

تخمین پارامترهای یک مدل رگرسیون خطی با توسعه روش حداقل مربعات خطا بر اساس محاسبات کوانتومی

مهدی رضانی^{*}، استادیار، صادق کلانتری^۲، دانشجوی دکتری، علی مددی^۳، دانشیار

^۱ گروه ریاضی - دانشکده ریاضی - دانشگاه تفرش - تفرش - ایران

ramezani@tafreshu.ac.ir

^۲ گروه کنترل - دانشکده مهندسی برق - دانشگاه تفرش - تفرش - ایران

sadeghkalantari@tafreshu.ac.ir

^۳ گروه کنترل - دانشکده مهندسی برق - دانشگاه تفرش - تفرش - ایران

madadi@tafreshu.ac.ir

چکیده: روش حداقل مربعات خطا با وجود سادگی دارای نتایج قابل قبولی در شناسایی سیستم‌ها می‌باشد. فرآیند تخمین پارامترها در شناسایی سیستم منجر به حل معادله خطی $Ax = b$ می‌شود. نکته حائز اهمیت این است که روش عادی حل مسئله فوق دارای پیچیدگی محاسباتی از مرتبه $O(n^3)$ برای یک ماتریس $n \times n$ می‌باشد. در حل این مسئله، پیچیدگی محاسباتی با افزایش n (سایز ماتریس داده) افزایش می‌یابد. از طرفی تعداد نمونه‌های بیشتر سبب مدلسازی بهتر سیستم می‌گردد. در مسائل عملی شناسایی سیستم، هنگامی که تعداد داده‌های ورودی زیاد است بار محاسباتی به شدت افزایش می‌یابد. در این مقاله هدف این است که الگوریتم کوانتومی توسعه یافته‌ای برای حل مسئله شناسایی حداقل مربعات خطا ارائه نماییم. در این مقاله دو روش کلاسیک-کوانتومی و تمام کوانتومی ارائه می‌گردد. روش‌های ارائه شده در این مقاله قادر هستند برخلاف روش مرسوم HHL با ماتریس‌های غیرهرمیتی، بدحال و با وجود نویز رنگی، پارامترهای بدون بایاس را محاسبه نمایند. روش پیشنهادی کلاسیک-کوانتومی، دارای پیچیدگی محاسباتی از مرتبه $O(n^2 \log n)$ و روش تمام کوانتومی از مرتبه $O(\text{polylog} n)$ نسبت به سایز ماتریس داده می‌باشند. نتایج و مقایسه‌های انجام شده نشان می‌دهند که روش‌های پیشنهادی مقاله نسبت به روش‌های کلاسیک (با پیچیدگی $O(n^3)$) دارای پیچیدگی و محدودیت کمتری می‌باشند.

واژه‌های کلیدی: شناسایی سیستم، حداقل مربعات خطا، محاسبات کوانتومی، پیچیدگی محاسباتی، الگوریتم کوانتومی

* نویسنده مسئول: مهدی رضانی، ramezani@tafreshu.ac.ir

Parameter Estimation of a Linear Regression Model By Extending Least Square Method Based on Quantum Computing

Mehdi Ramezani ^{1*}, Assistant Professor, Sadegh Kalantari ², PhD Student, Ali Madadi ³, Associate Professor

¹ Department of Mathematics, Tafresh University, Tafresh

ramezani@tafreshu.ac.ir

² Department of Electrical Engineering, Tafresh University, Tafresh

sadeghkalantari@tafreshu.ac.ir

³ Department of Electrical Engineering, Tafresh University, Tafresh

madadi@tafreshu.ac.ir

Abstract: The least square error method yields acceptable results in system identification. The process of parameter estimation in system identification helps solve the linear equation $Ax = b$. The important point is that the normal method of solving the above problem has a computational complexity of $O(n^3)$ for a $n \times n$ matrix. In solving this problem, the computational complexity increases with the increase of n (the size of the data matrix). On the other hand, availability of more samples leads to better modeling of the system. In the practical problems of system identification, when the number of input data is large, the computational complexity increases greatly. In this article, the goal is to present the developed quantum algorithm for solving the problem of least square error identification. In this article, two classical-quantum and all-quantum methods are presented. Unlike conventional HHL methods, the proposed methods in this article are able to calculate unbiased parameters with non-Hermitian matrices, and color noise. The proposed classical-quantum method has a computational complexity of $O(n^2 \log n)$ and the all-quantum method has an order of $O(\text{polylog} n)$ in relation to the size of the data matrix. The results and comparisons show that the methods proposed in the article have less complexity and limitations than classical methods.

Keywords: *System identification, least squares error, quantum computing, computational complexity, quantum algorithm.*

* Corresponding author: Mehdi Ramezani, Ramezani@tafreshu.ac.ir

۱. مقدمه

منجر به جواب‌های مختلفی برای سیستم شوند. بنابراین اطلاع از نوع داده‌ها و مدلی که قرار است بکار برده شود بسیار حیاتی می‌باشد. در بحث انتخاب مدل، مدلی مناسب است که در عین سادگی تطابق قابل قبولی با سیستم داشته باشد. سرعت اجرای روش پیشنهادی نیز از این حیث حائز اهمیت است که در بسیاری از کاربردهای زمان‌واقعی^۱ و شناسایی برخط^۲، بسیار حیاتی است که شناسایی پارامترها با سرعت و دقت بالایی انجام شود. زیرا بسیاری از مدل‌های به کاررفته در شناسایی سیستم منجر به حل مسائل بهینه‌سازی پیچیده با روش‌های حل زمان بری می‌گردند. به همین منظور محققان در تلاش هستند تا همواره مدلی انتخاب شود که در عین سادگی دارای پیچیدگی محاسباتی^۳ مناسبی باشد [۷]. برای حل مشکل پیچیدگی محاسباتی الگوریتم‌ها اخیراً روش‌های مختلفی پیشنهاد شده است [۸-۱۰].

