A Parallel Approximation Algorithm for Positive Semidefinite Programming

Rahul Jain
Centre for Quantum Technologies
Department of Computer Science
National University of Singapore
Block S15, 3 Science Drive 2, Singapore 11754
rahul@comp.nus.edu.sg

Penghui Yao
Centre for Quantum Technologies
Department of Computer Science
National University of Singapore
Block S15, 3 Science Drive 2, Singapore 11754
pyao@nus.edu.sg

***Abstract*—Positive semidefinite programs are an important subclass of semidefinite programs in which all matrices involved in the specification of the problem are positive semidefinite and all scalars involved are non-negative. We present a parallel algorithm, which given an instance of a positive semidefinite program of size N and an approximation factor $\varepsilon > 0$, runs in (parallel) time $\mathrm{poly}(\frac{1}{\varepsilon}) \cdot \mathrm{polylog}(N)$, using $\mathrm{poly}(N)$ processors, and outputs a value which is within multiplicative factor of $(1+\varepsilon)$ to the optimal. Our result generalizes analogous result of Luby and Nisan [10] for positive linear programs and our algorithm is inspired by the algorithm of [10].**

***Keywords*-Fast parallel algorithms, positive semidefinite programming, multiplicative weight update;**

I. Introduction

Fast parallel algorithms for approximating optimum solutions to different subclasses of semidefinite programs have been studied in several recent works (e.g. [2], [3], [9], [7], [6], [5]) leading to many interesting applications including the celebrated result $\mathrm{QIP} = \mathrm{PSPACE}$ [5]. However for each of the algorithms used for example in [7], [6], [5], in order to produce a $(1+\varepsilon)$ approximation of the optimal value for a given semidefinite program of size N, in the corresponding subclass that they considered, the (parallel) running time was $\mathrm{polylog}(N) \cdot \mathrm{poly}(\kappa) \cdot \mathrm{poly}(\frac{1}{\varepsilon})$, where κ was a 'width' parameter that depended on the input semidefinite program (and was defined differently for each of the algorithms). For the specific instances of the semidefinite programs arising out of the applications considered in [7], [6], [5], it was separately argued that the corresponding 'width parameter' κ is at most $\mathrm{polylog}(N)$ and therefore the running time remained $\mathrm{polylog}(N)$ (for constant ε). It was therefore desirable to remove the polynomial dependence on the 'width' parameter and obtain a truly polylog running time algorithm, for a reasonably large subclass of semidefinite programs.

In this work we consider the class of positive semidefinite programs. A positive semidefinite program can be expressed in the following standard form (we use symbols $\geq, \leq$ to also represent Löwner order).

<u>Primal problem P</u>

$$\begin{aligned} \text{minimize:} \quad & \mathrm{Tr}\, CX \\ \text{subject to:} \quad & \forall i \in [m] : \mathrm{Tr}\, A_i X \geq b_i, \\ & X \geq 0. \end{aligned}$$

<u>Dual problem D</u>

$$\begin{aligned} \text{maximize:} \quad & \sum_{i=1}^{m} b_i y_i \\ \text{subject to:} \quad & \sum_{i=1}^{m} y_i \cdot A_i \leq C, \\ & \forall i \in [m] : y_i \geq 0. \end{aligned}$$

Here $C, A_1, \ldots, A_m$ are $n \times n$ positive semidefinite matrices and $b_1, \ldots, b_m$ are non-negative reals (in a general semidefinite program $C, A_1, \ldots, A_m$ are Hermitian and $b_1, \ldots, b_m$ are reals). Let us assume that the conditions for strong duality are satisfied and the optimum value for P, denoted $\mathrm{opt}(P)$, equals the optimum value for D, denoted $\mathrm{opt}(D)$. We present an algorithm, which given as input, $(C, A_1, \ldots, A_m, b_1, \ldots, b_m)$, and an error parameter $\varepsilon > 0$, outputs a $(1+\varepsilon)$ approximation to the optimum value of the program, and has running time $\mathrm{polylog}(n) \cdot \mathrm{polylog}(m) \cdot \mathrm{poly}(\frac{1}{\varepsilon})$. As can be noted, there is no polynomial dependence on any 'width' parameter on the running time of our algorithm. The classes of semidefinite programs used in [7], [6] are a subclass of positive semidefinite programs and hence our algorithm can directly be applied to the programs in them without needing any other argument about the 'width' being polylog in the size of the program (to obtain an algorithm running in polylog time).

Our algorithm is inspired by the algorithm used by Luby and Nisan [10] to solve positive linear programs. Positive linear programs can be considered as a special case of positive semidefinite programs in which the matrices used in the description of the program are all pairwise commuting. Our algorithm (and the algorithm in [10]) is based on the 'multiplicative weights update' (MWU) method. This is a powerful technique for 'experts learning' and finds its origins in various fields including learning theory, game theory, and optimization. The algorithms used in [2], [3], [9], [7], [6], [5] are based on its matrix variant the 'matrix multiplicative weights update' method. The algorithm of Luby and Nisan [10] proceeds in phases, where in each phase the large eigenvalues of $\sum_{i=1}^{m} y_i^t A_i$ (y_i^ts represent the candidate dual variables at time t) are sought to be brought below a threshold determined for that phase. The primal variable at time step t is chosen to be the projection onto the large eigenvalues (above the threshold) eigenspace of $\sum_{i=1}^{m} y_i^t A_i$. Using the sum of the primal variables generated so far, the dual variables are updated using the MWU method. A suitable scaling parameter λ_t is chosen during this update, which is small enough so that the good properties needed in the analysis of MWU are preserved and at the same time is large enough so that there is reasonable progress in bringing down the large eigenvalues.

Due to the non-commutative nature of the matrices involved in our case, our algorithm primarily deviates from that of [10] in how the threshold is determined inside each phase. The problem that is faced is roughly as follows. Since A_i's could be non-commuting, when y_i^ts are scaled down, the sum of the large eigenvalues of $\sum_{i=1}^{m} y_i^t A_i$ may not come down and this scaling may just move the large eigenvalues eigenspace. Therefore a suitable extra condition needs to be ensured while choosing the threshold. Due to this, our analysis also primarily deviates from [10] in bounding the number of time steps required in any phase and is significantly more involved. The analysis requires us to study the relationship between the large eigenvalues eigenspaces before and after scaling (say W_1 and W_2). For this purpose we consider the decomposition of the underlying space into one and two-dimensional subspaces which are invariant under the actions of both Π_1 and Π_2 (projections onto W_1 and W_2 respectively) and this helps the analysis significantly. Such decomposition has been quite useful in earlier works as well for example in quantum walk [14], [13], [1] and quantum complexity theory [11], [12].

We present the algorithm in the next section and its analysis, both optimality and the running time, in the subsequent section. We defer some proofs to the Appendix.

II. Algorithm

Given the positive semidefinite program (P, D) as above, we first show in Appendix A that without loss of generality (P, D) can be in the following special form.

