

量子i.i.d.状態のシミュレーションと その理論的考察

電気通信大学 大学院情報システム学研究科
情報ネットワークシステム学専攻

坂下 達哉(博士後期課程)

スパコン若手利用者推薦報告会 於 東京大学武田先端知ホール

2010.5.21(金)

研究概要

背景

- 量子情報理論を理解するうえで、量子i.i.d.状態に関する極限式が重要である
- 量子i.i.d.状態のシミュレーションは計算時間の観点から、従来、不可能であると考えられてきた
- 数学の表現論におけるテンソル積の既約分解を用いたアルゴリズムが長岡[1]によって提案された
これを用いると、指数的な問題が多項式オーダーに変わる

目的

- 既に極限值が分かっている問題の数値的特性の理解を深める
- 未知の問題の理論式予測に活用できるようにする

量子力学の基礎

$$\rho \text{ は量子状態} \iff \rho \geq 0, \text{Tr} \rho = 1$$

(密度行列) def (半正定値)

$$M = \{M_x\}_{x \in X} \iff M_j \geq 0 (\forall x \in X)$$

は測定 def (半正定値)

$$\sum_{x \in X} M_j = I$$

状態 ρ のもとで測定 M を行う

$$P_\rho^M(x) := \text{Tr}[\rho M_x] \quad \text{測定値 } x \text{ が得られる確率}$$

数学的準備

A : エルミート行列 $d := \dim A$
 $\{\lambda_1, \dots, \lambda_d\}$: A の固有値

$\{\psi_1, \dots, \psi_d\}$: A の固有ベクトル

$A = \sum_{i=1}^d \lambda_i \psi_i \psi_i^*$ A のスペクトル分解
(ここで、 $*$ は共役転置)

$\{A > 0\} := \sum_{i: \lambda_i > 0} \psi_i \psi_i^*$ A の正部分の台への射影子

つまり A の正固有値に対応する固有ベクトル空間の直和への射影

$\{A > 0\}, \{A \leq 0\}$ の組は測定である

A と B のテンソル積

$$A = [a_{ij}] \quad d := \dim A$$

$$A \otimes B := \begin{bmatrix} a_{11}B & \cdots & a_{1d}B \\ \vdots & \ddots & \vdots \\ a_{d1}B & \cdots & a_{dd}B \end{bmatrix}$$

$$\dim A \otimes B = \dim A \cdot \dim B$$

$$\dim A^{\otimes n} = \dim A \otimes \cdots \otimes A = d^n$$

ρ : 量子状態 $\rightarrow \rho \otimes \rho$ は量子状態である

量子仮説検定(2状態識別)

ρ, σ : 2×2 の密度行列とする

本発表では、密度行列は 2×2 サイズのものしか扱わない

2つの 2×2 の密度行列に対しては、実対称行列に限って良い

ρ or σ : given

与えられた状態がどちらかを判別する問題を考える

1つだけでは判別しづらいので、 n 個独立に同じ状態を用意する

量子i.i.d.状態

数学的: n 個のテンソル積

$$\rho^{\otimes n} = \rho \otimes \rho \otimes \cdots \otimes \rho$$

物理的: 各成分の間に相関がない

結局、問題はこうなる: $\rho^{\otimes n}$ vs. $\sigma^{\otimes n}$

$a \in \mathbb{R}$: 閾値 (パラメータ)

$\alpha_n(a) := \text{Tr}[\rho^{\otimes n} \{\rho^{\otimes n} - e^{na} \sigma^{\otimes n} \leq 0\}]$: 第1種誤り確率

真の状態は $\rho^{\otimes n}$ だが、誤って $\sigma^{\otimes n}$ と判断する確率

$\beta_n(a) := \text{Tr}[\sigma^{\otimes n} \{\rho^{\otimes n} - e^{na} \sigma^{\otimes n} > 0\}]$: 第2種誤り確率

