

論文の内容の要旨

論文題目 A Study on Large Scale Eigensolvers
for Distributed Memory Parallel Machines

(分散メモリ型並列計算機における大規模固有値ソルバの研究)

氏名 片桐 孝洋

今日インターネットの発達により、それを用いた情報検索が重要になってきている。このような情報検索処理は従来、数値計算とは別の非数値計算処理であると考えられてきた。ところが近年、M.Berryらによりこの情報検索処理において検索情報から知識を抽出するために、固有値計算が必要となることが指摘された。この例のように、固有値計算は従来の数値計算に限定された科学技術計算だけではなく、幅広い意味での情報処理全般において基本的な処理であるといえる。また必要となる固有値計算の規模は、インターネット上の検索対象のデータベースが非常に膨大になってきていることや、数値計算が必要な問題は常に計算機性能の限界まで大規模な計算を要求することから、演算量とメモリ使用量の観点から非常に大規模となる。したがって実行時間に関しての高速化や、非常に多くのメモリを利用したいという要求は尽きることがない。この一方で近年、スーパーコンピュータに代表される大規模計算を高速に行うための計算機は、そのほとんどが分散メモリ型並列計算機である。このことから、アルゴリズムやメモリ使用の観点からの並列化が必須となっている。

これらの要求から本論文では第一部として、固有値計算のために必要な各種アルゴリズムの提案と実装評価を行う。次に第二部として、開発された並列プログラムに関する各種の応用を行う。その応用に関して、まず始めにプログラムをライブラリ化する場合において多種の並列計算機環境で高い性能を達成する必要がある、という要求がある。この要求を満たすため、固有値計算用の並列数値計算ライブラリにおいて始めて実行前最適化機能を付加することを提案し、複数の並列計算機を用いてその並列ライブラリの性能評価を行う。また現在、多数の数値計算ルーチンを自動的にチューニングをするソフトウェアが開発されてはいるが、最適化時間が大きな問題となっている。本論文では第二部において、この最適化時間の問題を解決するためのいくつかの方法を議論し、実機による評価を行う。さらに最適化された性能に関するパラメータを利用することで、従来よりも高効率な数値計算ルーチンが構築可能と考えられる新しい数値計算基本ルーチン (BLAS) の分類を提案する。また性能評価から、従来の BLAS よりも効率がよい例を確認した。

本論文の第一部としてまず始めに、対称密行列の固有値問題を解く場合で必要になる Householder-Bisection 法を取り上げる。このアルゴリズムでは Householder 三重対角化が必要になるが、従来ではプロセッサ台数が増加すると通信時間も増加するという問題が生じていた。そこで本論文では、プロセッサ台数が増加しても通信時間が減少するアルゴリズムを提案する。またこの三重対角化において、演算性能を向上させる技法であるブロック化という技法 (アルゴリズムブロッキング) が逐次処理においては利用されていた。一方でブロック化アルゴリズムを分散メモリ型並列計算機上に実装する場合に行列データの分散をしなくてはならないが、あるまとまった単位で行列データをプロセッサに分散する (データ分散ブロッキング)。従来のアルゴリズムでは、このアルゴリズムブロッキングのブロック幅とデータ分散ブロッキングのブロック幅を同一にしていた。このことから問題サイズが小さい場合や超並列環境では負荷バランスが劣化し、高性能なアルゴリズムを構築できないという問題が生じていた。そこで本論文では、アルゴリズムブロッキングのブロック幅とデータ分散ブロッキングのデータ幅を区別してアルゴリズムを設計する概念を提案し、実際の並列計算機にそのアルゴリズムを実装して評価を行った。その結果、現在標準的に使われている ScaLAPACK の同種ルーチンに対して、最大で 5 倍程度の速度向上を得ることができた。