یکی از مهمترین روش‌های ارائه شده استفاده از کامپیوترهای کوانتومی می‌باشد. این کامپیوترها بر اساس قوانین فیزیک کوانتوم ساخته می‌شوند. با توجه به خواص مفید ذرات کوانتومی همچون درهم‌تنیدگی^۴ و موازی‌سازی^۵ میزان بهبود عملکرد کامپیوترهای کوانتومی نسبت به کامپیوترهای فعلی بسیار بهتر می‌باشد. در سالهای اخیر الگوریتم‌هایی بر اساس محاسبات کوانتومی ارائه شده است که پیچیدگی محاسباتی حل یک مسئله را تا حد بسیار زیادی کاهش داده‌اند. الگوریتم‌های کوانتومی

برای توصیف و تعریف پدیده‌های گوناگون معمولاً از مدل‌های ریاضی استفاده می‌گردد. اغلب مدل‌های ریاضی، دارای پارامترهایی می‌باشند که باید به نحوی شناسایی شوند. به طور کلی فرآیند شناسایی به معنی ساخت یک مدل ریاضی از سیستم و تخمین پارامترهای مدل در نظر گرفته شده با استفاده از داده‌های نمونه‌برداری شده می‌باشد؛ به نحوی که اگر یک سیگنال ورودی مشابه به سیستم و مدل شبیه‌سازی شده اعمال شود، خروجی های سیستم اصلی و مدل تقریباً یکسان باشند [۱-۳]. این حوزه از تحقیق ریشه در ریاضیات، مهندسی کنترل و پردازش سیگنال دارد. در فرآیند شناسایی سیستم، پس از نمونه‌برداری (جمع‌آوری داده‌ها) یک مدل انتخاب می‌گردد. پس از تخمین پارامترهای مدل، در مرحله ارزیابی میزان موفقیت آن بررسی می‌شود. در صورت عدم موفقیت در کسب آزمون‌های ارزیابی، مدل دیگری انتخاب می‌شود تا با داده‌های نمونه‌برداری تطبیق بهتری داشته باشد [۴-۵]. این فرآیند دارای چالش‌های مختلفی می‌باشد که در ادامه توضیح داده خواهد شد.

یکی از چالش‌های اساسی در شناسایی و بدست آوردن اطلاعات مجهول از سیستم، انتخاب مدل‌ها و الگوریتم‌های عددی مناسب می‌باشد [۶]. مدل سبب می‌گردد تا کمبود اطلاعات درباره سیستم تا حدی جبران شود. پس از انتخاب مدل، مسئله اصلی تخمین پارامترها با استفاده از اطلاعات در دسترس می‌باشد. انتخاب صحیح مدل و سرعت اجرای تخمین پارامترها نیز فاکتورهای بسیار مهمی در مسائل شناسایی سیستم می‌باشند [۶]. اگر مدل به درستی انتخاب نگردد، حتی اگر نمونه‌ها به درستی برآورده گردند نتیجه قابل اعتماد نخواهد بود [۷]. با توجه به ماهیت داده‌های نمونه‌برداری شده، مدل‌های گوناگون می‌توانند

¹ Real Time

² Online Identification

³ Computational Complexity

⁴ Entanglement

⁵ Parallelism

شور⁶ و گراور⁷ اولین الگوریتم‌های ارائه شده در این حوزه می‌باشند [۱۰]. الگوریتم شور برای تجزیه یک عدد به عوامل اول ارائه شده است. پیچیدگی محاسباتی این الگوریتم از مرتبه چندجمله‌ای است؛ در صورتی که پیچیدگی محاسباتی بهترین روش کلاسیک از مرتبه نمایی می‌باشد. الگوریتم گراور برای جستجو در یک پایگاه داده نامنظم بکار می‌رود. پیچیدگی محاسباتی این الگوریتم برای یک مجموعه با n عضو از مرتبه $O(\sqrt{n})$ است؛ در صورتی که پیچیدگی محاسباتی بهترین روش کلاسیک $O(n)$ می‌باشد.

الگوریتم‌های کوانتومی در مسائل مختلفی می‌توانند کاربرد داشته باشند. در بسیاری از مسائل پایه‌ای شناسایی همچون روش حداقل مربعات معمولی⁸ (OLS)، محاسبه شبه‌معکوس ماتریس داده (هنگامی که تعداد داده‌های نمونه برداری زیاد است) بسیار زمان‌بر خواهد بود. در روش حداقل مربعات معمولی در محاسبه پارامترهای سیستم فرض بر این است که نویز حاصل از اندازه‌گیری و خطای ساختار سفید می‌باشد. برای محاسبه بدون بایاس پارامترهای سیستم با نویز رنگی باید از روش‌های پیشرفته‌تری همچون روش حداقل مربعات تعمیم داده شده استفاده شود. در مسائل مختلف شناسایی سیستم همچون شناسایی با استفاده از روش حداقل مربعات تعمیم داده شده، انتظار می‌رود که استفاده از محاسبات کوانتومی سبب افزایش سرعت قابل قبولی نسبت به روش‌های کلاسیک شود. به همین منظور در این مقاله قصد داریم تا از خواص مفید محاسبات کوانتومی در مبحث افزایش سرعت و کاهش بار محاسباتی روش حداقل مربعات تعمیم داده شده استفاده نماییم. بنابراین، در این مقاله هدف شناسایی پارامترهای یک مدل با ارائه روش

حداقل مربعات تعمیم داده شده با بکارگیری محاسبات کوانتومی است. در این روش کلیه سیستم‌هایی که به فرم رگرسیون خطی بیان می‌گردند قابل شناسایی می‌باشند.