真の状態は $\sigma^{\otimes n}$ だが、誤って $\rho^{\otimes n}$ と判断する確率

ρ と σ を入れ替えれば同様なので、以降は $\beta_n(a)$ にのみ着目する

$$r_n(a) := -\frac{1}{n} \log \beta_n(a) \quad \beta_n(a) \text{ の第一指数部}$$

$$r(a) := \lim_{n \rightarrow \infty} r_n(a) \quad \text{第2種誤りレート}$$

Hoeffdingの定理(長岡[2],林 [3]etc.2006)

$$r(a) = \max_{\theta \in \mathbb{R}} \{ \theta a - \psi(\theta) \} \quad (\text{以下、極限值と呼ぶ})$$

$$\text{ただし、} \psi(\theta) = \log \text{Tr}[\rho^\theta \sigma^{1-\theta}]$$

ρ, σ は1組固定する

数値計算により、 $(a, r_n(a))$ のグラフを描く

テンソル次数 n を大きくして繰り返し、極限值 $r(a)$

への近づき方を見る

$\beta_n(a)$ の計算方法

$$\{\rho^{\otimes n} - e^{na} \sigma^{\otimes n} > 0\}$$

基底変換 \longrightarrow

$$\left[\begin{array}{ccccccc} \{\rho_0 - e^{na} \sigma_0 > 0\} & & & & & & 0 \\ & \{\rho_1 - e^{na} \sigma_1 > 0\} & & & & & \\ & & \{\rho_1 - e^{na} \sigma_1 > 0\} & & & & \\ & & & \ddots & & & \\ 0 & & & & & & \{\rho_{\lfloor \frac{n}{2} \rfloor} - e^{na} \sigma_{\lfloor \frac{n}{2} \rfloor} > 0\} \end{array} \right]$$

従って、以下のように、 $\beta_n(a)$ は既約成分ごとに計算できる

$$\begin{aligned} \beta_n(a) &= \text{Tr}[\sigma^{\otimes n} \{\rho^{\otimes n} - e^{na} \sigma^{\otimes n} > 0\}] \\ &= \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n}{k} \binom{n}{k-1} \text{Tr}[\sigma_k \{\rho_k - e^{na} \sigma_k > 0\}] \end{aligned}$$

正部分の台への射影子の計算方法

$\rho_k - e^{na} \sigma_k$ は実対称行列ということしかわからない

よって、固有値 $\{\lambda_i\}$ 、固有ベクトル $\{\psi_i\}$ を数値的に計算する

$$\{\rho^{\otimes n} - e^{na} \sigma^{\otimes n} > 0\} = \sum_{i:\lambda_i > 0} \psi_i \psi_i^*$$

このためには

- 全固有値、全固有ベクトルが必要
- 固有値の正負を判断するために、絶対値の小さな固有値も正確に求めたい

以上の要求から、Jacobi法やQR法が適切

Jacobi法とQR法の共通利点

- 回転行列を掛け合わせて固有ベクトルを求める
よって、固有ベクトルの直交性が非常に良い

固有値解法:(2)Householder QR法

Householder法

QR法

A : 実対称行列 \rightarrow A_0 : 3重対角対称行列 \rightarrow 対角行列

QR法

$$\begin{array}{ll} A_0 = Q_0 R_0 & \text{とQR分解} \\ A_1 = Q_1 R_1 & \\ \vdots & \\ A_k = Q_k R_k & \end{array} \quad \begin{array}{l} A_1 = R_0 Q_0 \\ A_2 = R_1 Q_1 \\ \vdots \\ A_{k+1} = R_k Q_k \end{array}$$

ここで、 Q_i : ユニタリ行列 R_i : 上三角行列

A_k は対角行列に近づいていく

固有値の収束判定

ϵ : あらかじめ決めておく、小さな値

ϵ よりも小さくなったら、非対角成分が 0 になったと判断して計算を打ち切る

Jacobi法の場合

• 非対角成分の最大絶対値

$$\max_{i \neq j} |a_{ij}| < \epsilon$$

であれば終了

Householder QR法の場合

• 最右下の非対角成分 $<$ 最右下2つの対角成分の和 $\times \epsilon$

であれば、最右下隅は収束したと判断して切り離し、行列のサイズを1減らす(減次)

• 行列のサイズが2より小さくなれば終了

固有値解法の比較

- 計算速度 Jacobi法 < Householder QR法

このため、QR法の方が標準的に用いられる

QR法を高速化する手法

原点移動: 二次方程式の解法を使って、右下の
 2×2 行列の固有値の近似値を得る

減次

- 精度 Jacobi法 > Householder QR法

LAPACKの限界、多倍長演算の必要性

固有値が指数的に減少していく悪条件の行列を扱う
そのため、LAPACK(QR法)では $n = 200$ 程度までしか
誤差なく計算できない

よって、多倍長演算が必要になる

線形代数ライブラリ・ 並列化について

線形代数ライブラリ

Eigen2 (C++テンプレートライブラリ)

