

課題名 (タイトル) :

自動チューニング機能付き数値ライブラリの研究

利用者氏名 :

片桐 孝洋

所属 :

本所 情報基盤センター

1. 本課題の研究の背景、目的、関係するプロジェクトとの関係

科研姫野プロジェクトとの関係

文部科学省の「次世代スーパーコンピュータ (スパコン) 開発プロジェクト」に対し、計算科学のアプリケーション (応用問題) 分野の研究者や技術者から大きな期待が寄せられている。完成後には国内外で未踏の規模の数値シミュレーションが展開される。

ところが一方で、このプロジェクトで開発されるのはスパコン本体と一部分の基本的ソフトウェアのみ (従来から用意されている線形ライブラリの改良までしか含まれていない) であり、両者のバランスが悪いことが指摘されている。スパコンの性能を余すことなく十分に発揮させるために、応用問題の特性に応じ超並列計算時に最適な求解アルゴリズムを用いることと、スパコン利用を円滑にすることを支援する基盤ソフトウェアの開発が重要となる。本プロジェクトではこのような要請を受け、アプリケーション領域、求解アルゴリズム、基盤システム開発のコンピュータ科学分野の研究者が集う学際フレームワークを形成し、実際の大規模数値シミュレーションに対する有用な求解アルゴリズムの検討と、最適な計算性能を得るための基盤ミドルウェア開発を目的とする。

本課題は上記プロジェクトにおいて、応用問題の特性に応じ超並列計算時に最適な求解アルゴリズムを利用する方式開発を目的にし、新しい超並列数値計算法の開発と、数値計算ライブラリのための自動チューニング機能の開発に寄与するものである。

背景

人類はペタフロップスコンピュータを持った。その並列数 (コア数) は 10 万コアにも達する超並列計算機である。このことは、ペタフロップスの演算性能を達成する計算機環境 (ペタフロップス計算機環境) では、

10 万プロセッサもの超並列性を 1 つのアプリケーション内で達成しなくてはならないことを意味している。一方、従来開発されてきた並列計算機用の密行列数値計算ライブラリは、高効率並列実行のために超大規模行列での実行を前提とし、それに特化したアルゴリズムと実装方式が実現されている。ところが計算量が $O(n^3)$ で増加するため、大規模行列実行には並列化されているとはいえ時間的制約がある。加えてスーパーコンピュータの運営上の制約で、1 ユーザが全系占有できる時間は数時間に限定される。

目的

そこで本課題では、小規模行列に対してコア数を増加しても台数効果が見込める観点の超並列固有値ソルバの方式開発を目的にし、その方式に自動チューニング機能を付加する場合の問題点について、通信時間削減・単体性能向上、およびメモリ量削減の観点から性能評価を行うものである。

2. 具体的な利用内容、計算方法

計算方法

対称密行列 $A \in R^{n \times n}$, 実数 $\lambda \in R$, 実数ベクトル $x \in R^n$ とすると、以下の標準固有値問題

$$A x = \lambda x \quad \dots (1)$$

の解 λ を固有値、ベクトル x を固有ベクトルとよぶ。いま、式 (1) の固有値 n 個を対角要素に並べた行列 Λ を $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ 、固有値 λ_i に対応する固有ベクトル x_i を並べて構成された行列 X を、 $X = (x_1, x_2, \dots, x_n)$ とすると、式(1)は

$$A X = X \Lambda, \quad \dots (2)$$

となる。式(2)の固有値行列 Λ 、固有ベクトル行列 X を求めるが、その手順は以下である：

(Step1) 三重対角化: 行列 A を相似変換で
 三重対角行列 T に変換 ($Q^T A Q = T$)
 (Step2) 三重対角行列 T の固有値行列 Λ を求解
 (Step3) T の固有ベクトル行列 Y を求解
 (Step4) 逆変換: Y を行列 A の固有ベクトル
 行列 X に変換 ($X = Q Y$)