روش مرسوم حداقل مربعات دارای محدودیت‌های جدی می‌باشد. از جمله محدودیت‌های این روش این است که در رابطه $y = \Phi\theta + v$ ، برای رسیدن به پارامترهای بدون بایاس، v باید نویز سفید باشد. در رابطه فوق $\theta \in R^p, \gamma \in R^n, \Phi_{n \times p}$ می‌باشند. اما با استفاده از تعمیم‌هایی از روش OLS (همچون GLS⁹) می‌توان محدودیت نویز سفید را برطرف نمود. در روش OLS اگر پارامترهای بهینه با استفاده از رابطه $\theta^* = \Phi^+ \gamma$ محاسبه شود، هزینه محاسباتی $O(p^2n + n^2p)$ است که در داده‌های بزرگ بسیار کند می‌باشد [۱۱]. از طرفی برای شناسایی بهتر سیستم نیاز به نمونه‌های بیشتری می‌باشد که طبق رابطه فوق پیچیدگی محاسباتی بسیار زیاد خواهد بود. بنابراین یافتن یک الگوریتم سریع که به صورت تقریبی $\tilde{x} \simeq x^*$ را برآورده نماید بسیار ارزشمند است.

با توجه به اهمیت و کاربرد روش حداقل مربعات خطا در علوم مختلف، بسیاری از محققان بر روی کاهش بار محاسباتی این روش تمرکز کرده‌اند [۱۱]. در محاسبات کلاسیک الگوریتم‌های شناخته شده‌ای وجود دارند که \tilde{x} را با خطای نسبی $\|\tilde{x} - x^*\|_2 \leq \epsilon \|x^*\|_2$ (برای هر ثابت خطای $0 < \epsilon < 1$) محاسبه می‌کنند. این نوع از الگوریتم‌ها همچون CG¹⁰ برای محاسبه ماتریس‌های رتبه پایین و ماتریس‌های تُنک از روش‌های ساده سریع‌تر هستند؛ اما برای ماتریس‌های دلخواه مربعی و پُر دارای پیچیدگی محاسباتی از مرتبه $O(n^3)$ می‌باشند [۱۲]. بنابراین در صورت دستیابی به پیچیدگی‌های محاسباتی کمتر با استفاده از

⁶ Shor's Algorithm

⁷ Grover's Algorithm

⁸ Ordinary Least Squares

⁹ Generalized Least Squares

¹⁰ Conjugate Gradient

محاسبات کوانتومی روش ارائه شده بسیار ارزشمند می‌باشد. در ادامه برخی از روش‌های کوانتومی که تا کنون پیشنهاد شده اند مرور خواهد شد.

روش مرجع [۱۲] یا (HHL), یک الگوریتم کوانتومی برای محاسبه $\theta^* = \Phi^\dagger y$ در زمان $O(\log(n)\kappa^2 s^2/\epsilon)$ ارائه داده است. این روش به عنوان پایه‌ای‌ترین روش حل معادلات خطی شناخته می‌شود. از جمله محدودیت‌های آن می‌توان به هرمیتی بودن و خوش‌حالت بودن ماتریس داده اشاره کرد. همچنین تخمین پارامترها با این روش در حضور نویز رنگی سبب بایاس دار شدن تخمین می‌گردد. روش مرجع [۱۱] از لحاظ ساختار الگوریتم به روش HHL بسیار شبیه است. در این روش مشکل هرمیتی بودن حل شده است و پیچیدگی زمانی از مرتبه $O(\log(n+p)\kappa^3 s^2/\epsilon^2)$ می‌باشد. در این روش علاوه بر تخمین پارامترها مقدار نُرم خروجی $\|x\|$ نیز (بدون افزایش پیچیدگی محاسباتی) محاسبه می‌شود. روش مرجع [۱۳] یک الگوریتم کوانتومی برای بدست آوردن مقادیر تکین در زمان $O(\text{polylog}(mn)/\delta)$ پیشنهاد داده است که δ پارامتر دقت می‌باشد. این روش مشکل تُنک بودن و هرمیتی بودن ماتریس داده را با بکارگیری ایزومتری‌های مناسب حل کرده است. یکی از مشکلات اساسی این روش عدم تعیین علامت مقادیر تکین تخمین زده شده (که در ساخت خروجی به کار می‌روند) می‌باشد. روش [۱۴] بر مبنای تخمین مقادیر تکین کوانتومی، تصحیح تخمین فاز و استفاده از ایزومتری‌های مناسب بنا نهاده شده است. این روش محدودیت‌های هرمیتی بودن و تُنک بودن را برطرف نموده است؛ اما نحوه انتخاب ایزومتری‌های مناسب در این روش از چالش‌های اساسی آن می‌باشد. در تمامی روش‌های بیان شده مشکل بایاس دار بودن تخمین پارامترها در حضور نویز رنگی و محدودیت خوش‌حالت بودن ماتریس

