

# 量子 i.i.d. 状態のシミュレーションとその理論的考察

坂下達哉

電気通信大学情報システム学研究所

## 1 はじめに

量子情報理論を理解する上で、量子 i.i.d. 状態に関する極限式は重要である。これまで、古典情報理論において漸近性を扱うには数値的手法が活用されていたが、量子情報理論においては実現可能で有効であると考えられていなかった。

テンソル積の既約分解という手法は林 [5] により初めて量子 i.i.d. 状態に適用され、長岡 [8] により数値計算アルゴリズムが提案された。これにより、計算時間を大幅に削減できるようになった。その後、長岡の指導のもとで、柿崎 [11]、堂嶋 [13] により研究された。

これらの研究を受けて、我々は、既知の量子仮説検定問題を題材として、大規模で高精度の数値シミュレーションが可能であることを実証した [9, 12]。本研究の目的は、既知の問題を解析し、未解決の問題の極限式予測に活用できるようにすることである。例えば、[9] では、中心極限定理に関連した新しい予想を数値的に検証している。本稿では、既知の量子仮説検定問題のシミュレーションについて速度と精度の面から改善を行った結果を実装に重点をおいて報告する。

## 2 量子仮説検定

本節では量子仮説検定について必要最小限の事柄を述べる。まず、そのために必要な概念をいくつか導入する。

行列  $A$  ( $d \times d$  サイズ) と  $B$  のテンソル積は

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

と定義される。

密度行列とは正定値でトレースが 1 の行列のことである。密度行列は量子状態の一般の表現と考えられている。 $\rho$  を量子状態としたときに、量子 i.i.d. (independent and identically distributed) 状態とは数学的には  $n$  次のテンソル積

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

と定義され、物理的には独立に用意された  $n$  個の状態を意味している。ここで、 $n$  はテンソル次数とよばれる。量子 i.i.d. 状態は古典確率論における独立同一分布に対応する。

以上で必要な概念の準備は整った。いま、 $2 \times 2$  サイズの密度行列（量子状態の行列表現） $\rho, \sigma$  が与えられているとし、真の量子状態に関する単純仮説  $\rho^{\otimes n}$  v.s.  $\sigma^{\otimes n}$  についての以下の

ような仮説検定問題を考える。実数  $a$  に対して、次の量

$$\beta_n(a) := \text{Tr}[\sigma^{\otimes n} \{\rho^{\otimes n} - e^{na} \sigma^{\otimes n} > 0\}] \quad (1)$$

の振る舞いを調べる。ここで、エルミート行列  $A$  に対して、 $\{A > 0\}$  は  $A$  の正固有値に対応する固有空間の直和への射影を表す。すなわち、 $d$  次元エルミート行列  $A$  の固有値を  $\{\lambda_1, \dots, \lambda_d\}$ , 固有ベクトルを  $\{\psi_1, \dots, \psi_d\}$  とし、

$$A = \sum_{i=1}^d \lambda_i \psi_i \psi_i^* \quad (\text{ここで, } * \text{ は共役転置を表す})$$

とスペクトル分解したときに、

$$\{A > 0\} := \sum_{i: \lambda_i > 0} \psi_i \psi_i^*$$

と定義する。

(1) は量子 Neyman-Pearson 検定  $\{\rho^{\otimes n} - e^{na} \sigma^{\otimes n} > 0\}$  の二つの誤り確率のうちの一つであり、 $\sigma^{\otimes n}$  が真のときに  $\rho^{\otimes n}$  を誤って真と判断する確率を意味する [8]。また、

$$r_n(a) := -\frac{1}{n} \log \beta_n(a), \quad (2)$$

$$r(a) := \lim_{n \rightarrow \infty} r_n(a) \quad (3)$$

とすると、量子 Hoeffding の定理 [6, 7] より、任意の  $a > -D(\sigma \parallel \rho)$  に対して、

$$r(a) = \max_{\theta \in \mathbb{R}} \{\theta a - \psi(\theta)\} \quad (4)$$

$$\text{ここで, } \psi(\theta) = \log \text{Tr}[\rho^\theta \sigma^{1-\theta}]$$

が成り立つ。(4) は Newton 法などを用いることで求められ、計算する行列は  $2 \times 2$  サイズなので計算時間はかからない。本稿の目的は十分大きな  $n$  について十分な精度で数値的に  $\beta_n(a)$  と  $r_n(a)$  を計算し、 $r_n(a)$  が極限值 (4) に近づいていく様子を観察することである。