これらの要求から本論文では第一部として、固有値計算のために必要な各種アルゴリズムの提案と実装評価を行う。次に第二部として、開発された並列プログラムに関する各種の応用を行う。その応用に関して、まず始めにプログラムをライブラリ化する場合において多種の並列計算機環境で高い性能を達成する必要がある、という要求がある。この要求を満たすため、固有値計算用の並列数値計算ライブラリにおいて始めて実行前最適化機能を付加することを提案し、複数の並列計算機を用いてその並列ライブラリの性能評価を行う。また現在、多数の数値計算ルーチンを自動的にチューニングをするソフトウェアが開発されてはいるが、最適化時間が大きな問題となっている。本論文では第二部において、この最適化時間の問題を解決するためのいくつかの方法を議論し、実機による評価を行う。さらに最適化された性能に関するパラメータを利用することで、従来よりも高効率な数値計算ルーチンが構築可能と考えられる新しい数値計算基本ルーチン (BLAS) の分類を提案する。また性能評価から、従来の BLAS よりも効率がよい例を確認した。

本論文の第一部としてまず始めに、対称密行列の固有値問題を解く場合で必要になる Householder-Bisection 法を取り上げる。このアルゴリズムでは Householder 三重対角化が必要になるが、従来ではプロセッサ台数が増加すると通信時間も増加するという問題が生じていた。そこで本論文では、プロセッサ台数が増加しても通信時間が減少するアルゴリズムを提案する。またこの三重対角化において、演算性能を向上させる技法であるブロック化という技法 (アルゴリズムブロッキング) が逐次処理においては利用されていた。一方でブロック化アルゴリズムを分散メモリ型並列計算機上に実装する場合に行列データの分散をしなくてはならないが、あるまとまった単位で行列データをプロセッサに分散する (データ分散ブロッキング)。従来のアルゴリズムでは、このアルゴリズムブロッキングのブロック幅とデータ分散ブロッキングのブロック幅を同一にしていた。このことから問題サイズが小さい場合や超並列環境では負荷バランスが劣化し、高性能なアルゴリズムを構築できないという問題が生じていた。そこで本論文では、アルゴリズムブロッキングのブロック幅とデータ分散ブロッキングのデータ幅を区別してアルゴリズムを設計する概念を提案し、実際の並列計算機にそのアルゴリズムを実装して評価を行った。その結果、現在標準的に使われている ScaLAPACK の同種ルーチンに対して、最大で 5 倍程度の速度向上を得ることができた。

は低いことを利用しているからである。このような単純な最適化方法を利用しても、結果として日立 SR2201, SR8000 の各分散メモリ型並列計算機において約 1.1–2.3 倍の性能向上を得ることができた。この結果は、いかに実行前最適化機能の付加が有用であるかを示している。

次に第二部では、本論文で提案された実行前最適化機能を付加した数値計算ライブラリのような自動チューニングソフトウェアをどのように実現したらよいかを議論する。そのため「高性能な」自動チューニングソフトウェアとは何かを定義する。それは (1) プログラムを高速に実行できるパラメタを高速に見つけることができること、(2) その高速なパラメタを見つける時間もまた高速であること、そして (3) 最適化時間を増やせばよりよいパラメタを発見できること、である。このような高性能な自動チューニングソフトウェアを実現するため、4 種の方法を議論した。実機による性能評価の結果から、数値計算ソフトウェアにおいては「不完全な分解」による方法を利用する方法が良いことが明らかになった。この「不完全な分解」による方法とは、多くの線形代数計算はいくつかの基本演算からなる分解処理により構成されていることを利用したものである。実機による実験の結果、同じ性能を達成できるパラメタを発見するまでの最適化時間を約 1/74.1 にできることが判明した。これは約 33 時間の最適化時間が、1591 秒程度に削減されることを意味している。この一方で著者は、*Intelligent BLAS (IBLAS)* とよばれる新しい BLAS における分類を提案する。この IBLAS の概念は、実行前最適化機能の概念を自然に拡張することによって得られたものであり、実行前最適化の概念なしには得られることはできなかった。実機による評価の結果、従来の BLAS は最適ではないことが判明した。なぜならば IBLAS の方がより高速である場合が存在したからである。具体的には IBLAS を利用することで、従来の BLAS を利用した場合に比べ、約 1.02–1.26 倍程度の高速化を達成する場合があった。BLAS を利用しているソフトウェアは容易に IBLAS を利用できる。このことから本論文で提案される IBLAS の適用性は広く、高性能な数値計算ルーチンを構築するために非常に有用であることが期待される。