Special form Primal problem P

$$\begin{aligned} \text{minimize:} \quad & \operatorname{Tr} X \\ \text{subject to:} \quad & \forall i \in [m] : \operatorname{Tr} A_i X \geq 1, \\ & X \geq 0. \end{aligned}$$

Special form Dual problem D

$$\begin{aligned} \text{maximize:} \quad & \sum_{i=1}^{m} y_i \\ \text{subject to:} \quad & \sum_{i=1}^{m} y_i \cdot A_i \leq I, \\ & \forall i \in [m] : y_i \geq 0. \end{aligned}$$

Here $A_1, \ldots, A_m$ are $n \times n$ positive semidefinite matrices and I represents the identity matrix. Furthermore, for all i, norm of A_i, denoted $\|A_i\|$, is at most 1 and the minimum non-zero eigenvalue of A_i is at least $\frac{1}{\gamma}$ where $\gamma = \frac{m^2}{\varepsilon^2}$. Also $m \geq n$.

In order to compactly describe the algorithm, and also the subsequent analysis, we introduce some notation. Let $Y = \operatorname{Diag}(y_1, \ldots, y_m)$ ($m \times m$ diagonal matrix with $Y(i,i) = y_i$ for $i \in [m]$). Let Φ be the map (from $n \times n$ positive semidefinite matrices to $m \times m$ positive semidefinite diagonal matrices) defined by $\Phi(X) = \operatorname{Diag}(\operatorname{Tr} A_1 X, \ldots, \operatorname{Tr} A_m X)$. Then its adjoint map Φ^* acts as $\Phi^*(Y) = \sum_{i=1}^{m} Y(i,i) \cdot A_i$ (for all diagonal matrices $Y \geq 0$). We let I represent the identity matrix (in the appropriate dimensions clear from the context). For Hermitian matrix B and real number l, let $N_l(B)$ represent the sum of eigenvalues of B which are at least l. The algorithm is presented in Figure 1.

III. Analysis

For all of this section, let $\varepsilon_1 = \frac{3\varepsilon}{\ln n}$. In the following we assume that n is sufficiently large and ε is sufficiently small.

A. Optimality

In this section we present the analysis assuming that all the operations performed by the algorithm are perfect. We claim, without going into further details, that similar analysis can be performed while taking into account the accuracy loss due to the actual operations of the algorithm in the limited running time.

We start with following claims.

Claim 1. *For all $t \leq t_f$, λ_t satisfies the conditions* 1. *and* 2. *in Step (3d) in the Algorithm.*

Proof: Easily verified. ■

Claim 2. $\alpha > 0$.

Proof: Follows since $\frac{1}{m^{1/\varepsilon}} \geq \operatorname{Tr} Y_{t_f} = \operatorname{Tr} \exp(-\Phi(X_{t_f})) > \exp(-\alpha)$. ■

Following lemma shows that for any time t, $\|\Phi^*(Y_t)\|$ is not much larger than $(1+\varepsilon_0)^{\mathsf{thr}}$.

Lemma 3. *For all $t \leq t_f$, $\|\Phi^*(Y_t)\| \leq (1+\varepsilon_0)^{\mathsf{thr}}(1+\varepsilon_1)$.*

Proof: Fix any $t \leq t_f$. As $\operatorname{Tr}(\Phi^*(Y_t)) \leq n N_{(1+\varepsilon_0)^k}(\Phi^*(Y_t))$, the loop at Step 3(c) runs at most

Input : Positive semidefinite matrices $A_1, \ldots, A_m$ and error parameter $\varepsilon > 0$.

Output : X^* feasible for P and Y^* feasible for D.

1) Let $\varepsilon_0 = \frac{\varepsilon^2}{\ln^2 n}$, $t = 0, X_0 = 0$. Let k_s be the smallest positive number such that $(1+\varepsilon_0)^{k_s} \leq \|\Phi^*(I)\| < (1+\varepsilon_0)^{k_s+1}$. Let $k = k_s$.
2) Let $Y_t = \exp(-\Phi(X_t))$.
3) If $\operatorname{Tr} Y_t > \frac{1}{m^{1/\varepsilon}}$, do
   a) If $\|\Phi^*(Y_t)\| < (1+\varepsilon_0)^k$, then set $k \leftarrow k-1$ and repeat this step.
   b) Set $\mathsf{thr}' = k$.
   c) If
$$\begin{aligned} &N_{(1+\varepsilon_0)^{\mathsf{thr}'-1}}(\Phi^*(Y_t)) \\ &\geq (1+\frac{2}{5}\varepsilon) N_{(1+\varepsilon_0)^{\mathsf{thr}'}}(\Phi^*(Y_t)). \end{aligned}$$
   then $\mathsf{thr}' \leftarrow \mathsf{thr}' - 1$ and repeat this step. Else set $\mathsf{thr} = \mathsf{thr}'$.
   d) Let Π_t be the projector on the eigenspace of $\Phi^*(Y_t)$ with eigenvalues at least $(1+\varepsilon_0)^{\mathsf{thr}}$. For $\lambda > 0$, let $P_\lambda^{\geq}$ be the projection onto eigenspace of $\Phi(\lambda\Pi_t)$ with eigenvalues at least $2\sqrt{\varepsilon}$. Let $P_\lambda^{\leq}$ be the projection onto eigenspace of $\Phi(\lambda\Pi_t)$ with eigenvalues at most $2\sqrt{\varepsilon}$. Find λ_t such that
      1. $\operatorname{Tr}(P_{\lambda_t}^{\geq} Y_t P_{\lambda_t}^{\geq})\Phi(\Pi_t) \geq \sqrt{\varepsilon} \operatorname{Tr} Y_t \Phi(\Pi_t)$ and,
      2. $\operatorname{Tr}(P_{\lambda_t}^{\leq} Y_t P_{\lambda_t}^{\leq})\Phi(\Pi_t) \geq (1-\sqrt{\varepsilon}) \operatorname{Tr} Y_t \Phi(\Pi_t)$ as follows.
      i) Sort $\{\operatorname{Tr} A_i \Pi_t\}_{i=1}^m$ in non-increasing order. Suppose $\operatorname{Tr} A_{j_1}\Pi_t \geq \operatorname{Tr} A_{j_2}\Pi_t \geq \cdots \geq \operatorname{Tr} A_{j_m}\Pi_t$.
      ii) Let y_j be the j-th diagonal entry of Y_j. Find index $r \in [m]$ satisfying
$$\sum_{k=1}^{r} y_{j_k} \operatorname{Tr} A_{j_k}\Pi_t \geq \sqrt{\varepsilon} \sum_{k=1}^{m} y_{j_k} \operatorname{Tr} A_{j_k}\Pi_t, \text{ and}$$
$$\sum_{k=r}^{m} y_{j_k} \operatorname{Tr} A_{j_k}\Pi_t \geq (1-\sqrt{\varepsilon}) \sum_{k=1}^{m} y_{j_k} \operatorname{Tr} A_{j_k}\Pi_t.$$
      iii) Let $\lambda_t = \frac{2\sqrt{\varepsilon}}{\operatorname{Tr} A_{j_r}\Pi_t}$.
   e) Let $X_{t+1} = X_t + \lambda_t \Pi_t$. Set $t \leftarrow t+1$ and go to Step 2.
4) Let $t_f = t$, $k_f = k$. Let α be the minimum eigenvalue of $\Phi(X_{t_f})$. Output $X^* = X_{t_f}/\alpha$.
5) Let t' be such that $\operatorname{Tr} Y_{t'} / \|\Phi^*(Y_{t'})\|$ is the maximum among all time steps. Output $Y^* = Y_{t'} / \|\Phi^*(Y_{t'})\|$.