ورودی همچنان برقرار می‌باشد. با توجه به اهمیت روش GLS ، در این مقاله قصد داریم با بهره‌گیری از ایده‌های بیان شده در مقاله [۱۲] و [۱۳] و برخی از ایده‌های پیشنهادی، روش‌های الگوریتم کلاسیک-کوانتومی و تمام کوانتومی برای روش GLS را ارائه نماییم. روش‌های پیشنهادی در عمل کاربرد بیشتری داشته و برای نویزهای رنگی نیز قابل استفاده هستند. بنابراین این روش‌ها پارامترهای سیستم را بدون بایاس محاسبه می‌کنند. لازم به ذکر است که در روش کلاسیک-کوانتومی پیشنهادی، برای حل مشکل خوش حالت بودن ماتریس داده، از روش SVD و تجزیه اسکلت استفاده می‌شود. برای حل مشکل بایاس دار بودن پارامترها نیز از قضیه GLS استفاده خواهیم کرد. در روش تمام کوانتومی پیشنهادی نیز برای حل خوش حالت بودن ماتریس داده از تقطیع مقادیر تکین استفاده شده است. در قسمت دوم مقاله به مرور برخی از پیش‌نیازهای اصلی خواهیم پرداخت. در بخش سوم روش‌های پیشنهادی به همراه نتایج ارائه خواهند شد. در بخش چهارم شبیه‌سازی یک مسئله ساده در محیط $QISKIT$ شرکت IBM انجام شده است و بخش پنجم به نتیجه‌گیری می‌پردازد. لازم به ذکر است که در این مقاله مباحث پایه‌ای محاسبات کوانتومی همچون آشنایی با گیت‌های کوانتومی، تخمین فاز، تقویت دامنه، موازی‌سازی کوانتومی و الگوریتم‌های کوانتومی بیان نشده است و خوانندگان را به مرجع [۱۵] ارجاع می‌دهیم.

۲. پیش‌نیازها

در این بخش برخی از مباحث ریاضی لازم برای حل مشکل خوش‌حالت بودن ماتریس داده بیان می‌گردد. به همین منظور ابتدا روش SVD و سپس روش تجزیه اسکلت توضیح داده خواهد شد.

¹¹Harrow Hassidim Lloyd

تقریب رتبه پایین ماتریس در شناسایی سیستم بسیار سودمند می‌باشد. با توجه به نمونه‌برداری‌های مختلف، ماتریس داده ممکن است بدحالت شود. محاسبه معکوس یک ماتریس بدحالت با چالش‌هایی روبرو می‌باشد. استفاده از تقریب رتبه پایین یکی از روش‌های موثر رفع بدحالتی ماتریس است. تقریب یک ماتریس با رتبه پایین تر را می‌توان با استفاده از تجزیه SVD انجام داد. بنابراین هدف در این بخش تخمین ماتریس رتبه پایین B با استفاده از ماتریس A می‌باشد. طبق رابطه زیر در تقریب رتبه پایین ماتریس A با رتبه r خواهیم داشت:

$$A = [U_1|U_2] \begin{bmatrix} \sigma_1 & 0 & 0 & & 0 \\ 0 & \ddots & 0 & & 0 \\ & & \sigma_r & & 0 \\ 0 & & 0 & & 0 \end{bmatrix} \begin{bmatrix} [V_1^T] \\ [V_2^T] \end{bmatrix}, \text{rank}(A) = r. \quad (۳)$$

با توجه به تعریف دو ماتریس U و V ، تجزیه مقدار تکین ماتریس را می‌توان به صورت زیر نشان داد:

$$A = u_1 \sigma_1 v_1^t + u_2 \sigma_2 v_2^t + \dots + u_r \sigma_r v_r^t, \quad \sigma_1 > \sigma_2 > \dots > \sigma_r. \quad (۴)$$

به طور کلی ماتریس A را به صورت خطی در قالب معادلات زیر با توجه به تجزیه مقادیر تکین A نشان می‌دهیم.

$$A = U_{1a}|U_{1b}|U_{2a}|U_{2b}| \begin{bmatrix} \sigma_1 & & & & 0 \\ & \ddots & & & 0 \\ & & \sigma_k & & 0 \\ & & & \sigma_{k+1} & 0 \\ & & & & \ddots \\ & & & & & \sigma_{r+1} \\ & & & & & & 0 \end{bmatrix} \begin{bmatrix} [V_{1a}^T] \\ [V_{1b}^T] \\ [V_{2a}^T] \\ [V_{2b}^T] \end{bmatrix} \quad (۵)$$

$$A = u_1 \sigma_1 v_1^t + u_2 \sigma_2 v_2^t + \dots + u_k \sigma_k v_k^t + u_{k+1} \sigma_{k+1} v_{k+1}^t + \dots + u_r \sigma_r v_r^t \Rightarrow A = \sum_{i=1}^r a_i \phi_i$$

در رابطه (۵) σ_i ها ضرایب و $u_i v_i$ ها پایه‌های متناظر می‌باشند که بر خلاف پایه‌های ثابت سری فوریه، با استفاده از اطلاعات سیگنال به صورت تطبیقی انتخاب می‌شوند. با توجه به معادله $\sigma_1 > \sigma_2 > \dots > \sigma_r$ ضرایب اولیه نقش بیشتری در بازسازی

در این قسمت هدف معرفی روش SVD و استفاده از آن در محاسبه تقریب رتبه پایین ماتریس^{۱۲} می‌باشد. همانطور که می‌دانیم در ماتریس‌های بدحالت عدد شرط (نسبت بزرگترین به کوچکترین مقدار تکین) بسیار بزرگ است. به همین علت در روش HHL فرض بر آن است که ماتریس خوش حالت است و عدد شرط کوچک می‌باشد. اما در مسئله شناسایی سیستم بعضاً با ماتریس‌های بدحالت سروکار داریم. در این مقاله این مسئله بوسیله تقریب رتبه پایین ماتریس با استفاده از روش SVD و روش تقریب شبه اسکلت حل شده است. بنابراین روش به کار برده شده در این قسمت با جزئیات توضیح داده خواهد شد.