### 3 既約成分の計算アルゴリズム

次に、 $2 \times 2$  行列のテンソル積の既約成分の計算アルゴリズムを記述する。前節で定義した量子 i.i.d. 状態  $\rho^{\otimes n}$  のサイズは  $2^n$  になるため、この行列成分の計算は大変である。そこで、次に導入するテンソル積の既約分解という方法を用いる。

任意の  $2 \times 2$  行列

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad (5)$$

に対し適当な基底変換を施すことで、 $A^{\otimes n}$  は

$$\bigoplus_{k=0}^{\lfloor n/2 \rfloor} \bigoplus_{m_k} A_k \quad (6)$$

という形にブロック対角行列で表される [4]。各  $A_k$  はテンソル積  $A^{\otimes n}$  の既約成分と呼ばれ、そのサイズは  $\dim A_k = n + 1 - 2k$  で、重複度  $m_k = {}_n C_k - {}_n C_{k-1}$  回だけ現れる。ここで、便宜的に  ${}_n C_{-1} := 0$  とする。各  $A_k$  の行列成分  $[\alpha_{ij}]$  は

$$(a_{11}y + a_{21}z)^{k-j}(a_{12}y + a_{22}z)^j = \sum_{i=0}^k \alpha_{ij}^0 y^{k-i} z^i \quad (7)$$

と多項式の係数  $[\alpha_{ij}^0]$  を求め、

$$\alpha_{ij} = (\det A)^{\frac{n-k}{2}} \frac{\sqrt{{}_k C_j}}{\sqrt{{}_k C_i}} \alpha_{ij}^0 \quad (8)$$

によって計算できる。ここで、肝要なことは既約分解 (6) に用いる基底変換が  $A$  の行列成分によらないことである。よって、テンソル積の加算・減算は既約成分ごとに行える。また、後でみるようにトレースをとるので、この基底変換は存在することは重要であるが具体的に求める必要はない。

このように、既約分解を用いることで、指数サイズであるテンソル積の行列成分の計算が多項式オーダーで行える。

## 4 既約分解を用いた誤り確率の計算

本稿では、既約分解を用いた誤り確率の計算法を与える。

既約分解 (6) を用いることによって、誤り確率  $\beta_n(a)$  は

$$\sum_{k=0}^{\lfloor n/2 \rfloor} ({}_n C_k - {}_n C_{k-1}) \text{Tr}[\sigma_k \{\rho_k - e^{na} \sigma_k > 0\}] \quad (9)$$

と表される。ここで、 $\{\rho_k\}$  と  $\{\sigma_k\}$  はそれぞれ  $\rho^{\otimes n}$  と  $\sigma^{\otimes n}$  の既約成分である。

$\beta_n(a)$  の計算手順は以下ようになる。

1. (7) と (8) を用いて  $\{\rho_k\}$  と  $\{\sigma_k\}$  を計算する。
2. 各  $k$  について、実対称行列  $\rho_k - e^{na} \sigma_k$  の固有値問題を解き、正固有値に対応する正規直交化された固有ベクトルの集合  $\{v_{n,k,i}\}$  を求める。
- 3.

$$\beta_n(a) = \sum_k ({}_n C_k - {}_n C_{k-1}) \sum_i v_{n,k,i}^* \sigma_k v_{n,k,i}$$

を求める。ここで、 $v_{n,k,i}^*$  は  $v_{n,k,i}$  の共役転置を表す。

$2 \times 2$  サイズの2つの密度行列については、実対称行列に限って良いことが知られているので、実際の計算では  $\rho, \sigma$  は実対称行列のみを用いる。したがって、固有値分解にも実対称行列向けのルーチンを用いる。固有値ルーチンには以下のような要求がある。

- (a) 全固有値、全固有ベクトルが必要である。

- (b) テンソル積  $\rho^{\otimes n}, \sigma^{\otimes n}$  は指数的に減少する固有値を持つ。したがって、実対称行列  $\rho_k - e^{na} \sigma_k$  には絶対値の小さい正負の固有値があるが、固有値の正負が数値誤差によって変わることがないように正確に求める必要がある。
- (c) 計算時間が短くて済む。