Figure 1. Algorithm

$\frac{\ln n}{\ln(1+\frac{2\varepsilon}{5})}$ times. Hence

$$\begin{aligned} \|\Phi^*(Y_t)\| &\leq (1+\varepsilon_0)^{k+1} \leq (1+\varepsilon_0)^{\mathsf{thr}}(1+\varepsilon_0)^{\frac{\ln n}{\ln(1+\frac{2\varepsilon}{5})}+1} \\ &< (1+\varepsilon_0)^{\mathsf{thr}}(1+\frac{3\varepsilon}{\ln n}) = (1+\varepsilon_0)^{\mathsf{thr}}(1+\varepsilon_1). \end{aligned}$$

■

Following lemma shows that as t increases, there is a reduction in the trace of the dual variable in terms of the trace of the primal variable.

Lemma 4. *For all $t \leq t_f$ we have,* $\operatorname{Tr} Y_{t+1} \leq \operatorname{Tr} Y_t - \lambda_t \cdot (1-4\sqrt{\varepsilon}) \cdot \|\Phi^*(Y_t)\| \cdot (\operatorname{Tr}\Pi_t)$.

Proof: Fix any $t \leq t_f$. Let $B = P_{\lambda_t}^{\leq}\Phi(\lambda_t\Pi_t)P_{\lambda_t}^{\leq}$. Note that $B \leq \Phi(\lambda_t\Pi_t)$ and also $B \leq 2\sqrt{\varepsilon}I$. Second last inequality below follows from Lemma 3 which shows that all eigenvalues of $\Pi_t\Phi^*(Y_t)\Pi_t$ are at least $(1-\varepsilon_1)\|\Phi^*(Y_t)\|$.

$$\begin{aligned} \operatorname{Tr} Y_{t+1} &= \operatorname{Tr}\exp(-\Phi(X_t) - \Phi(\lambda_t\Pi_t)) \\ &\leq \operatorname{Tr}\exp(-\Phi(X_t) - B) \\ &= \operatorname{Tr}\exp(-\Phi(X_t))\exp(-B) \\ &\leq \operatorname{Tr}\exp(-\Phi(X_t))(I - (1-2\sqrt{\varepsilon})B) \\ &= \operatorname{Tr} Y_t - (1-2\sqrt{\varepsilon})\operatorname{Tr} Y_t B \\ &\leq \operatorname{Tr} Y_t - (1-\sqrt{\varepsilon})(1-2\sqrt{\varepsilon})\operatorname{Tr} Y_t\Phi(\lambda_t\Pi_t) \\ &= \operatorname{Tr} Y_t - (1-\sqrt{\varepsilon})(1-2\sqrt{\varepsilon})\operatorname{Tr}\Phi^*(Y_t)\lambda_t\Pi_t \\ &\leq \operatorname{Tr} Y_t - (1-\varepsilon_1)(1-\sqrt{\varepsilon})(1-2\sqrt{\varepsilon})\lambda_t\|\Phi^*(Y_t)\|(\operatorname{Tr}\Pi_t) \\ &\leq \operatorname{Tr} Y_t - (1-4\sqrt{\varepsilon})\lambda_t\|\Phi^*(Y_t)\|(\operatorname{Tr}\Pi_t). \end{aligned}$$

The first inequality holds because $A_1 \geq A_2$ implies $\operatorname{Tr}\exp(A_1) \geq \operatorname{Tr}\exp(A_2)$, the second equality because both B and $\Phi(X_t)$ are diagonal, the second inequality because $A \leq I$ implies $\exp(-\delta A) \leq I - \delta(1-\delta)A$, and the third inequality is from step 3(d) part 1. ■

Following lemma relates the trace of X_{t_f} with the trace of Y^* and Y_{t_f}.

Lemma 5. $\operatorname{Tr} X_{t_f} \leq \frac{1}{(1-4\sqrt{\varepsilon})} \cdot (\operatorname{Tr} Y^*) \cdot \ln(m/\operatorname{Tr} Y_{t_f})$.

Proof: Using Lemma 4 we have,

$$\begin{aligned} \frac{\operatorname{Tr} Y_{t+1}}{\operatorname{Tr} Y_t} &\leq 1 - \frac{(1-4\sqrt{\varepsilon})\lambda_t\|\Phi^*(Y_t)\|(\operatorname{Tr}\Pi_t)}{\operatorname{Tr} Y_t} \\ &\leq \exp\left(-\frac{(1-4\sqrt{\varepsilon})\lambda_t\|\Phi^*(Y_t)\|(\operatorname{Tr}\Pi_t)}{\operatorname{Tr} Y_t}\right) \\ &\leq \exp\left(-\frac{(1-4\sqrt{\varepsilon})\lambda_t\operatorname{Tr}\Pi_t}{\operatorname{Tr} Y^*}\right) \\ &= \exp\left(-\frac{(1-4\sqrt{\varepsilon})\operatorname{Tr}(X_{t+1}-X_t)}{\operatorname{Tr} Y^*}\right). \end{aligned}$$

The second inequality holds because $\exp(-x) \geq 1-x$, and the third inequality is from property of Y^*. This implies,

$$\begin{aligned} &\operatorname{Tr} Y_{t_f} \leq (\operatorname{Tr} Y_0)\exp\left(-\frac{(1-4\sqrt{\varepsilon})\operatorname{Tr} X_{t_f}}{\operatorname{Tr} Y^*}\right) \\ &\Rightarrow \operatorname{Tr} X_{t_f} \leq \frac{(\operatorname{Tr} Y^*)\ln(m/(\operatorname{Tr} Y_{t_f}))}{(1-4\sqrt{\varepsilon})} \quad (\text{since } \operatorname{Tr} Y_0 = m). \end{aligned}$$

■

We can now finally bound the trace of X^* in terms of the trace of Y^*.

Theorem 6. *X^* and Y^* are feasible for the P and D respectively and*

$$\operatorname{Tr} X^* \leq (1+5\sqrt{\varepsilon})\operatorname{Tr} Y^* \ .$$

Therefore, since $\mathrm{opt}(P) = \mathrm{opt}(D)$,

$$\mathrm{opt}(D) = \mathrm{opt}(P) \leq \operatorname{Tr} X^* \leq (1 + 5\sqrt{\varepsilon}) \operatorname{Tr} Y^* \\ \leq (1 + 5\sqrt{\varepsilon})\mathrm{opt}(D) = (1 + 5\sqrt{\varepsilon})\mathrm{opt}(P).$$

Proof: It is easily verified that X^* and Y^* are feasible for P and D respectively. From Lemma 5 we have,

$$\alpha \operatorname{Tr} X^* = \operatorname{Tr} X_{t_f} \leq \frac{1}{1 - 4\sqrt{\varepsilon}} \cdot (\operatorname{Tr} Y^*) \cdot \ln(m / \operatorname{Tr} Y_{t_f}) \ .$$

Since $Y_{t_f} = \exp(-\Phi(X_{t_f}))$ we have

$$\operatorname{Tr} Y_{t_f} \geq \left\| \exp(-\Phi(X_{t_f})) \right\| = \exp(-\alpha) \ .$$