(Step1) 及び (Step4) は $O(n^3)$ である。(Step3) は解法と問題の性質に依存し、 $O(n) \sim O(n^2)$ である。したがって、ソルバ全体の演算量は $O(n^3)$ となる。メモリ量については、入出力行列 A と X が密行列であるため $O(n^2)$ となるのは自明であるが、解法に必要なメモリ量もソルバ全体で $O(n^2)$ となることが知られている。なお、三重対角化に必要な行列 Q は、各反復 k において、 $(n-k+1) \times (n-k+1)$ の行列 A_k から計算される枢軸ベクトル u_k から計算される Q_k を用いて構成される。

$$Q_k = (I - \alpha_k u_k u_k^T), \dots (3)$$

ここで、 α_k は A_k から計算されるスカラー $\in R$ である。

このとき

$$Q = Q_1 Q_2 \dots Q_{n-2}, \dots (4)$$

となる。

利用内容

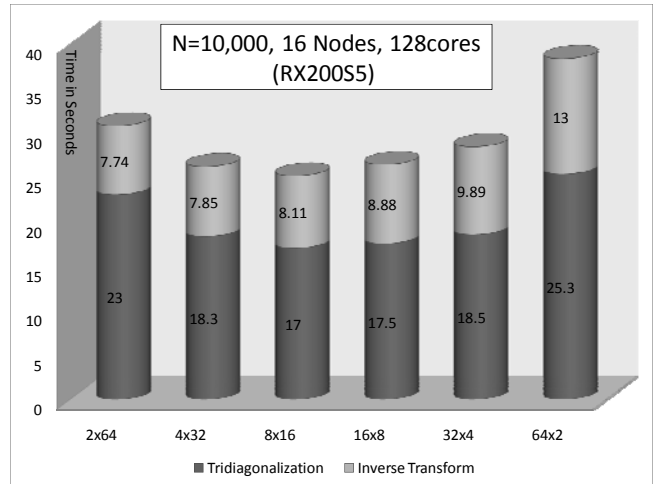
本研究では、既に著者により開発したライブラリ ABCLib_DRSSSED ver. 1.04 に、逆変換処理 (Step4) に新規開発したアルゴリズムを実装し、理研の計算機で性能評価を行うものである。

従来法では並列動作のため、プロセス数 np とすると、 $np=p*q$ のとき (この $p*q$ をプロセッサ・グリッドとよぶ)、従来法では $p=q$ の時のアルゴリズムしか提案されていない。そこで、任意の p, q について動作するアルゴリズムの開発に成功した。

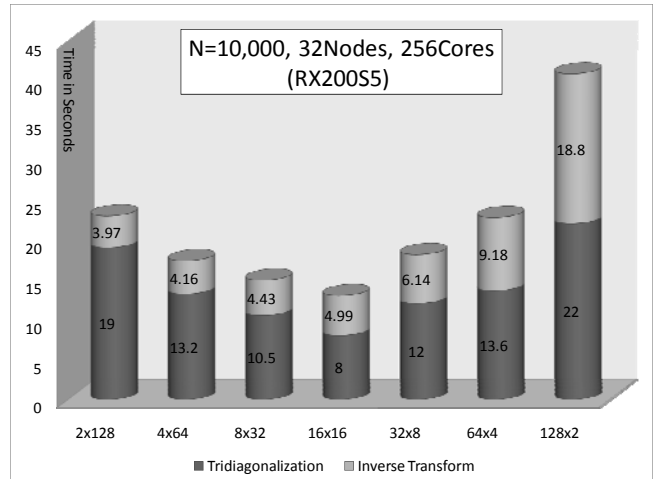
本課題の三重対角アルゴリズムにおいては、必要なメモリ量が p に応じて削減するが、 p が増えると通信時間が増す。一方、三重対角化 (Step1) においては、必要なメモリ量が p に応じて削減するが、 $p=q$ のとき最も通信時間が短い。したがってソルバ全体として、利用メモリ量と p, q の値が影響する全体の実行速度について、現在明確な指針が示されていない。自動チューニング機構を導入する場合、チューニング戦略が立てられないという現状があり、本利用による評価により、その解決策を明らかにする。

3. 結果

図 1 に実行時間、表 1 に速度とメモリ量に関する尺度での評価値を示す。



(a) 128 コア。 $p \neq q$ の場合。



(b) 256 コア。 $p=q$ の場合。

図 1 理研 Fujitsu PRIMERGY RX200S5 における異なるプロセッサ・グリッドでの実行時間。

図 1 から、この計算機環境では三重対角化の時間が逆変換の時間に比べて多く、三重対角化を最適化するチューニング戦略をとるべきであることが明らかとなった。

表 1 中の SPM とは、 $(p=q \text{ の実行時間}) / (\text{対象の実行時間}) / (\text{対象の } p=q \text{ に対するメモリ量の増加})$ で定義される。したがって、もし $p=q$ に対しメモリ量が 2 倍になっても、実行時間が 1/2 になれば SPM=1 であり、これは効率が良いことをさす。SPM < 1 の時、速度とメモリ量の関連から、効率が悪いことをさす。