Jacobi 法は (a),(b) を満たし、QR 法は (a),(c) を満たす [2, 3]。Jacobi 法は現実的な時間内で計算が終了しないので、我々は QR 法を主に用い、(b) の精度の問題は多倍長化することで補うことにする。

7 節では、 $\rho, \sigma$  を固定し、関数  $\beta_n(a)$  のグラフを描画する。なお、 $\rho^{\otimes n}$  と  $\sigma^{\otimes n}$  の既約成分  $\{\rho_k\}$  と  $\{\sigma_k\}$  は全ての  $a$  について共通なので、一度だけ計算すればよい。

## 5 計算機環境

使用した計算機環境は以下の通りである。

- 計算機：スーパーコンピュータ 東京大学 HA8000 (AMD Opteron processor 8356), Power Edge T410 (Intel Xeon)
- コンパイラ：Intel C++ compiler
- C++ テンプレート線形代数ライブラリ：Eigen2[1]
- 多倍長演算ライブラリ：Exfib[10]
- 固有値分解に用いるライブラリ：LAPACK dsyev 関数 (倍精度・QR 法の場合), Eigen2 の固有値分解関数を改変 (多倍長・QR 法の場合)

工夫した点としては、Eigen2 に多倍長型を追加し、多倍長型を成分に持つ行列を扱えるようにしたことがあげられる。これにより、倍精度に限らず多倍長も MATLAB などと同様に行列演算を記述できる。また、倍精度・多倍長の切り替えも、行列クラス名の置換のみで済む。なお、日立のコンパイラも試したかったが、ライブラリ Eigen2 が対応していないようなので、Intel のコンパイラを用いた。

## 6 並列化方法

本節では、並列化方法について記述する。

まず、異なる  $k$  について、既約成分  $\rho_k$  および  $\sigma_k$  の行列成分の計算は互いに独立しているので、自明な並列化が行える。さらに、各  $a$  に対して、(9) において  $\beta_n(a)$  の第  $k$  項の計算は、 $\rho_k - e^{na} \sigma_k$  の固有値分解も含めて異なる  $k$  について並列化できる。各既約成分の格納には膨大なメモリが必要で、データの送受信が少ないので、並列化には分散メモリ向けの MPI を用いた。各プロセスのメモリ使用量・計算量ができるだけ均一にするため、担当する既約成分たちは巡回的に割り当てておく。また、各  $a$  に対して担当する既約成分たちは固定する。

例として、テンソル次数  $n = 100$  の場合をとりあげる。このとき、既約成分の番号  $k$  は  $0, 1, \dots, \lfloor \frac{n}{2} \rfloor = 50$  で、対応する既約成分の次元は  $101, 99, \dots, 3, 1$  となる。プロセス数が 4 の場合について、 $\rho^{\otimes n}$  と  $\sigma^{\otimes n}$  の既約成分の格納方式と計算結果の集約方法をそれぞれ図 1, 図

2に示す。計算した既約成分は倍精度型（または多倍長型）の行列に収める。サイズが違う既約成分を添字つけて管理するために、「行列クラスへのポインタ」からなる配列を用いる。