Using above two equations we have,

$$\begin{aligned} \operatorname{Tr} X^* &\leq \frac{1}{1 - 4\sqrt{\varepsilon}} \cdot (\operatorname{Tr} Y^*) \cdot \frac{\ln(m / \operatorname{Tr} Y_{t_f})}{\ln(1 / \operatorname{Tr} Y_{t_f})} \\ &= \frac{1}{1 - 4\sqrt{\varepsilon}} \cdot (\operatorname{Tr} Y^*) \cdot \left(1 + \frac{\ln m}{\ln(1 / \operatorname{Tr} Y_{t_f})}\right) \\ &\leq \frac{1 + \varepsilon}{1 - 4\sqrt{\varepsilon}} \cdot (\operatorname{Tr} Y^*) \quad \left(\text{since } \operatorname{Tr} Y_{t_f} \leq \frac{1}{m^{1/\varepsilon}}\right) \\ &\leq (1 + 5\sqrt{\varepsilon}) \cdot \operatorname{Tr} Y^* \ . \end{aligned}$$

■

B. Time complexity

In this section we are primarily interested in bounding the number of iterations of the algorithm, that is we will bound k_f and also the number of iterations for any given k. We claim, without going into further details, that the actions required by the algorithm in any given iteration can all be performed in time $\mathrm{polylog}(n) \cdot \mathrm{polylog}(m) \cdot \mathrm{poly}(\frac{1}{\varepsilon})$ (since operations for Hermitian matrices like eigenspace decomposition, exponentiation, and other operations like sorting and binary search for a list of real numbers etc. can be all be performed in polylog time).

Let us first introduce some notation. Let A be a Hermitian matrix and l be a real number. Let

- Π_l^A denote the projector onto the space spanned by the eigenvectors of A with eigenvalues at least l. Let Π^A be shorthand for Π_1^A.
- $N_l(A)$ denote the sum of eigenvalues of A at least l. Thus $N_l(A) = \operatorname{Tr} \Pi_l^A A$. Let $N(A)$ be shorthand for $N_1(A)$.
- $\lambda_k(A)$ denote the k-th largest eigenvalue of A.
- $\lambda^{\downarrow}(A) \stackrel{\text{def}}{=} (\lambda_1(A), \cdots, \lambda_n(A))$.
- for any two vectors $u, v \in \mathcal{R}^n$ we say u majorizes v, denoted $u \succeq v$, iff $\sum_{i=1}^n u_i = \sum_{i=1}^n v_i$ and for any $j \in [n]$ we have, $\sum_{i=1}^j u_i \geq \sum_{i=1}^j v_i$.

We will need the following facts.

Fact 7. *[4] For $n \times n$ Hermitian matrices A and B, $A \geq B$ implies $\lambda_i(A) \geq \lambda_i(B)$ for all $1 \leq i \leq n$. Thus $N_l(A) \geq N_l(B)$ for any real number l.*

Fact 8. *[4] Let A be an $n \times n$ Hermitian matrix and $P_1, \cdots, P_r$ be a family of mutually orthogonal projections. Then $\lambda^{\downarrow}(A) \succeq \lambda^{\downarrow}(\sum_i P_i A P_i)$.*

Fact 9. *[8] For any two projectors Π and Δ, there exits an orthogonal decomposition of the underlying vector space into one dimensional and two dimensional subspaces that are invariant under both Π and Δ. Moreover, inside each two-dimensional subspace, Π and Δ are rank-one projectors.*

Lemma 10. *Let k_f be the final value of k. Then $k_s - k_f = \mathcal{O}(\frac{\log m \log^2 n}{\varepsilon^3})$.*

Proof: Note that $\|\Phi^*(I)\| = \|\sum_{i=1}^m A_i\| \leq m$, since for each i, $\|A_i\| \leq 1$. Hence

$$k_s = \mathcal{O}((\log m)/\varepsilon_0) \ .$$

Let $Y_{t_f - 1} = \mathrm{Diag}(y_1, \dots y_m)$. We have (since $m \geq n$ and for each $i :$ $\operatorname{Tr} A_i \geq \frac{1}{\gamma} = \frac{\varepsilon^2}{m^2}$),

$$\begin{aligned} & m(1 + \varepsilon_0)^{k_f + 1} \geq m \left\| \Phi^*(Y_{t_f - 1}) \right\| \geq n \left\| \Phi^*(Y_{t_f - 1}) \right\| \\ & \quad \geq \operatorname{Tr} \Phi^*(Y_{t_f - 1}) = \sum_{i=1}^m y_i \operatorname{Tr} A_i \geq \frac{\sum_{i=1}^m y_i}{\gamma} = \frac{\operatorname{Tr} Y_{t_f - 1}}{\gamma} \\ & \geq \frac{1}{m^{1/\varepsilon} \gamma} \geq \frac{\varepsilon^2}{m^{2 + 1/\varepsilon}} \ . \end{aligned}$$

Hence $k_f \geq -\mathcal{O}(\frac{\log m}{\varepsilon \varepsilon_0})$. Therefore $k_s - k_f = \mathcal{O}(\frac{\log m}{\varepsilon \varepsilon_0}) = \mathcal{O}(\frac{\log m \log^2 n}{\varepsilon^3})$. ■

Theorem 11. *For any fixed k, the number of iterations of the algorithm is at most $\mathcal{O}(\frac{\log^2 n}{\varepsilon_1^9 \varepsilon})$. Hence combined with Lemma 10, the total number of iterations of the algorithm is at most $\mathcal{O}(\frac{\log^{13} n \log m}{\varepsilon^{13}})$.*

Proof: Fix k. Assume that the Algorithm has reached step $3(d)$ for this fixed k , $\frac{6 \log^2 n}{\varepsilon_1^9 \varepsilon}$ times. As argued in the proof of Lemma 3, whenever Algorithm reaches step $3(d)$, $\mathsf{thr} \geq k - \frac{3 \ln n}{\varepsilon}$. Thus there exists a value s between k and $k - \frac{3 \ln n}{\varepsilon}$ such that $\mathsf{thr} = s$ at least $\frac{2 \log n}{\varepsilon_1^9}$ times.

From Lemma 3 we get that the sum of the eigenvalues above $(1 + \varepsilon_0)^s$, is at most $n(1 + \varepsilon_1)(1 + \varepsilon_0)^s$ at the beginning of this phase. Whenever $\mathsf{thr} \neq s$ in this phase, using Fact 7, we conclude that the eigenvalues of $\Phi^*(Y_t)$ above $(1 + \varepsilon_0)^s$ do not increase. Whenever $\mathsf{thr} = s$ in this phase, using Lemma 12, we conclude that the eigenvalues of $\Phi^*(Y_t)$ above $(1 + \varepsilon_0)^s$ reduce by a factor of $(1 - \varepsilon_1^9)$. This can be seen by letting A in Lemma 12 to be $\frac{1 - \exp(-2\sqrt{\varepsilon})}{(1 + \varepsilon_0)^s} . \Phi^*(P_{\lambda_t}^{\geq} Y_t P_{\lambda_t}^{\geq})$ and B to be $\frac{1}{(1 + \varepsilon_0)^s} \Phi^*(Y^t) - A$. Now condition $3(d)(1.)$ of the Algorithm gives condition (2) of Lemma 12. Condition (1) of Lemma 12 can also be seen to be satisfied (using Lemma 3) and condition (4) of Lemma 12 is false due to condition $3(c)$ of the Algorithm. This implies condition (3) of Lemma 12 must also be false which gives us the desired conclusion.