- ベクトル化に対応  BLAS, ATLASに匹敵する性能！
- MATLABなどに似た行列演算の表記が可能
(例) `MatrixXd a(3,3), b(3,3);` 倍精度型の 3×3 行列 a, b を宣言
`MatrixXd c = a*b;` 行列 a と行列 b の乗算
`MatrixXd d=a.part<LowerTriangular>().square().sum();`
aの下三角部分の2乗和を求める
- 範囲外の添字によるアクセスを確実に検出  デバッグが容易に！

多倍長ライブラリ `exflib` (並列化対応した多倍長ライブラリ)

この2つを組み合わせ、多倍長型を成分に持つ行列を扱えるようにした。多倍長型の行列でもベクトル化の効果が現れるのかは疑問

並列化

- 各既約成分に対する処理は独立

➡ MPIによるプロセス単位の並列化

- 各プロセスが担当する既約成分たちはいつも同じにする

➡ $\rho^{\otimes n}$ と $\sigma^{\otimes n}$ の既約成分は一度だけ計算すればよい

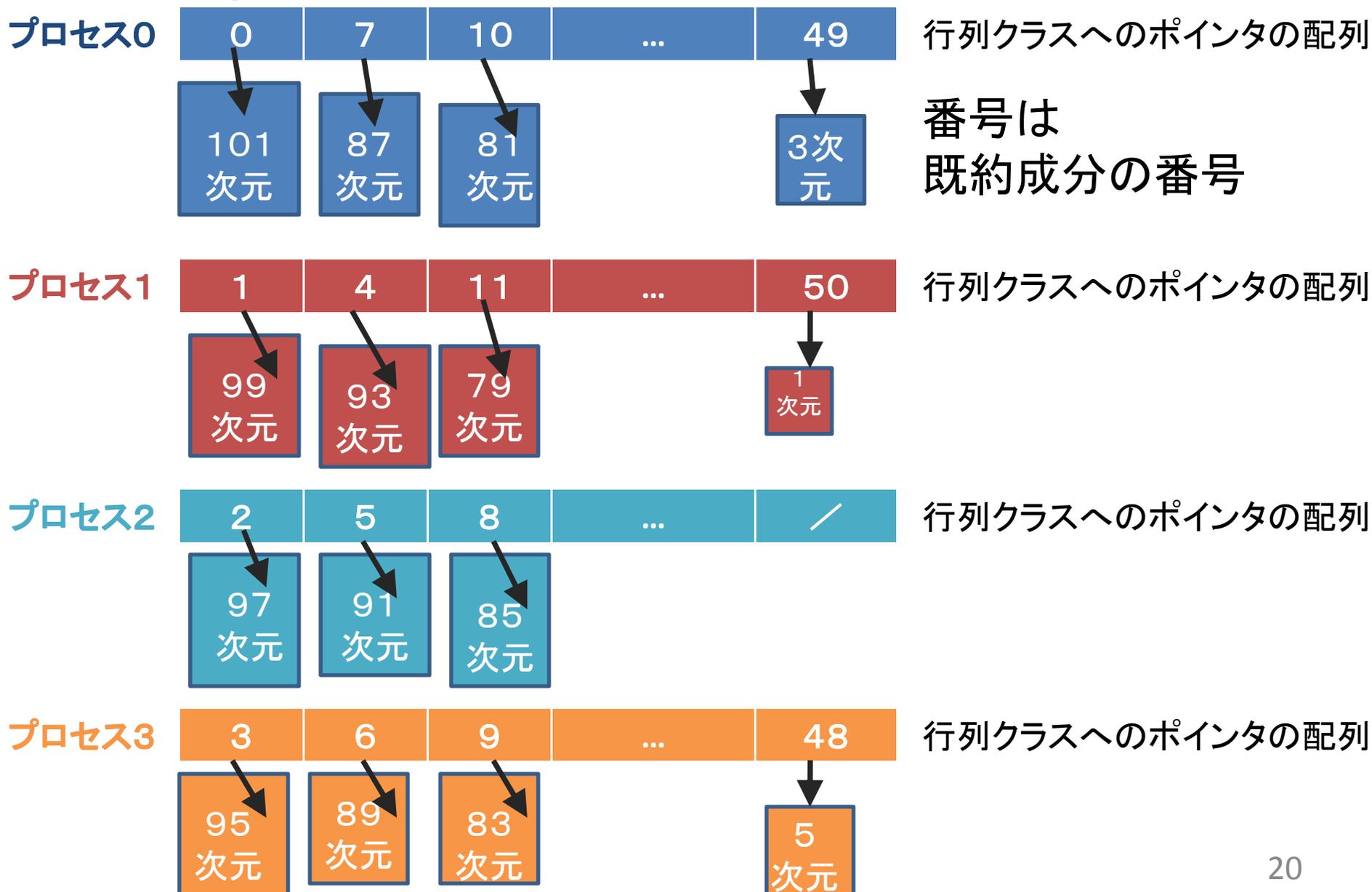
- メモリ使用量・計算量ともに均一になるように、担当する既約成分を決める

(例)テンソル次数 $n = 100$ の場合

既約成分の番号 $k = 0, 1, \dots, \lfloor \frac{n}{2} \rfloor = 50$

対応する既約成分の次元 = $101, 99, \dots, 3, 1$

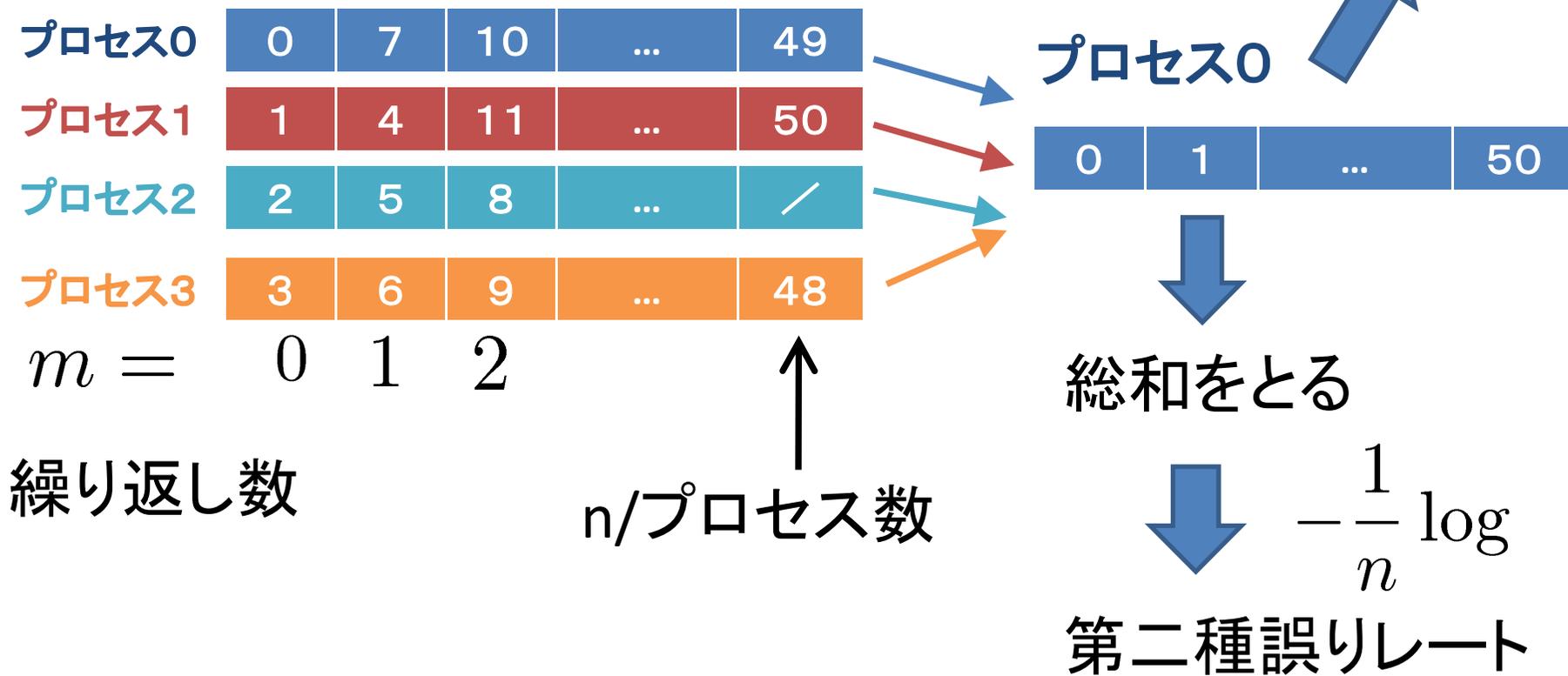
並列化: $\rho^{\otimes n}$ と $\sigma^{\otimes n}$ の既約成分の格納方式