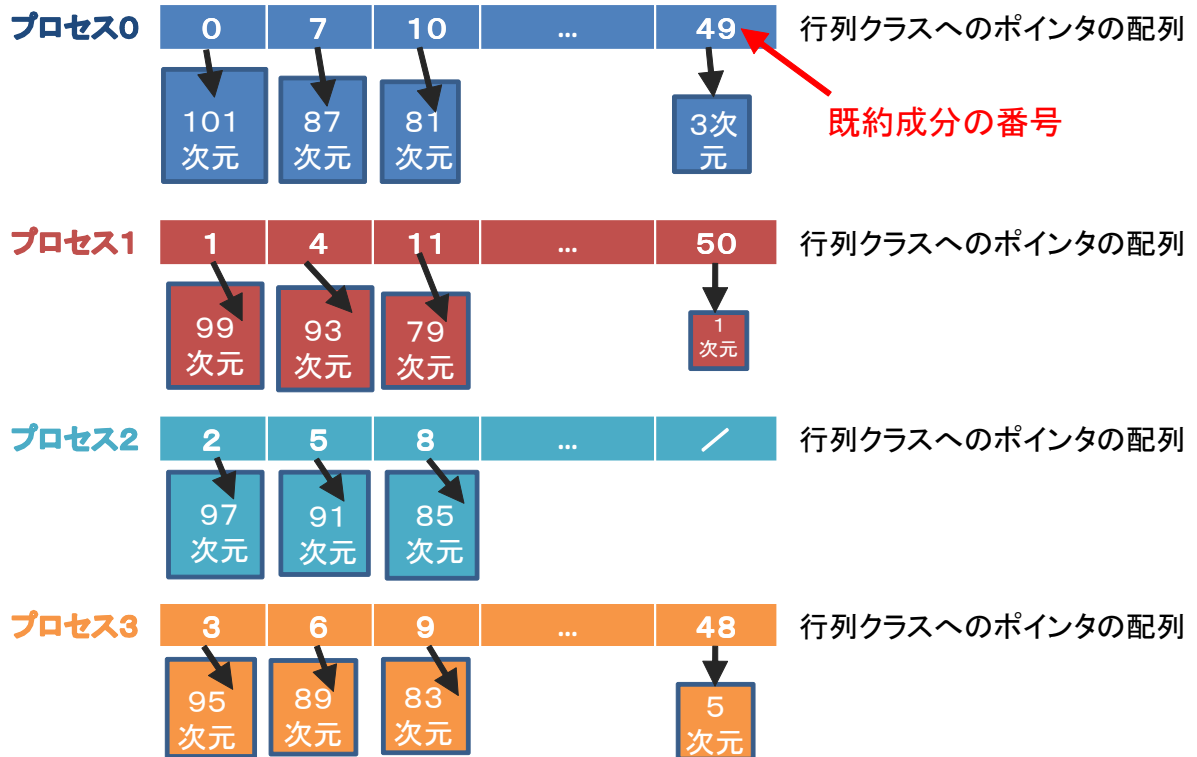


図1  $\rho^{\otimes n}$  と  $\sigma^{\otimes n}$  の既約成分の格納方式

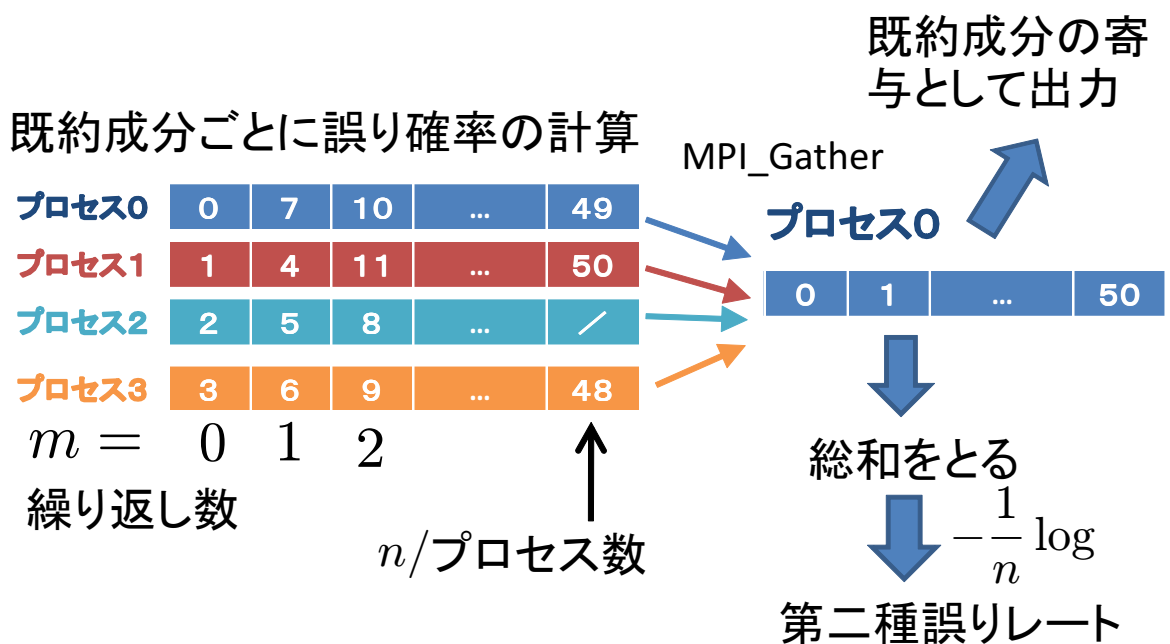


図2 計算結果の集約方法

## 7 シミュレーション結果

シミュレーションを行う際に使用する密度行列は

$$\rho = U\left(\frac{\pi}{6}\right)^* \begin{bmatrix} 0.75 & 0 \\ 0 & 0.25 \end{bmatrix} U\left(\frac{\pi}{6}\right), \quad \sigma = \begin{bmatrix} 0.9 & 0 \\ 0 & 0.1 \end{bmatrix} \quad (10)$$

とする。ここで、

$$U(\theta) := \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (11)$$

である。つまり、 $\rho$ にだけ回転をかけ、 $\rho$ と $\sigma$ を非可換にしているわけである。

まず、HA8000 (128 プロセス) を用いた Jacobi 法・倍精度による誤りレート  $r_n(a)$  の計算結果を図3に示す。