Therefore the eigenvalues of $\Phi^*(Y_t)$ above $(1+\varepsilon_0)^s$ (in particular above $(1+\varepsilon_0)^k$) will vanish before $\mathsf{thr} = s$, $\frac{2\log n}{\varepsilon_1^9}$ times. Hence k must decrease before the Algorithm has reached step $3(d)$, $\frac{6\log^2 n}{\varepsilon_1^9\varepsilon}$ times. ∎

Following is a key lemma. It states that for two positive semidefinite matrices A, B, if A has good weight in the large (above 1) eigenvalues space of $A+B$ and if the sum of large (above 1) eigenvalues of B is pretty much the same as for $A+B$, then the sum of eigenvalues of $A+B$, slightly below 1 should be a constant fraction larger than the sum above 1.

Lemma 12. *Let $\varepsilon' = \frac{\varepsilon_0}{1+\varepsilon_0}$. Let A, B be two $n\times n$ positive semidefinite matrices satisfying*

$$\|A+B\| \le 1+\varepsilon_1 \text{ and } \|B\| \ge 1, \tag{1}$$

$$\operatorname{Tr}\Pi^{A+B}A \ge \varepsilon \operatorname{Tr}\Pi^{A+B}(A+B), \text{ and} \tag{2}$$

$$\operatorname{Tr}\Pi^{B}B \ge (1-\varepsilon_1^9)\operatorname{Tr}\Pi^{A+B}(A+B). \tag{3}$$

Then

$$N_{1-\varepsilon'}(A+B) > (1+\frac{2}{5}\varepsilon)N(A+B). \tag{4}$$

Proof: In order to prove this Lemma we will need to first show a few other Lemmas. By Fact 9, Π^B and Π^{A+B} decompose the underlying space V as follows,

$$V = \left(\bigoplus_{i=1}^{k} V_i\right)\bigoplus W.$$

Above for each $i\in[k]$, V_i is either one-dimensional or two-dimensional subspace, invariant for both Π^B and Π^{A+B} and inside V_i at least one of Π^B and Π^{A+B} survives. W is the subspace where both Π^B and Π^{A+B} vanish. We identify the subspace V_i, and the projector onto itself, by the same symbol. For any matrix M, define M_i to be V_iMV_i. We can see that both the projectors Π^B and Π^{A+B} are decomposed into the direct sum of one-dimensional projectors as follows.

$$\Pi^B = \bigoplus_{i=1}^{k}\Pi_i^B \quad \text{and} \quad \Pi^{A+B} = \bigoplus_{i=1}^{k}\Pi_i^{A+B}.$$

Lemma 13. *For any $i\in[k]$, $\Pi^{B_i} = \Pi_i^B$ and $\Pi_i^{A+B} = \Pi^{A_i+B_i}$. That is, the eigenspace of B_i with eigenvalues at least 1, is exactly the restriction of Π^B to V_i and similarly for A_i+B_i.*

Proof: We prove $\Pi^{B_i} = \Pi_i^B$ and the other equality follows similarly. If $\operatorname{Tr}\Pi^BV_i = 0$ then $\Pi_i^B = 0$. It also means that all eigenvalues of B_i are strictly less than 1 and hence $\Pi^{B_i} = 0$. Now let us assume otherwise. If $\dim V_i = 1$, i.e. $V_i = \operatorname{span}\{|v\rangle\}$, then $\Pi^B|v\rangle = |v\rangle$. Therefore, $\Pi_i^B = |v\rangle\langle v|$, and $B_i = \langle v|B|v\rangle|v\rangle\langle v|$ and $\langle v|B|v\rangle \ge 1$, which means $\Pi^{B_i} = |v\rangle\langle v|$.

Now let $\dim V_i = 2$. Since $\dim \Pi_i^B = 1$, we can write orthogonal decomposition of $V_i = \Pi_i^B\bigoplus(V_i - \Pi_i^B)$. Let $\Pi_i^B = |v_1\rangle\langle v_1|$ and $V_i - \Pi_i^B = |v_0\rangle\langle v_0|$. Then,

$$\begin{aligned} B_i &= V_iBV_i = \Pi_i^B B\Pi_i^B + (V_i-\Pi_i^B)B(V_i-\Pi_i^B) \\ &= \langle v_1|B|v_1\rangle|v_1\rangle\langle v_1| + \langle v_0|B|v_0\rangle|v_0\rangle\langle v_0|. \end{aligned}$$

is the spectral decomposition of B_i. As $\Pi^B|v_1\rangle = \Pi_i^B|v_1\rangle = |v_1\rangle$ and $\Pi^B|v_0\rangle = \Pi_i^B|v_0\rangle = 0$, we have $\langle v_1|B|v_1\rangle \ge 1$ and $\langle v_0|B|v_0\rangle < 1$, and hence $\Pi^{B_i} = |v_1\rangle\langle v_1|$. ∎

Lemma 14.

$$\operatorname{Tr}\Pi^B B = \sum_{i=1}^{k}\operatorname{Tr}\Pi^{B_i}B_i, \tag{5}$$

$$\operatorname{Tr}\Pi^{A+B}B = \sum_{i=1}^{k}\operatorname{Tr}\Pi^{A_i+B_i}B_i, \text{ and} \tag{6}$$

$$\operatorname{Tr}\Pi^{A+B}(A+B) = \sum_{i=1}^{k}\operatorname{Tr}\Pi^{A_i+B_i}(A_i+B_i) \tag{7}$$

Then using Eq.(2) *and Eq.*(3) *we get,*

$$\sum_{i=1}^{k}\operatorname{Tr}\Pi^{A_i+B_i}B_i \le (1-\varepsilon)\sum_{i=1}^{k}\operatorname{Tr}\Pi^{A_i+B_i}(A_i+B_i). \tag{8}$$

$$\sum_{i=1}^{k}\operatorname{Tr}\Pi^{B_i}B_i \ge (1-\varepsilon_1^9)\sum_{i=1}^{k}\operatorname{Tr}\Pi^{A_i+B_i}(A_i+B_i). \tag{9}$$

Proof: We prove (5) and (6) and (7) follow similarly.

$$\operatorname{Tr}\Pi^B B = \sum_{i=1}^{k}\operatorname{Tr}\Pi_i^B B = \sum_{i=1}^{k}\operatorname{Tr}V_i\Pi_i^B V_i B$$

$$= \sum_{i=1}^{k}\operatorname{Tr}\Pi_i^B V_iBV_i = \sum_{i=1}^{k}\operatorname{Tr}\Pi_i^B B_i = \sum_{i=1}^{k}\operatorname{Tr}\Pi^{B_i}B_i.$$

∎

Remarks:

1) In any one-dimensional subspace $V_i = \operatorname{span}\{|v\rangle\}$ in the decomposition of V as above, if $\Pi^{A+B}|v\rangle = 0$, then $\langle v|(A+B)|v\rangle < 1$, which implies $\langle v|B|v\rangle < 1$, that is $\Pi^B|v\rangle = 0$. But this contradicts the fact that at least one of Π^B and Π^{A+B} does not vanish in V_i. Thus Π^{A+B} never vanishes in any of V_i. Therefore for all $i\in[k]$ we have $\operatorname{Tr}\Pi^{A_i+B_i}(A_i+B_i) = \operatorname{Tr}\Pi_i^{A+B}(A_i+B_i) \ge 1$.
2) From (1), for all $i\in[k]$, $\operatorname{Tr}\Pi^{A_i+B_i}(A_i+B_i) \le 1+\varepsilon_1$. Combined with (7), we have

$$k \le N(A+B) \le k(1+\varepsilon_1).$$

Lemma 15. *Let*

$$I = \{i\in[k] : \operatorname{Tr}\Pi^{A_i+B_i}B_i \le (1-\varepsilon^2)\operatorname{Tr}\Pi^{A_i+B_i}(A_i+B_i)\},$$

and

$$J = \{i\in[k] : \operatorname{Tr}\Pi^{B_i}B_i \ge (1-\varepsilon_1^8)\operatorname{Tr}\Pi^{A_i+B_i}(A_i+B_i)\}.$$