並列化：計算結果の集約

閾値 a を変えて、以下を繰り返す

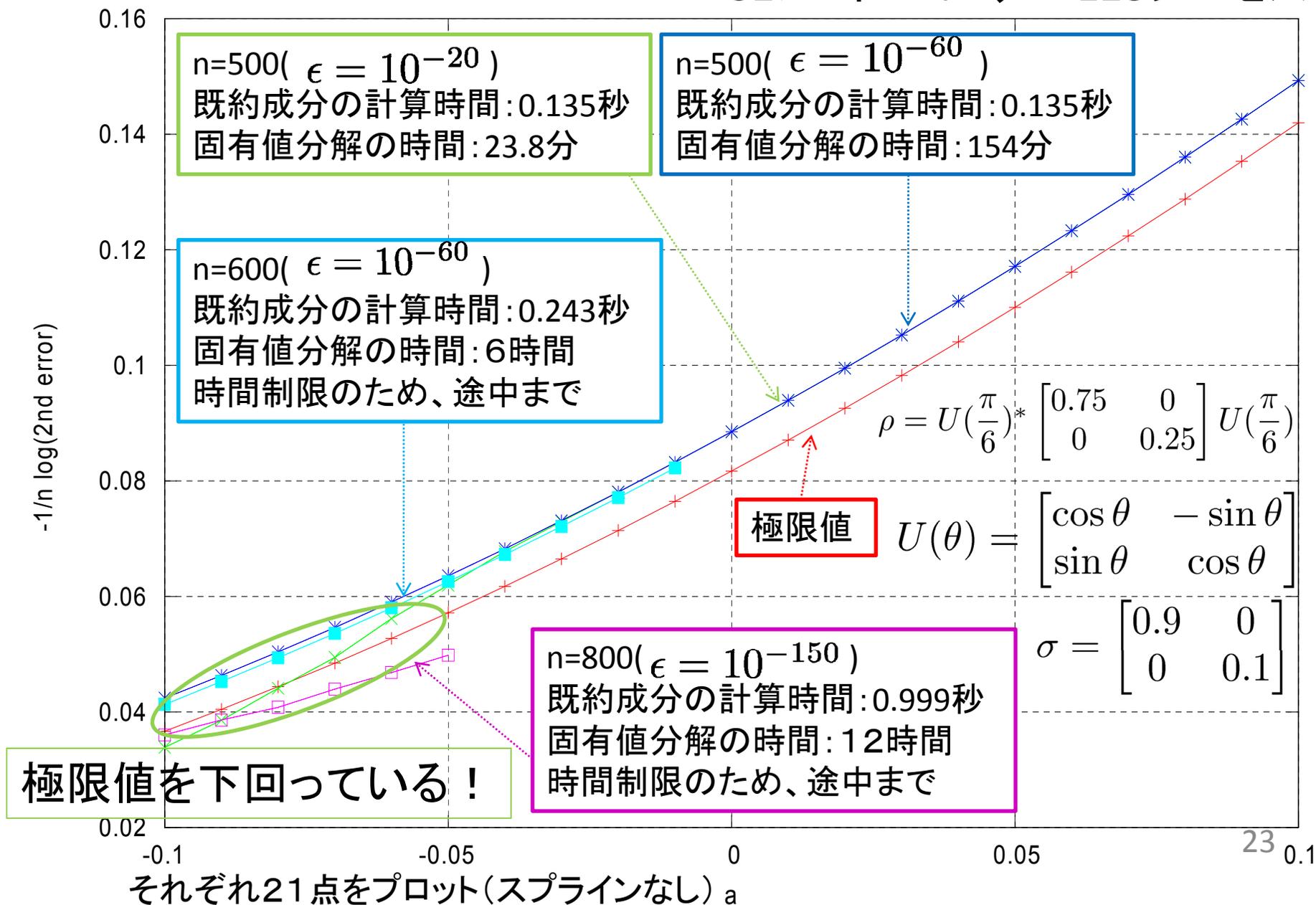
既約成分ごとに誤り確率の計算 MPI_Gather



誤り確率の計算への適用結果

Jacobi法 (倍精度)