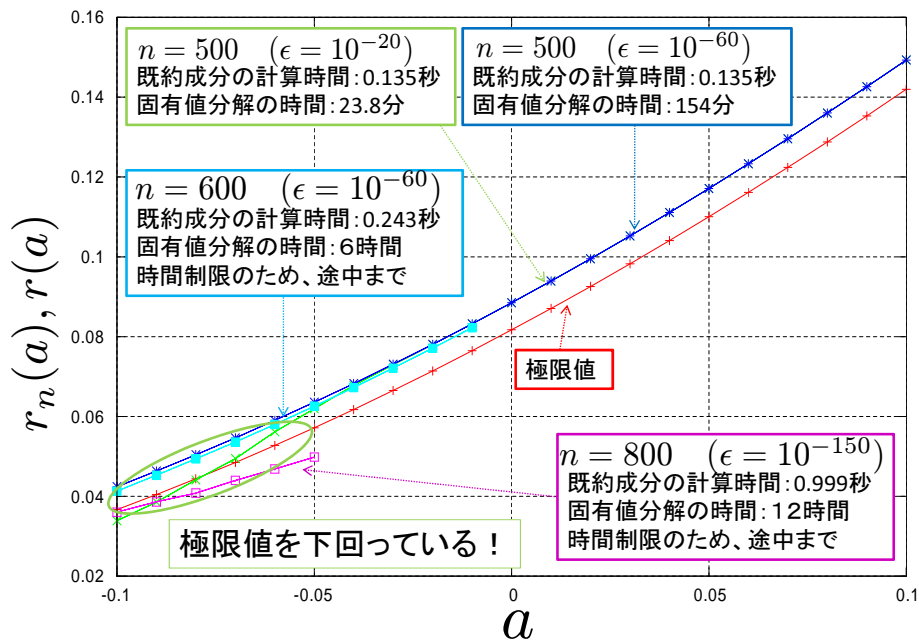


図3  $r_n(a)$  (Jacobi 法, 倍精度) と極限值  $r(a)$

テンソル次数  $n = 500$  の時のグラフを Jacobi 法の収束判定に用いる  $\epsilon$  を二通りに変えて描いている。緑色のグラフは  $\epsilon = 10^{-20}$  と甘く、青色のグラフは  $\epsilon = 10^{-60}$  ときつくとした。 $a$  が大きいところでは緑色のグラフは青色と重なって見えないが、 $a$  が小さいところでは、緑色のグラフは極限值を下回る歪みが発生している。一方、青色のグラフは常に滑らかである。 $r_n(a)$  はテンソル次数  $n$  の増加に伴って上から極限值  $r(a)$  に近づいていくことが理論的にわかっているので、極限值  $r(a)$  を下回っているのは数値誤差が原因である。したがって、数値誤差の原因は  $\epsilon$  が甘いことが原因である。数値誤差なく計算するには、 $\epsilon$  をテンソル次数  $n$  に応じて変える必要がある。 $n = 600$  のグラフは  $n = 500$  のグラフよりも、わずかに極限值に近づいている。 $n = 800$  の場合は  $\epsilon = 10^{-150}$  ときつくとっているが、グラフが大幅に極限值を下回っている。