فرض کنید A نشان‌دهنده یک سیگنال می‌باشد. در نظریه SVD هر ماتریس را می‌توان به صورت معادله زیر تجزیه کرد:

$$A = U \Sigma V^t \quad (۱)$$

که در آن $U_{m \times m} = [u_1 \dots u_m]$ (بردار تکین چپ) و $V_{n \times n} = [v_1 \dots v_n]$ (بردار تکین راست) متعامد هستند [۱۶]. بنابراین رابطه $VV^t = V^tV = I_n$, $UU^t = U^tU = I_m$ می‌باشد. لازم به ذکر است که ستون‌های ماتریس‌های $U_{m \times m}$ و $V_{n \times n}$ به ترتیب از بردارهای ویژه متعامد یک ماتریس AA^t و A^tA تشکیل شده است. ماتریس $\Sigma_{m \times n}$ نیز شبه قطری است که مقادیر روی قطر آن مقادیر تکین ماتریس AA^t یا A^tA می‌باشند. بنابراین خواهیم داشت:

$$\Sigma_{m \times n} = \text{diag}(\sigma_1, \dots, \sigma_p), p = \min\{m, n\} \quad (۲)$$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k > 0, \sigma_{k+1} = \dots = \sigma_p = 0$$

که در آن σ_1 بزرگترین و σ_k کوچکترین مقدار منفرد غیر صفر ماتریس می‌باشد.

^{۱۲}Low Rank Approximation

اسکلت^{۱۴} و تجزیه شبه اسکلت پیشنهاد شده است که در قسمت بعد به آنها می‌پردازیم.

۳.۲. تجزیه اسکلت

ماتریس $A_{m \times n}$ را در نظر بگیرید. تقریب اسکلت این ماتریس به صورت $A = CGR$ می‌باشد که R, C به ترتیب شامل k ستون و $G = M^{-1}$ و A از ماتریس $M_{k \times k} = A(I, J)$ است که (I, J) سطر و ستون‌های انتخاب شده از ماتریس A با بُعد حداکثر می‌باشند. در رابطه فوق به دست آوردن زیرماتریس با بُعد حداکثر M بسیار دشوار است؛ بنابراین با یک زیرماتریس شبه بهینه جایگزین می‌شود [۱۹]. برای بدست آوردن این ماتریس از الگوریتم \maxvol موجود در مرجع [۲۰] استفاده می‌کنیم. برای ساختن یک تقریب اسکلت رتبه k از ماتریس A با استفاده از الگوریتم \maxvol ، مراحل زیر اجرا می‌شود.

۱- k ستون از A را با اندیس‌های

$$J = (j^{(1)}, j^{(2)}, \dots, j^{(k)})$$

ماتریس ذخیره کرده و آن را C بنامید. یعنی قرار دهید:

$$C_{m \times k} = A(:, J)$$

۲- ماتریس مناسب را با بکارگیری پروسه \maxvol

[۲۰] بر روی C بدست آورید. خروجی در این

قسمت اندیس سطرهای $(i^{(1)}, i^{(2)}, \dots, i^{(k)})$

می‌باشد.

۳- k سطر بدست آمده را در ماتریس $R_{k \times n} = A(I, :)$

قرار دهید.

۴- تجزیه اسکلت به صورت $A \approx CGR$ است که

$$G = M^{-1} \text{ و } M_{k \times k} = A(I, J) \text{ می‌باشد.}$$

اگر در ابتدای الگوریتم اندیس‌های $J = (j^{(1)}, j^{(2)}, \dots, j^{(k)})$ به

درستی انتخاب نشوند، تقریب بدست آمده ممکن است قابل

ماتریس دارند. این موضوع همچنین برای نمایش یک سیگنال توسط سری فوریه بر اساس لم ریمان-لیگ صدق می‌کند. اکنون مسئله اصلی تعیین B (تقریب رتبه پایین ماتریس) به عنوان یک مسئله بهینه‌سازی به فرم زیر است:

$$B = \underset{z}{\operatorname{argmin}} \|A - z\|_2^2 \quad \text{st. } \operatorname{rank}(z) = k. \quad (6)$$

با فرض اعمال SVD بر روی ماتریس، و با در نظر گرفتن قضیه اکارت-یونگ-میرسکی^{۱۳} که در مرجع [۱۸] موجود است، پاسخ فرم بسته برای مسئله بهینه‌سازی به صورت زیر خواهد بود:

$$\begin{aligned} \Sigma_B &= \operatorname{diag}(\sigma_1, \sigma_2, \dots, \sigma_k, 0, 0, \dots, 0) \\ B &= U \Sigma_B V^t = u_1 \sigma_1 v_1^t + u_2 \sigma_2 v_2^t + \dots + u_k \sigma_k v_k^t \Rightarrow \\ B &= U_{1a} \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_k \end{bmatrix} V_{1a}^t = U_{1a} \Sigma_B V_{1a}^t. \end{aligned} \quad (7)$$