Then

$$|I \cap J| > \frac{99}{100}\varepsilon k.$$

Proof: From (8),

$$\begin{aligned}
& (1-\varepsilon^2)\sum_{i \notin I} N(A_i+B_i) \le (1-\varepsilon)\sum_{i=1}^{k} N(A_i+B_i) \\
\Rightarrow\ & (\varepsilon-\varepsilon^2)\sum_{i \notin I} N(A_i+B_i) \le (1-\varepsilon)\sum_{i \in I} N(A_i+B_i) \\
\Rightarrow\ & \varepsilon(k-|I|) \le (1+\varepsilon_1)|I| \quad \text{(from Remarks 1. and 2.)} \\
\Rightarrow\ & |I| \ge \frac{\varepsilon}{1+\varepsilon_1+\varepsilon}k.
\end{aligned}$$

From (9) (since for all $i \in [k], N(A_i+B_i) \ge N(B_i)$),

$$\begin{aligned}
& \sum_{i \in J} N(A_i+B_i) + (1-\varepsilon_1^8)\sum_{i \notin J} N(A_i+B_i) \\
& \ge (1-\varepsilon_1^9)\sum_{i=1}^{k} N(A_i+B_i) \\
\Rightarrow\ & \varepsilon_1 \sum_{i \in J} N(A_i+B_i) \ge (1-\varepsilon_1)\sum_{i \notin J} N(A_i+B_i) \\
\Rightarrow\ & \varepsilon_1(1+\varepsilon_1)|J| \ge (1-\varepsilon_1)(k-|J|) \\
\Rightarrow\ & |J| \ge \frac{1-\varepsilon_1}{1+\varepsilon_1^2}k.
\end{aligned}$$

The second last implication is from Remarks 1 and 2. Thus

$$|I \cap J| \ge \left(\frac{\varepsilon}{1+\varepsilon_1+\varepsilon} + \frac{1-\varepsilon_1}{1+\varepsilon_1^2} - 1\right)k > \frac{99}{100}\varepsilon k.$$

■

Remark:

3) Note that for any $i \in I \cap J$, $\dim V_i = 2$. Otherwise, either $\Pi^{A_i+B_i} = \Pi^{B_i}$ or $\Pi^{B_i} = 0$ and neither of these can happen in $I \cap J$ (from definitions of I and J).

The following lemma states that for each $i \in I \cap J$, the second eigenvalue of $A_i + B_i$ is close to 1. The proof goes essentially via direct calculation.

Lemma 16. *Let P and Q be 2×2 positive semidefinite matrices satisfying*

$$\|Q\| \ge 1, \quad \|P+Q\| \le 1+\varepsilon_1, \quad \lambda_2(P+Q) < 1, \tag{10}$$

$$\operatorname{Tr}\Pi^{P+Q}P \ge \varepsilon^2 \operatorname{Tr}\Pi^{P+Q}(P+Q) \quad \textit{and} \tag{11}$$

$$\operatorname{Tr}\Pi^{Q}Q \ge (1-\varepsilon_1^8)\operatorname{Tr}\Pi^{P+Q}(P+Q)\ . \tag{12}$$

Then $\lambda_2(P+Q) > 1 - \frac{1}{9}\varepsilon_1^3$.

Proof: Let η be the maximum non-negative real number such that $P - \eta(I - \Pi^{P+Q}) \ge 0$. Set $P_1 = P - \eta(I - \Pi^{P+Q})$ and $Q_1 = Q + \eta(I - \Pi^{P+Q})$. P_1, Q_1 satisfy all the conditions in this Lemma and P_1 is a rank one matrix. Furthermore, set $P_2 = P_1/\|Q_1\|$ and $Q_2 = Q_1/\|Q_1\|$. Again all the conditions in this Lemma are still satisfied by P_2, Q_2 since $\Pi^{Q_2} = \Pi^{Q_1}$ and $\Pi^{P_2+Q_2} = \Pi^{P_1+Q_1}$. As $\lambda_2(P_2+Q_2) \le \lambda_2(P_1+Q_1) = \lambda_2(P+Q)$, it suffices to prove that $\lambda_2(P_2+Q_2) > 1 - \frac{1}{9}\varepsilon_1^3$. Consider P_2, Q_2 in the diagonal bases of Q_2.

$$P_2 = \begin{pmatrix} |r|\cos^2\theta & r\sin\theta\cos\theta \\ r^*\sin\theta\cos\theta & |r|\sin^2\theta \end{pmatrix}, Q_2 = \begin{pmatrix} 1 & 0 \\ 0 & b \end{pmatrix}.$$

where $r \in \mathbb{C}$ and $0 \le b < 1$. Set $\lambda = \|P_2 + Q_2\|$. Eq. (12) (for P_2, Q_2) implies that

$$\lambda \le \frac{1}{1-\varepsilon_1^8} < 1 + 2\varepsilon_1^8. \tag{13}$$

Since

$$\begin{aligned}
\operatorname{Tr}\Pi^{Q_2}P_2 &= \operatorname{Tr}\Pi^{Q_2}(P_2+Q_2) - \operatorname{Tr}\Pi^{Q_2}Q_2 \\
&\le \operatorname{Tr}\Pi^{P_2+Q_2}(P_2+Q_2) - \operatorname{Tr}\Pi^{Q_2}Q_2 \\
&\le \varepsilon_1^8 \operatorname{Tr}\Pi^{P_2+Q_2}(P_2+Q_2) = \varepsilon_1^8\lambda < 2\varepsilon_1^8,
\end{aligned}$$

we have,

$$|r|\cos^2\theta < 2\varepsilon_1^8. \tag{14}$$

Observe that,

$$|v\rangle = \frac{1}{\sqrt{1 + \left(\frac{|r|\sin\theta\cos\theta}{\lambda-b-|r|\sin^2\theta}\right)^2}} \begin{pmatrix} 1 \\ \frac{r^*\sin\theta\cos\theta}{\lambda-b-|r|\sin^2\theta} \end{pmatrix},$$

is the eigenvector of $P_2 + Q_2$ with eigenvalue λ. Hence $\Pi^{P_2+Q_2} = |v\rangle\langle v|$. Note that $\lambda > b + |r|\sin^2\theta$, because $\lambda_2(P_2+Q_2) = 1 + |r| + b - \lambda < 1$. Consider

$$\begin{aligned}
& \operatorname{Tr}(\Pi^{P_2+Q_2}P_2) = \langle v|P_2|v\rangle \\
=\ & \frac{|r|\cos^2\theta + \frac{2|r|^2\sin^2\theta\cos^2\theta}{\lambda-b-|r|\sin^2\theta} + \frac{|r|^3\sin^4\theta\cos^2\theta}{(\lambda-b-|r|\sin^2\theta)^2}}{1 + \frac{|r|^2\sin^2\theta\cos^2\theta}{(\lambda-b-|r|\sin^2\theta)^2}} \\
=\ & \frac{|r|(\lambda-b)^2\cos^2\theta}{(\lambda-b-|r|\sin^2\theta)^2 + |r|^2\sin^2\theta\cos^2\theta} \\
\le\ & \frac{|r|\cos^2\theta}{(1-\frac{|r|\sin^2\theta}{\lambda-b})^2} \\
<\ & \frac{2\varepsilon_1^8}{(1-\frac{|r|\sin^2\theta}{\lambda-b})^2}.
\end{aligned}$$

