱う共有メモリ型並列プログラムの自動最適化		
審査委員	区分	職名	氏名
	主査	教授	金田悠紀夫
	副査	教授	田村直之
	副査	教授	増田澄男
	副査		
			印
要 旨			
<p>本論分は、リンクを有する共有データ構造を対象とした並列プログラムに関する、自動高速化/最適化技術を取り扱った論文である。具体的には、(A)プログラムの振る舞いの変化を捉えた変数アクセス解析手法と (B) その解析結果を踏まえた排他制御の緩和技法、加えて、(C) 共有メモリ計算機に適したオブジェクト配置法の自動メモリ管理機構への統合と、(D) 分散メモリ計算環境のためのオブジェクトキャッシュ技術を取り扱う。</p> <p>並列計算は大きな計算パワーを提供できる一方で、効率的なプログラムの実現にはしばしば専門的な知識や、ユーザの詳細なコーディング能力が必要となる。リンクデータ構造などを対象とした不規則な計算においては、現状では自動並列化などによる最適化はあまり期待できない。このため、プログラムを高速化するためには、しばしば多大な実装技術と職人的コーディング技術が必要とされ、並列計算が一般用途に普及するための障害になっている。</p> <p>このような現状を打開するため、本研究はプログラマには当初は正しい排他制御を行うプログラムの実現に専念してもらい、その後の高速化に関しては多くを処理系最適化に任せるための方法を模索する。但し、不規則データを対象としたプログラムの解析は一般には難しいため、問題によってはユーザの知識をヒントとしてプログラム中に記述してもらい、あるいは対象プログラムをサンプル実行しプロファイル結果を取得することで、最適化を施していく。</p> <p>排他制御の問題は、複数スレッドで共有されたデータ構造に対し変数のアクセスを解析し、データ競合が起らない範囲で排他制御区間を短縮することにある。但し、そのためには変数が更新されない区間を求める必要があるが、複数スレッドから共有される変数の状況変化を解析することは従来難しかった。本研究では、ユーザが重要だと考えるオブジェクトの状態を局面として捉え、局面毎のオブジェクトの振る舞いを解析することで変数への更新がない区間を解析し、排他制御の緩和を実現している。</p> <p>一方で、共有メモリ型並列計算機上でのオブジェクトレイアウトの問題は、より微妙な最適化が必要である。本研究では、共有メモリ並列計算機向けのオブジェクトレイアウト法を提案し、自動メモリ管理機構に統合することで、プログラマの明示的メモリ管理からの開放を目指す。本システムは、プログラムの実行プロファイルからオブジェクトの変数アクセス状況を取得し、それに基づき適切なレイアウトへの自動変換を行う。</p>			

氏名	鎌田十三郎
<p>本論文では、これら実装技術に関して実際に実アプリケーションを通した評価を行い、その有効性を示すことができた。また、対象言語モデルや開発ターゲットとしてJava言語という一般のオブジェクト指向言語を選んでおり、今後、一般的に利用可能な技術となることが期待できる。</p> <p>本研究は、リンク構造を扱う共有メモリ型並列プログラムについて、その自動最適化の研究をしたものであり、最適化技術について重要な知見を得たものとして価値ある集積であると認める。よって学位申請者の鎌田十三郎は、博士(工学)の学位を得る資格があると認める。</p>	
<ul style="list-style-type: none"> <li>・特記事項 特になし</li> <li>・特許登録数 0件</li> <li>・発表論文数 11編</li> </ul>	