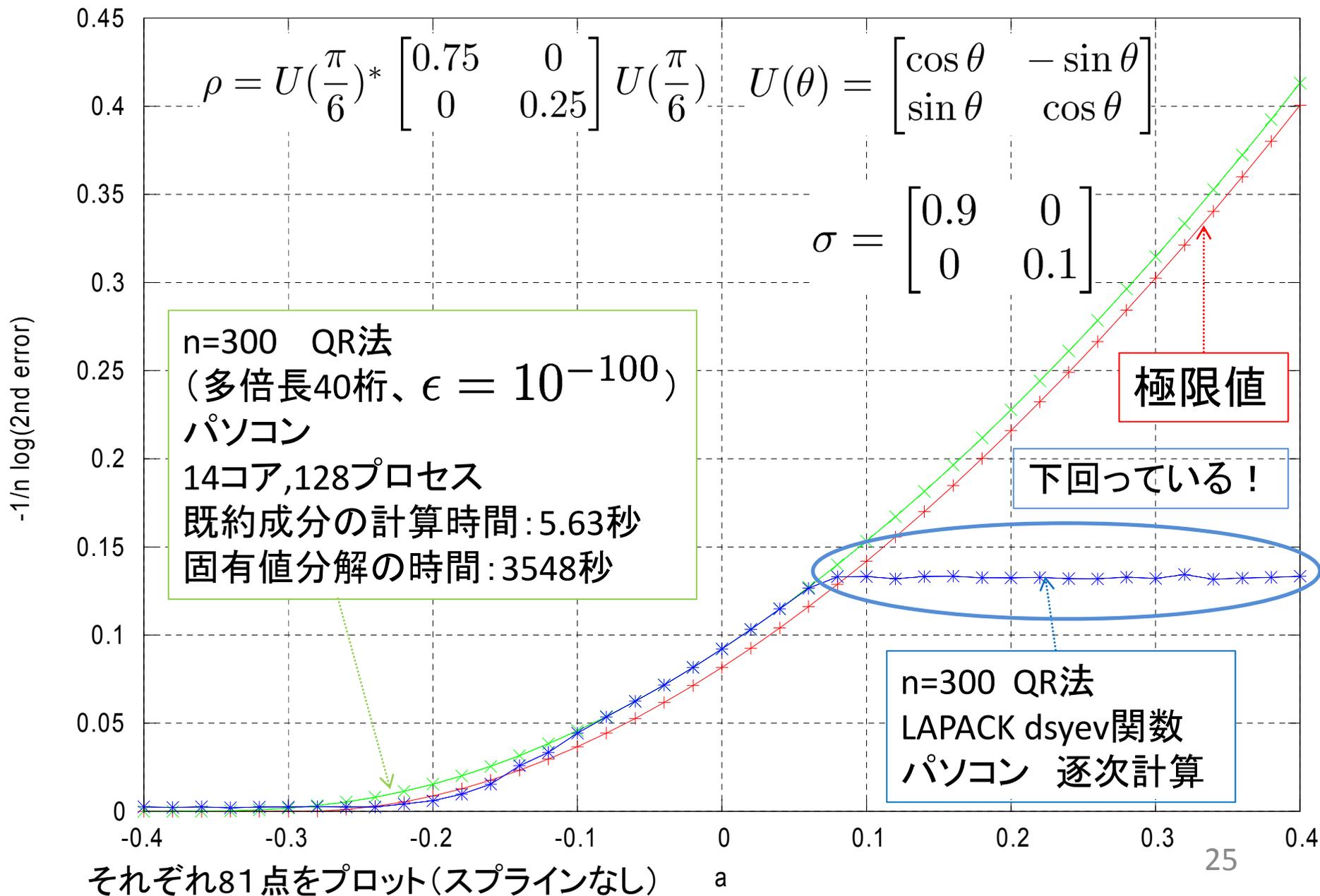
東大HA8000による
32ノード×4コア=128プロセス



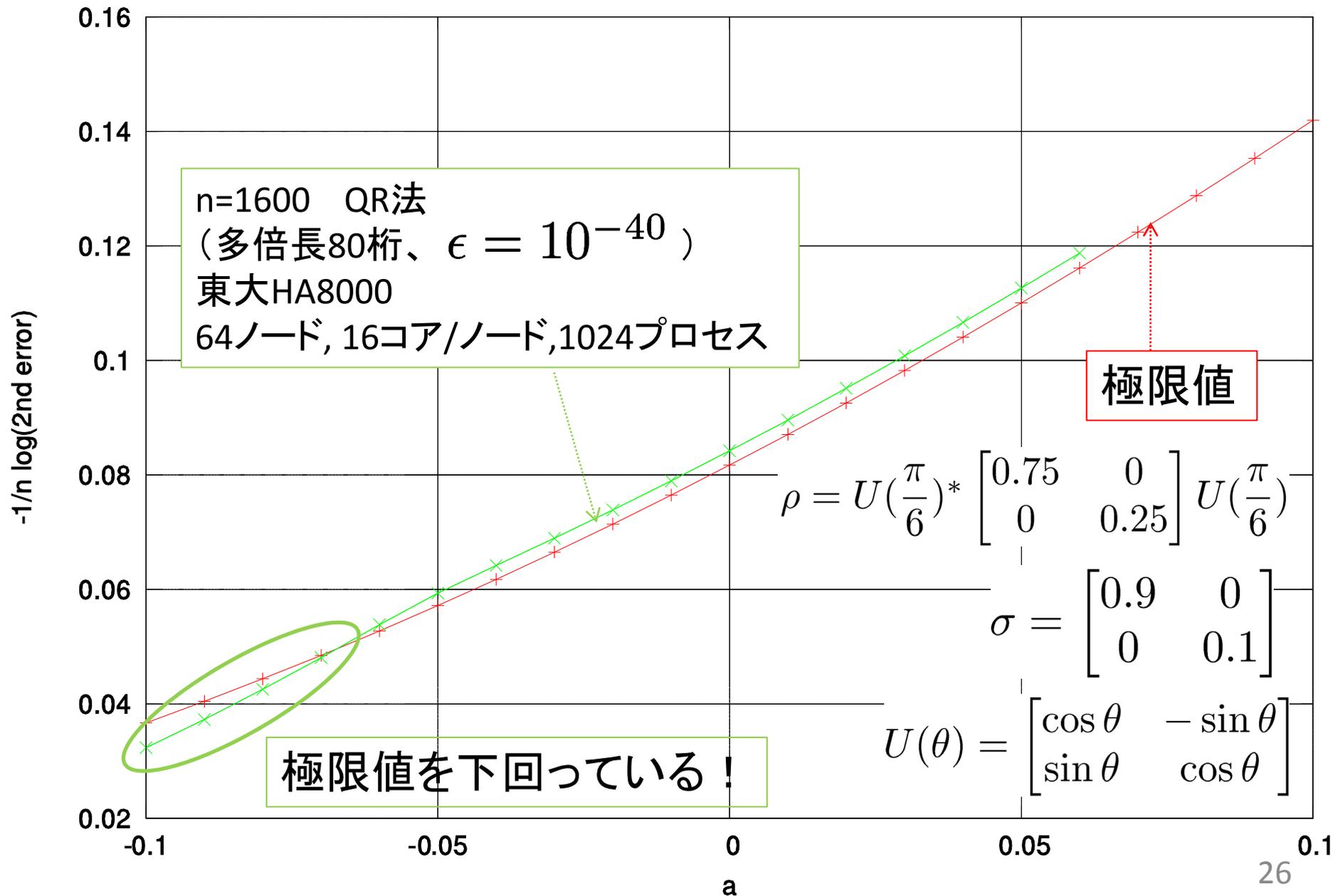
Jacobi法による結果

- n が大きくなると、極限值に上から近づくことがわかっていて、
よって、下回っている箇所は数値誤差が原因である
下回る現象は ϵ が甘いところで起こっている
- 誤差なく計算するには、 ϵ をテンソル次数に応じて変える必要がある
- Jacobi法は計算時間・誤差の観点から見て、倍精度では $n = 600$ くらいが限界
- Jacobi法を多倍長で行うと、大きな n でも正確な結果が出るが、収束に時間がかかるので用いない