表 1 (a) から、SPM の基準について、0.95 倍の速度低下でメモリ量が 1/2 になる点において、正方に近いグリッド 8x16 での実行は、4x32 での実行に対して効率的ではないことが明らかになった。また、表 2 (b) につい

ては、正方グリッド 16x16 の実行について、0.71 倍の速度低下を許容できるのであれば、1/2 のメモリ量削減が可能となる。

正方に近いグリッドを取るかとらないかの判断は、ユーザが実行速度を優先するのか、メモリ量削減を優先するのかに依存するので、一概に判断できない。したがって、ユーザの指針（ポリシー）を自動チューニング機能として用意する必要があることが明らかとなった。

表 1 理研 Fujitsu PRIMERGY RX200S5 における速度向上と必要メモリ量。メモリ量は、Householder ベクトル u_k を保持するための要求メモリ量について計算している。

(a) 128 コア。 $p \neq q$ の場合。

Grid ($p \times q$)	実行時間 [秒]	速度向上	メモリ量	SPM 値
8x16	25.1	1x	1x	1
4x32	26.1	0.96x	2x	0.48
2x64	30.7	0.81x	4x	0.20
16x8	26.3	0.95x	0.5x	1.9
32x4	28.3	0.88x	0.25x	3.5
64x2	38.3	0.65x	0.125x	5.2

(b) 1024 コア。 $p = q$ の場合。

Grid ($p \times q$)	実行時間 [秒]	速度向上	メモリ量	SPM 値
16x16	12.9	1x	1x	1
32x8	18.1	0.71x	0.5x	1.4
64x4	22.7	0.56x	0.25x	2.2
128x2	40.8	0.31x	0.125x	2.4

4. まとめ

本課題では、計算機環境に依存し、メモリ量と演算速度が変化するアプリケーションとして対称実数密行列用の固有値ソルバを例に挙げ、性能評価を理研 Fujitsu PRIMERGY RX200S5 を用いて行った。性能評価の結果、高々0.95 倍の速度低下で 1/2 のメモリ量削減が可能である例を発見した。また、このような速度とメモリ量に影響するチューニングを行う場合において、自動チューニング機能にユーザによるポリシーの設定機

能の必要性があることを示した。

5. 今後の計画・展望

本課題による性能評価により、提案したプロセッサ・グリッド自在な新アルゴリズムの有効性の一例が示されたが、他機種における性能評価において、以下の特徴が発見された。

- 超並列実行時もしくは小規模行列実行時において、提案するプロセッサ・グリッド自在アルゴリズムを利用すると、利用メモリ量を増加させることで逆変換時間を三重対角化時間に対し、ほとんど無視できる時間まで削減可能である。

この事実は、従来の固有値ソルバの演算方針を変換できる可能性がある興味深いものである。また、自動チューニング機能の構成法と連結し、新規アルゴリズムの性能評価を行うべきであることも示唆している。今後の計画として、この評価を中心に行いたい。

6. RICC の継続利用を希望の場合は、これまで利用した状況（どの程度研究が進んだか、研究においてどこまで計算出来て、何が出来ていないか）や、継続して利用する際に行う具体的な内容

継続申請を希望する。以下を中心に性能評価を行う。

- 小規模行列・高並列実行時において、提案の逆変換アルゴリズムにおける実行時間の評価
- ユーザがメモリ量と実行時間のポリシーを与えた時における、自動チューニング機能の戦略と、その品質の評価
- 対称密行列ソルバ以外の数値計算ライブラリの評価。たとえば、疎行列反復解法ソルバの性能評価と、その自動チューニング機能の品質評価

以上

平成 21 年度 RICC 利用研究成果リスト

【国際会議などの予稿集、proceeding】

Takahiro Katagiri and Shoji Itoh: A Massively Parallel Dense Symmetric Eigensolver with Communication Splitting Multicasting Algorithm, Proceeding of 9th International Meeting of High performance Computing for Computational Science (VECPAR' 10), Berkeley, California, USA, June 22-25 (2010) 採録決定

以上