Combining with Eq. (11) (for P_2, Q_2), we obtain

$$\begin{aligned}
& 2\varepsilon_1^8 \ge \varepsilon^2(1 - \frac{|r|\sin^2\theta}{\lambda-b})^2 \\
& \Rightarrow (1 - \frac{|r|\sin^2\theta}{\lambda-b})^2 < \frac{\varepsilon_1^6}{100} \\
& \Rightarrow |r|\sin^2\theta > (1 - \frac{1}{10}\varepsilon_1^3)(\lambda-b) \\
& \Rightarrow |r|\sin^2\theta + (1 - \frac{1}{10}\varepsilon_1^3)b > (1 - \frac{1}{10}\varepsilon_1^3)\lambda \\
& \Rightarrow |r| + b > (1 - \frac{1}{10}\varepsilon_1^3)\lambda > 1 - \frac{1}{10}\varepsilon_1^3\ .
\end{aligned}$$

Hence

$$\lambda_2(P_2+Q_2) = \mathrm{Tr}(P_2+Q_2) - \lambda = 1+|r|+b-\lambda$$
$$> 2 - \frac{1}{10}\varepsilon_1^3 - (1+2\varepsilon_1^8) > 1 - \frac{1}{9}\varepsilon_1^3\ .$$

■

We can finally prove Lemma 12. By Fact 7, $\lambda^{\downarrow}(A+B) \succeq \lambda^{\downarrow}(\sum_i A_i + B_i)$. Let $j_1 = \max\{j : \lambda_j(A+B) \geq 1\}$, $j_2 = \max\{j : \lambda_j(\sum_i(A_i+B_i)) \geq 1\}$, and $j_0 = j_1 + \frac{99}{100}\varepsilon k$. Then

$$\sum_{j\leq j_0} \lambda_j(A+B) \geq \sum_{j\leq j_0} \lambda_j\left(\sum_i (A_i+B_i)\right).$$

According to the decomposition in Fact 9, Lemma 14 and the remarks below it, $j_1 = j_2 = k$ and

$$\sum_{j\leq j_1} \lambda_j(A+B) = \mathrm{Tr}\,\Pi^{A+B}(A+B), \quad \text{and}$$

$$\sum_{j\leq j_2} \lambda_j\left(\sum_i (A_i+B_i)\right) = \sum_i \mathrm{Tr}\,\Pi^{A_i+B_i}(A_i+B_i).$$

The RHS of both the equations are equal by Lemma 14. Therefore,

$$\sum_{k<j\leq j_0} \lambda_j(A+B) \geq \sum_{k<j\leq j_0} \lambda_j\left(\sum_i (A_i+B_i)\right).$$

By Lemma 15 and Lemma 16,

$$\sum_{k<j\leq j_0} \lambda_j\left(\sum_i (A_i+B_i)\right) \geq \frac{99}{100}\varepsilon k\left(1-\frac{1}{9}\varepsilon_1^3\right).$$

Let $x = N_{1-\varepsilon'}(A+B) - N(A+B)$, then

$$\sum_{k<j\leq j_0} \lambda_j(A+B) \leq x + \left(\frac{99}{100}\varepsilon k - x\right)(1-\varepsilon').$$

Therefore from previous three inequalities,

$$\frac{99}{100}\varepsilon k\left(1-\frac{1}{9}\varepsilon_1^3\right) \leq x + \left(\frac{99}{100}\varepsilon k - x\right)(1-\varepsilon'),$$

which implies

$$x \geq \frac{99}{100}\varepsilon k\left(1-\frac{\varepsilon_1^3}{9\varepsilon'}\right).$$

Note that $\varepsilon_1^3 \ll \varepsilon'$, therefore from Remark 2.,

$$N_{1-\varepsilon'}(A+B) \geq k + \frac{99}{100}\varepsilon k\left(1-\frac{\varepsilon_1^3}{9\varepsilon'}\right) > \left(1+\frac{1}{2}\varepsilon\right)k$$
$$> \left(1+\frac{2}{5}\varepsilon\right)(1+\varepsilon_1)k \geq \left(1+\frac{2}{5}\varepsilon\right)N(A+B).$$

■

Acknowledgement: Penghui Yao would like to thank Attila Pereszlényi and Huangjun Zhu for helpful discussions.

REFERENCES

[1] Andris Ambainis, Andrew M. Childs, Ben W. Reichardt, Robert Spalek, and Shengyu Zhang. Any AND-OR formula of size n can be evaluated in time $n^{1/2+o(1)}$ on a quantum computer. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 363–372, Washington, DC, USA, 2007. IEEE Computer Society.

[2] S. Arora, E. Hazan, and S. Kale. Fast algorithms for approximate semidefinite programming using the multiplicative weights update method. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 339–348, 2005.

[3] S. Arora and S. Kale. A combinatorial, primal-dual approach to semidefinite programs. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing*, pages 227–236, 2007.

[4] Rajendra Bhatia. *Matrix Analysis*. Springer, New York, USA, 1996.

[5] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. QIP = PSPACE. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 573–582, New York, NY, USA, 2010. ACM.

[6] Rahul Jain, Sarvagya Upadhyay, and John Watrous. Two-message quantum interactive proofs are in PSPACE. In *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '09, pages 534–543, Washington, DC, USA, 2009. IEEE Computer Society.

[7] Rahul Jain and John Watrous. Parallel approximation of non-interactive zero-sum quantum games. In *Proceedings of the 2009 24th Annual IEEE Conference on Computational Complexity*, CCC '09, pages 243–253, Washington, DC, USA, 2009. IEEE Computer Society.

[8] Camille Jordan. Essai sur la géométrie à n dimensions. *Bulletin de la S. M. F*, 3:103–174, 1875.

[9] S. Kale. *Efficient algorithms using the multiplicative weights update method*. PhD thesis, Princeton University, 2007.

[10] Michael Luby and Noam Nisan. A parallel approximation algorithm for positive linear programming. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, STOC '93, pages 448–457, New York, NY, USA, 1993. ACM.

[11] Chris Marriott and John Watrous. Quantum Arthur—Merlin games. *Comput. Complex.*, 14:122–152, June 2005.

[12] Daniel Nagaj, Pawel Wocjan, and Yong Zhang. Fast amplification of QMA. *Quantum Information and Computation*, 9(11), 2009.

[13] Oded Regev. Witness-preserving QMA amplification, quantum computation lecture notes. 2006.

[14] Mario Szegedy. Quantum speed-up of markov chain based algorithms. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 32–41, Washington, DC, USA, 2004. IEEE Computer Society.

Appendix

Let us consider an instance of a positive semidefinite program as follows.