QR法 テンソル次数 $n = 300$



QR法 テンソル次数 $n = 1600$



QR法による結果

- 多倍長のQR法を用いると、 $n=300$ 程度はパソコンで計算できる
- 倍精度では、Jacobi法の方が時間はかかるが精度は良い

- スパコンを用いると、 $n=1600$ 程度まで計算できる
- $n=1600$ のとき、桁数が少ないため、極限值を下回る誤差が生じている

- $n=2000$ で、使用メモリが限度の28GBを超えてしまう
現状の対策：既約成分の保存をやめる
1ノードあたりの使用コア数を減らす

まとめ

- 既約成分の計算はすでに高速化した
- ほとんどの計算時間は固有値分解が占める
- Jacobi法は繰り返し回数が多い($n=600$ が限界)
そのため、特に n が大きい時はHouseholder QR法で計算したい
- n が大きくなるほど、Householder QR法のうち三重対角化の占める時間が多くなる
- $n=2000$ で、使用メモリが限度の28GBを超えてしまう

今後の課題

- 誤差の詳細な考察
- できるだけ少ない多倍長桁数で誤差なく計算を行えるようにする
- 三重対角化の単体高速化・スレッド並列化
- 全体でスレッド並列化も加えてハイブリッド並列化
- 他の固有値解法の適用

参考文献

1. Hiroshi Nagaoka, Masahito Hayashi “An Information-Spectrum Approach to Classical and Quantum Hypothesis Testing for Simple Hypotheses”, IEEE Trans. (2007)
2. Hiroshi Nagaoka “The Converse Part of The Theorem for Quantum Hoeffding Bound” quant-ph/0611289v1 (2006)
3. Masahito Hayashi “Error exponent in asymmetric quantum hypothesis testing and its application to classical-quantum channel coding”, Phys.Rev.A (2007)
4. 柿崎晃 修士論文「量子的i.i.d.情報源に関する数値計算によるアプローチ」
5. 堂島隆幸 修士論文「量子i.i.d.状態に関する仮説検定の漸近特性について」
6. 長岡浩司 講義ノート for 堂嶋氏
7. 小川朋宏 博士論文「量子力学系における仮説検定と通信路符号化の漸近特性に関する研究」
8. 杉浦光夫 「連続群論入門」
9. 片桐孝洋 「高性能プログラミング(I)入門編」
10. P.パチエコ 「MPI並列プログラミング」
11. W.Groop, E.Lusk & A.Skjellum “Using MPI 2nd edition” (1999)
12. 有本卓「数値解析(1)」
13. 森正武「数値解析」 共立出版
14. 森正武「数値解析法」 朝倉書店
15. 奥村晴彦「C言語による最新アルゴリズム事典」, 技術評論社 (1991)
16. 戸川隼人「マトリックスの数値計算」, オーム社 (1971)
17. G.Golub, C.Loan ”Matrix Computation 3rd Editon”
18. B.Parlett “The Symmetric Eigenvalue Problem”, SIAM, Philadelphia (1998)

19. Y.Saad, Numerical methods for large eigenvalue problems, Manchester University Press (1992)
20. N.Higham, Accuracy and Stability of Numerical Algorithms, SIAM, Philadelphia (1996)
21. J.Wilkinson, The algebraic eigenvalue problem, Clarendon Press (1965)
22. H.Park, V.Hari “A Real Algorithm for the Hermitian eigenvalue decomposition”, BIT 33, 158-171 (1993)
23. E.Anderson, “Discontinuous plane rotation and the symmetric eigenvalue problem”, LAPACK Working Note 150 (2000)
24. D.Bindel, J.Demmel, W.Kahan, and O.Marques, “On computing givens rotations reliably and efficiently”, LAPACK Working Note 148
<http://www.netlib.org/lapack/lawns/downloads/>
25. 山本有作 「密行列固有値解法の最近の発展 (I) - Multiple Relatively Robust Representations アルゴリズム -」, 日本応用数学会論文誌(2005)
26. 山本有作 「密行列固有値解法の最近の発展 (II) - Multiple Relatively Robust Representations アルゴリズム -」, 日本応用数学会論文誌(2006)
27. 藤原宏志 並列化対応多倍長ライブラリExflib
<http://www-an.acs.i.kyoto-u.ac.jp/~fujiwara/exflib/>
28. 線形代数計算用C++テンプレートライブラリEigen2
http://eigen.tuxfamily.org/index.php?title=Main_Page