قضیه در مرجع بالا معیاری را برای محاسبه مقدار بهینه به صورت زیر بیان می‌کند:

$$\sum_{i=1}^k \sigma_i^2 \geq \sum_{i=k+1}^p \sigma_i^2, \quad p = \min\{m, n\}. \quad (8)$$

مطابق با معیار فوق، مجموع مقادیر تکین باقی مانده باید بزرگتر

(یا مساوی) از مجموع مقادیر تقطیع شده باشد تا بتوان یک

تقریب رتبه پایین مناسب را بدست آورد. بنابراین طبق رابطه (۸)

برای حل موضوع بدحالت بودن ماتریس داده ابتدا تقریب رتبه

پایین ماتریس توسط روش SVD محاسبه می‌گردد. با این عمل

مقادیر تکین کوچک‌تر که سبب بدحالت شدن ماتریس شده‌اند

حذف می‌شوند. پیچیدگی محاسباتی روش SVD برای یک

ماتریس با سایز $n \times n$ از مرتبه $O(n^3)$ می‌باشد. بنابراین استفاده

از این روش سبب افزایش بار محاسباتی می‌گردد. به منظور

کاهش پیچیدگی محاسباتی در تخمین رتبه پایین، روش تجزیه

¹⁴ Skeleton Approximation

¹³ Eckart-Young-Mirsky

$$M^{\dagger} = (U_r S_r V_r^T)^{-1} = V_r S_r^{-1} U_r^T \quad (9)$$

در نهایت ماتریس A به صورت $A \sim CV_r S_r^{-1} U_r^T R$ تقریب زده می‌شود. مراحل توضیح داده شده به صورت الگوریتم در زیر بیان شده است.

۱- k ستون را از ماتریس A به صورت رندوم با اندیس‌های $J = (j^{(1)}, j^{(2)}, \dots, j^{(k)})$ انتخاب نمایید و آنها را در ماتریس $C_{m \times k} = A(:, J)$ قرار دهید.

۲- ماتریس مناسب را با بکارگیری پروسه maxvol $([20])$ بر روی C بدست آورید. خروجی در این قسمت اندیس سطرهای $I = (i^{(1)}, i^{(2)}, \dots, i^{(k)})$ می‌باشد. بنابراین داریم: $R_{k \times n} = A(I, :)$ و $M_{k \times k} = A(I, J)$

۳- بر روی ماتریس M روش SVD را اعمال نمایید.

۴- فرآیند تقطیع مقادیر تکین را برای مقدار r بر روی ماتریس‌های U, V, S انجام دهید.

۵- شبه معکوس M را با استفاده از رابطه $M^{\dagger} = (U_r S_r V_r^T)^{-1} = V_r S_r^{-1} U_r^T$ بدست آورید.

۶- تقریب A به صورت $A \sim CV_r S_r^{-1} U_r^T R$ خواهد بود.

پیچیدگی محاسباتی الگوریتم فوق همانند بخش قبلی به صورت $O(k^3 + rk(m+n) + mr^2 + mnr)$ محاسبه می‌گردد. عبارت k^3 اضافه شده در رابطه بالا مربوط به فرآیند SVD می‌باشد. با توجه به رابطه $k \ll m, n$ پیچیدگی محاسباتی الگوریتم فوق نسبت به مقادیر ورودی m, n از مرتبه $O(rmn)$ خواهد بود. بنابراین در این قسمت با استفاده از روش شبه اسکلت می‌توانیم مشکل بدحالت بودن ماتریس ورودی را با پیچیدگی محاسباتی کمتری حل نماییم.

اعتماد نباشد؛ بنابراین ماتریس M نیز بدحالت بوده و معکوس پذیر نخواهد بود. برای حل این مشکل روش تقریب شبه اسکلت ارائه شده است که در قسمت بعد توضیح خواهیم داد. پیچیدگی محاسباتی روش maxvol برابر با $O(mk)$ می‌باشد [20]. پیچیدگی محاسباتی الگوریتم بالا نیز به صورت $O(mk + mk + kn + k^2m + kmn + k^3)$ می‌باشد. با توجه به $k \ll m, n$ ، پیچیدگی محاسباتی الگوریتم فوق از مرتبه $O(kmn)$ خواهد بود.