以上の観察より、Jacobi 法は計算時間・誤差の観点から見て、倍精度では  $n = 600$  程度が限界であると思われる。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。また、全ての場合において、固有値分解の時間に比べて既約成分の計算時間はほとんど無視できる程度であることに注意して欲しい。

次に、 $n = 300$  の場合を QR 法を用いて倍精度と多倍長で計算した結果を図 4 に掲げる。

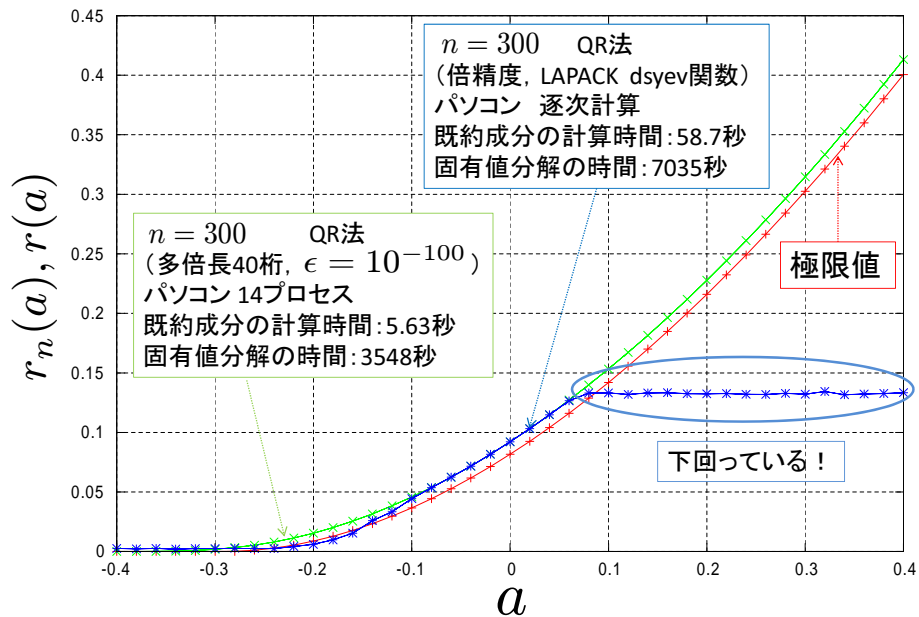


図 4  $n = 300$  の場合の  $r_n(a)$  (QR 法, 倍精度・多倍長) と極限值  $r(a)$

まず、逐次計算で行った LAPACK QR 法による計算時間が非常に短いことがわかる。しかし、LAPACK QR 法では、 $a$  が大きいところで歪んで極限值を下回っている。一方、多倍長 40 桁によるグラフは十分に滑らかで、極限值を下回っていない。したがって、LAPACK QR 法によるグラフの歪みは計算桁数の不足が原因であったことがわかる。