Primal problem P

$$\begin{aligned} \text{minimize:} \quad & \operatorname{Tr} CX \\ \text{subject to:} \quad & \forall i \in [m] : \operatorname{Tr} A_i X \geq b_i, \\ & X \geq 0. \end{aligned}$$

Dual problem D

$$\begin{aligned} \text{maximize:} \quad & \sum_{i=1}^{m} b_i y_i \\ \text{subject to:} \quad & \sum_{i=1}^{m} y_i \cdot A_i \leq C, \\ & \forall i \in [m] : y_i \geq 0. \end{aligned}$$

Above $C, A_1, \ldots, A_m$ are $n \times n$ positive semidefinite matrices and $b_1, \ldots, b_m$ are non-negative. Let us assume that conditions for strong duality are satisfied and optimum value for P, denoted $\operatorname{opt}(P)$, equals the optimum value for D, denoted $\operatorname{opt}(D)$.

We show how to transform the primal problem to the special form and a similar transformation can be applied to dual problem. First observe that if for some i, $b_i = 0$, the corresponding constraint in primal problem is trivial and can be removed. Similarly if for some i, the support of A_i is not contained in the support of C, then y_i must be 0 and can be removed. Therefore we can assume w.l.o.g. that for all $i, b_i > 0$ and the support of A_i is contained in the support of C. Hence w.l.o.g we can take the support of C as the whole space, in other words, C is invertible. Also at this stage assume w.l.o.g $m \geq n$ (by repeating the first constrain in P if necessary). For all $i \in [m]$, define $A'_i \stackrel{\text{def}}{=} \frac{C^{-1/2} A_i C^{-1/2}}{b_i}$. Consider the normalized Primal problem.

Normalized Primal problem P'

$$\begin{aligned} \text{minimize:} \quad & \operatorname{Tr} X' \\ \text{subject to:} \quad & \forall i \in [m] : \operatorname{Tr} A'_i X' \geq 1, \\ & X' \geq 0. \end{aligned}$$

Claim 17. *If X is a feasible solution to P, then $C^{1/2} X C^{1/2}$ is a feasible solution to P' with the same objective value. Similarly if X' is a feasible solution to P', then $C^{-1/2} X' C^{-1/2}$ is a feasible solution to P with the same objective value. Hence* $\operatorname{opt}(P) = \operatorname{opt}(P')$.

Proof: Easily verified. ∎

The next step to transforming the problem is to limit the range of eigenvalues of A'_is. Let $\beta = \min_i \|A'_i\|$.

Claim 18. $\frac{1}{\beta} \leq \operatorname{opt}(P') \leq \frac{m}{\beta}$.

Proof: Note that $\frac{1}{\beta} I$ is a feasible solution for P'. This implies $\operatorname{opt}(P') \leq \frac{n}{\beta} \leq \frac{m}{\beta}$. Let X' be an optimal feasible solution for P'. Let j be such that $\|A'_j\| = \beta$. Then $\beta \operatorname{Tr} X' \geq \operatorname{Tr} A'_j X' \geq 1$, hence $\frac{1}{\beta} \leq \operatorname{opt}(P')$. ∎

Let $A'_i = \sum_{j=1}^{n} a'_{ij} |v_{ij}\rangle\langle v_{ij}|$ in its spectral decomposition. Define for all $i \in [m]$ and $j \in [n]$,

$$a''_{ij} \stackrel{\text{def}}{=} \begin{cases} \frac{\beta m}{\varepsilon} & \text{if } a'_{ij} > \frac{\beta m}{\varepsilon}, \\ 0 & \text{if } a'_{ij} < \frac{\varepsilon \beta}{m}, \\ a'_{ij} & \text{otherwise.} \end{cases} \tag{15}$$

Define $A''_i = \sum_{j=1}^{n} a''_{ij} |v_{ij}\rangle\langle v_{ij}|$. Consider the transformed Primal problem P''.

Transformed Primal problem P''

$$\begin{aligned} \text{minimize:} \quad & \operatorname{Tr} X'' \\ \text{subject to:} \quad & \forall i \in [m] : \operatorname{Tr} A''_i X'' \geq 1, \\ & X'' \geq 0. \end{aligned}$$

Lemma 19. 1) *Any feasible solution to P'' is also a feasible solution to P'.*
2) $\operatorname{opt}(P') \leq \operatorname{opt}(P'') \leq \operatorname{opt}(P')(1 + \varepsilon)$.

Proof:

1) Follows immediately from the fact that $A''_i \leq A'_i$.
2) First inequality follows from 1. Let X' be an optimal solution to P' and let $\tau = \operatorname{Tr}(X')$. Let $X'' = X' + \frac{\varepsilon \tau}{m} I$. Then, since $m \geq n$, $\operatorname{Tr} X'' \leq (1+\varepsilon) \operatorname{Tr} X'$. Thus it suffices to show that X'' is feasible to P''.
Fix $i \in [m]$. Assume that there exists $j \in [n]$ such that $a'_{ij} \geq \frac{\beta m}{\varepsilon}$. Then, from Claim 18

$$\operatorname{Tr} A''_i X''_i \geq \operatorname{Tr} \frac{\beta m}{\varepsilon} |v_{ij}\rangle\langle v_{ij}| \cdot \frac{\varepsilon \tau}{m} I = \beta \tau \geq 1.$$

Now assume that for all $j \in [n]$, $a_{ij} \leq \frac{\beta m}{\varepsilon}$. By (15) and definition of β, $\|A''_i\| = \|A'_i\| \geq \beta$ and $A''_i \geq A'_i - \frac{\varepsilon \beta}{m} I$. Therefore

$$\begin{aligned} \operatorname{Tr} A''_i X'' &\geq \operatorname{Tr} A''_i X' + \beta \frac{\varepsilon \tau}{m} \\ &\geq \operatorname{Tr} A'_i X' + \beta \frac{\varepsilon \tau}{m} - \operatorname{Tr} \frac{\varepsilon \beta}{m} X' = \operatorname{Tr} A'_i X' \geq 1. \end{aligned}$$

∎

Note that for all $i \in [m]$, the ratio between the largest eigenvalue and the smallest nonzero eigenvalue of A''_i is at most $\frac{m^2}{\varepsilon^2} = \gamma$.

Finally, we get the special form Primal problem $\hat{P}$ as follows. Let $t = \max_{i \in [m]} \|A''_i\|$ and for all $i \in [m]$ define $\hat{A}_i \stackrel{\text{def}}{=} \frac{A''_i}{t}$. Consider,

Special form Primal problem $\hat{P}$

$$\begin{aligned} \text{minimize:} \quad & \operatorname{Tr} \hat{X} \\ \text{subject to:} \quad & \forall i \in [m] : \operatorname{Tr} \hat{A}_i \hat{X} \geq 1, \\ & \hat{X} \geq 0. \end{aligned}$$

It is easily seen that there is a one-to-one correspondence between the feasible solutions to $P^{''}$ and $\hat{P}$ and $\mathrm{opt}(\hat{P}) = t \cdot \mathrm{opt}(P'')$. Therefore $\hat{P}$ satisfies all the properties that we want and cumulating all we have shown above, we get following conclusion.

Lemma 20. *Let $\hat{X}$ be a feasible solution to $\hat{P}$ such that $\mathrm{Tr}\,\hat{X} \leq (1+\varepsilon)\mathrm{opt}(\hat{P})$. A feasible solution X to P can be derived from $\hat{X}$ such that $\mathrm{Tr}\,X \leq (1+\varepsilon)^2\mathrm{opt}(P)$.*

Furthermore we claim, without giving further details, that X can be obtained from $\hat{X}$ in time $\mathrm{polylog}(n) \cdot \mathrm{polylog}(m)$.