最後に、HA8000 を用いて  $n = 1200$  の場合について計算した結果を図 5 に示す。多倍長の

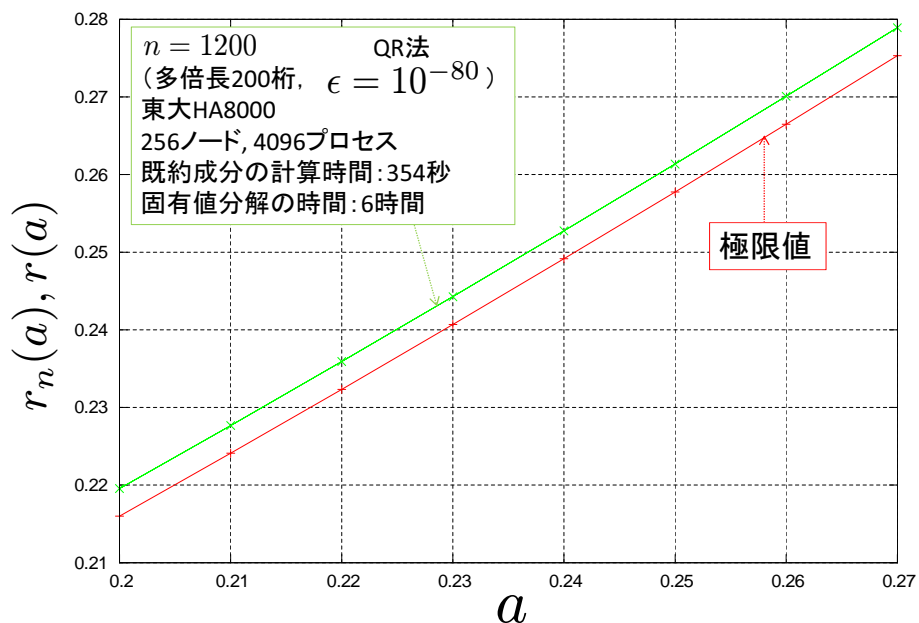


図 5  $n = 1200$  の場合の  $r_n(a)$  (QR 法, 多倍長) と極限值  $r(a)$

桁数を大きくとることによって、極限值を下回らず滑らかなグラフが得られており、十分な精

度が得られている。なお、計算時間の関係で、図 3, 図 4 よりも横軸は狭くなっている。

## 8 まとめ

既約分解を用いて、量子仮説検定を題材として量子 i.i.d. 状態のシミュレーションを行った。実装にあたっては、並列化や多倍長演算を用いて高速化・高精度化を実現した。

今後の課題としては、数値誤差の詳細な考察や量子推定など他の問題への応用が挙げられる。また、固有値分解が計算時間の大半を占めるので、多倍長の固有値分解ルーチンの性能の向上を目指したい。特に、ボトルネックになっている三重対角化の単体高速化・スレッド並列化を行いたい。

また、 $n = 2000$  の場合について計算を試みたが、使用メモリが限度の 28GB を超えて強制終了してしまうことがわかっている。そこで、既約成分の保存をやめる、または 1 ノードあたりの使用コア数を減らすといった対策が考えられる。しかし、既約成分の保存をやめる対策をとると、各  $a$  に対して既約成分の計算を行わなければならないので、既約成分の計算速度がもっと速くなることが望ましい。

## 謝辞

本研究を進めるにあたって、ご指導頂いた長岡浩司教授に感謝申し上げます。また、数値計算法に関して助言をいただいた片桐孝洋氏に感謝する。最後に、スパコン若手利用者推薦の支援をいただいた東京大学情報基盤センター・スーパーコンピューティング部門に謝意を表す。

## 参考文献

- [1] 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](http://eigen.tuxfamily.org/index.php?title=Main_Page).
- [2] J. W. Demmel. *Applied Numerical Linear Algebra*. SIAM, 1997.
- [3] G. H. Golub. *Matrix Computation*. The Johns Hopkins University Press, third edition edition, 1996.
- [4] R. Goodman and N. R. Wallach. *Representations and Invariants of the Classical Groups*. Cambridge University Press, 1998.
- [5] M. Hayashi. Asymptotics of quantum relative entropy from a representation theoretical viewpoint. *J. Phys. A: Math. Gen.*, 34(16):3413–3419, 2001.
- [6] M. Hayashi. Error exponent in asymmetric quantum hypothesis testing and its application to classical-quantum channel coding. *Phys. Rev. A*, 76(6):062301–1,062301–4, 2007.
- [7] H. Nagaoka. The converse part of the theorem for quantum Hoeffding bound. *quant-ph/0611289v1*, 2006.
- [8] H. Nagaoka and M. Hayashi. An information-spectrum approach to classical and



- quantum hypothesis testing for simple hypotheses. *IEEE Trans. Inf. Theory*, 53(2):534–549, 2007.
- [9] T. Sakashita and H. Nagaoka. A numerical study of hypothesis testing for quantum i.i.d. states. *Asian conference on quantum information science*, pp. 213–214, 2010.
- [10] 藤原. 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/>.
- [11] 柿崎. 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. Master's thesis, 電気通信大学情報システム学研究科, 2006.
- [12] 坂下, 長岡. 量子 i.i.d. 状態のシミュレーションとその理論的考察. 日本応用数理学会 2010 年度年会 講演予稿集, pp. 245–246, 2010.
- [13] 堂嶋, 片桐, 長岡. 量子 i.i.d. 状態における仮説検定の漸近特性に関する数値的アプローチ. 第 29 回情報理論とその応用シンポジウム予稿集 (SITA), pp. 767–770, 2006.