۴.۲ روش تجزیه شبه اسکلت

همانطور که در بخش قبل بیان شد، ماتریس M باید معکوس پذیر باشد. شبه معکوس این ماتریس یک انتخاب مناسب برای M می‌باشد. برای ساخت تقریب شبه اسکلت ماتریس $A_{m \times n}$ با رتبه $r \leq \min(m, n)$ ، به صورت زیر عمل می‌کنیم. به جای انتخاب r ستون از ماتریس A ، $k (k > r)$ ستون تصادفی با اندیس‌های $J = (j^{(1)}, j^{(2)}, \dots, j^{(k)})$ را انتخاب می‌کنیم و آن را $C_{m \times k} = A(:, J)$ می‌نامیم. سپس از الگوریتم maxvol برای یافتن k سطر با اندیس‌های $I = (i^{(1)}, i^{(2)}, \dots, i^{(k)})$ استفاده کرده و ماتریس $R_{k \times n} = A(I, :)$ را تشکیل می‌دهیم. بنابراین، ماتریس $M_{k \times k} = A(I, J)$ و تقریب شبه اسکلت به صورت $A \sim C_{m \times k} M_{k \times k}^{-1} R_{k \times n}$ می‌باشد. به دلیل انتخاب تصادفی ستون‌ها، معکوس ماتریس M ممکن است پایدار نباشد. برای حل این مشکل، ماتریس M با استفاده از روش SVD تجزیه می‌گردد. بنابراین، $M = U_M S_M V_M^T$ و V_M ماتریس‌های متعامد می‌باشند و S_M نیز ماتریس قطری شامل k مقدار تکین $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k$ است. حال با استفاده از روش یانگ-میرسکی تقریب رتبه پایین ماتریس M محاسبه می‌گردد و به جای مقادیر تکین بسیار کوچک صفر قرار داده می‌شود. بنابراین ماتریس‌های U, V, S با r مقدار تکین تقطیع می‌شوند و شبه معکوس M از رابطه (۹) بدست می‌آید.

۵.۲. روش حداقل مربعات وزن دار^{۱۵} (WLS)

$$R_e(e) = \begin{bmatrix} E(e_1^2) & E(e_1 \cdot e_2) & \cdots & E(e_1 e_N) \\ E(e_2 \cdot e_1) & E(e_2^2) & \cdots & E(e_2 \cdot e_N) \\ \vdots & \vdots & \ddots & \vdots \\ E(e_N \cdot e_1) & E(e_N \cdot e_2) & \cdots & E(e_N^2) \end{bmatrix} \quad (11)$$

همچنین ماتریس همبستگی با تابع خود همبستگی $R(k)$ دارای رابطه زیر می‌باشد:

$$R(k) = \begin{bmatrix} R(0) & R(1) & \cdots & R(N-1) \\ R(1) & R(0) & \cdots & R(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ R(N-1) & R(N-2) & R(1) & R(0) \end{bmatrix} \quad (12)$$

اگر مدل $e_t = C(z^{-1}) \cdot v_t$ مشخص باشد از روی آن می‌توان ماتریس همبستگی را بدست آورد. فرض می‌کنیم ضرائب چند جمله‌ای $e_t = C(z^{-1})v_t = v_t + C_1 v_{t-1}$ معلوم است. بنابراین داریم:

$$R_e(e) = \sigma^2 \cdot \begin{bmatrix} 1 + C_1^2 & C_1 & \cdots & 0 \\ C_1 & 1 + C_1^2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & & C_1 & 1 + C_1^2 \end{bmatrix} \quad (13)$$

حال فرض کنید در روش حداقل مربعات، e_t نویز سفید نیست اما دینامیک نویز مشخص می‌باشد. هدف در GLS این است که نویز e_t به نحوی به نویز سفید تبدیل شود تا بتوان تخمین بدون بایاس برای θ بدست آورد. با استفاده از قضیه زیر این مشکل حل می‌شود.

قضیه ۱: اگر R_e معلوم باشد $\hat{\theta}$ بدون بایاس را می‌توان از روی روش حداقل مربعات وزن داده شده وقتی $W = R_e^{-1}$ است بدست آورد [۵].

یکی از مشکلات روش حداقل مربعات معمولی تخمین بایاس دار در حضور نویز رنگی می‌باشد. در قسمت قبل دیدیم که در صورت شناخت دینامیک نویز می‌توان به تخمین بدون بایاس دست یافت. در توضیحات روش HHL یکی از فرضیات خوش حالت بودن ماتریس A (به منظور تضمین معکوس پذیر بودن) و تُنک بودن آن می‌باشد. در این قسمت ابتدا روشی (ترکیب روش های کوانتومی و کلاسیک) ارائه خواهد شد که نسبت به

در روش حداقل مربعات وزن داده شده خطا در لحظات مختلف از ارزش یکسانی برخوردار نیست. در WLS ماتریس وزن می‌تواند هر ماتریس دلخواهی باشد و تنها شرط این است که مثبت معین باشد. با مشتق گرفتن از S نسبت به θ می‌توان رابطه پارامترهای بهینه را به صورت زیر بدست آورد:

$$\hat{\theta}_{WLS} = (U^T W U)^{-1} \cdot U^T W y. \quad (10)$$

۳. روش GLS تعمیم یافته کوانتومی

روش GLS به منظور حل مشکل تخمین پارامترها در حضور نویز رنگی ارائه شده است. در این روش فرض بر آن است که دینامیک نویز مشخص می‌باشد و تلاش می‌شود که تخمین بدون بایاس از پارامترهای سیستم ارائه شود. در شناسایی سیستم با استفاده از حداقل مربعات، با بکارگیری تابع همبستگی، ماتریس همبستگی و مدل $e_t = C(z^{-1}) \cdot v_t$ می‌توان به تخمین بدون بایاس رسید. با استفاده از روش تکراری IGLS، دینامیک نویز نیز بدست می‌آید [۵]. در این روش با فرض داشتن دینامیک نویز (مقداردهی اولیه) دینامیک سیستم شناسایی می‌شود و در بخش دوم با فرض شناختن سیستم دینامیک نویز بدست می‌آید. این مراحل به طور مداوم تکرار می‌گردد تا تخمین بهینه بدست آید. بنابراین فرض داشتن دینامیک نویز در مسئله GLS محدودیتی ایجاد نمی‌کند. با توجه به اینکه در روش GLS باید دینامیک نویز یا ماتریس همبستگی خطا (R_e) معلوم باشد، می‌توان $\hat{\theta}$ بدون بایاس را از روی روش حداقل مربعات وزن داده شده هنگامی که $W = R_e^{-1}$ بدست آورد. همانطور که می‌دانیم ماتریس همبستگی خطا به صورت زیر می‌باشد:

^{۱۵}Weighted Least Square

روش های کلاسیک دارای پیچیدگی محاسباتی کمتری می باشد. در روش پیشنهادی، فقط کفایت که ماتریس داده دارای تقریب رتبه پایین باشد. پس از تقریب رتبه پایین ماتریس توسط روش شبه اسکلت (متنی بر SVD)، مقادیر تکین که سبب بد حالت شدن ماتریس می شوند حذف می گردد. با توجه به رابطه $\hat{\theta}_{WLS} = (U^T W U)^{-1} U^T W y$ یافته کوانتومی می توانیم مسئله $Ax = b$ را با مقادیر ورودی $A = U^T W U, b = U^T W y$ حل کنیم. بر اساس قضیه ۱ با روش پیشنهادی تخمین پارامترها بدون بایاس خواهد بود. مراحل الگوریتم ترکیبی کلاسیک-کوانتومی پیشنهادی به منظور حل مسئله حداقل مربعات تعمیم داده شده کوانتومی به صورت زیر می باشد.

۱،۳. الگوریتم تمام کوانتومی حداقل مربعات تعمیم یافته GLS

در این قسمت یک الگوریتم تمام کوانتومی برای حل مسئله حداقل مربعات تعمیم یافته ارائه خواهد شد. در الگوریتم پیشنهادی، برخلاف روش قبلی مرحله اول نیز به صورت کوانتومی انجام می گردد. در این روش از مقادیر تکین به صورت مستقیم در ساخت خروجی استفاده خواهد شد. قبل از ارائه الگوریتم پیشنهادی به مرور قضیه QSVE می پردازیم.

قضیه ۲: قضیه QSVE (برگرفته از مرجع [۱۳]): فرض کنید

که $A \in R^{m \times n}$ یک ماتریس با تجزیه مقادیر تکین $A = \sum_i \sigma_i u_i v_i^T$ می باشد که در ساختار داده لم ذخیره شده است و $\delta > 0$ نیز پارامتر دقت باشد. آنگاه الگوریتم QSVE در زمان $O(\text{polylog}(mn)/\delta)$ اجرا می گردد و نگاشت $\sum_i \alpha_{v_i} |v_i\rangle |0\rangle \rightarrow \sum_i \alpha_{v_i} |v_i\rangle |\tilde{\sigma}_i\rangle$ انجام می شود که در آن $\tilde{\sigma}_i \in \sigma_i \pm \delta \|A\|_F$ می باشد. خوانندگان محترم برای اطلاعات بیشتر درباره لم ۱ و اثبات قضیه به مرجع [۱۳] مراجعه نمایند.

در الگوریتم پیشنهادی تمام کوانتومی مقاله از روش QSVE استفاده خواهیم کرد. با استفاده از این روش مشکل خوش حالت بودن ماتریس حل خواهد شد و الگوریتم توانایی حل مسائل با

ورودی ها: ماتریس A_i, b_i و $W = R_e^{-1}$

$$A_n = A_i^T W A_i, \quad b_i = A_i^T W b_i$$

$$A = \begin{bmatrix} 0 & A_n^T \\ A_n & 0 \end{bmatrix}, \quad b = \begin{bmatrix} b_n \\ 0 \end{bmatrix}, \quad x = \begin{bmatrix} 0 \\ x_n \end{bmatrix}$$

۱- با استفاده از روش شبه اسکلت تقریب رتبه پایین ماتریس A محاسبه گردد.

۲- حالت کوانتومی $|b\rangle = \sum_{i=1}^N \beta_i |v_i\rangle$ را با استفاده از روش [۲۱] آماده کنید.

۳- با استفاده از روش تخمین فاز مقادیر ویژه محاسبه گردد:

$$\sum_{i=1}^N \beta_i |v_i\rangle |\lambda_i\rangle$$

۴- با استفاده از چرخش کنترلی حالت زیر را ایجاد نمایید.

$$\sum_{i=1}^N \beta_i |v_i\rangle |\lambda_i\rangle (\sqrt{1 - \lambda_i^2} |0\rangle + \lambda_i^{-1} |1\rangle)$$

۵- فرآیند عدم محاسبه (عکس فرآیند تخمین مقدار ویژه) را برای $|\lambda_i\rangle$ انجام دهید.

۶- دامنه حالت $|1\rangle$ را $|x\rangle = \sum_{i=1}^N \beta_i \lambda_i^{-1} |v_i\rangle$ اندازه گیری نمایید.

خروجی: $|x\rangle$ (تخمین پارامترها)

در الگوریتم فوق مرحله اول به صورت کلاسیک و بقیه مراحل با تبدیل مقادیر کلاسیک به کوانتومی انجام می گردد. با توجه به پیچیدگی روش HHL در ماتریس های پر $(O(n^2 \log(n)))$ ، و روش شبه اسکلت که در بخش قبلی مورد مطالعه قرار گرفت، پیچیدگی محاسباتی الگوریتم فوق نسبت به ورودی از مرتبه

$$A_n = A_i^T W A_i, \quad b_i = A_i^T W b_i$$

$$A = \begin{bmatrix} 0 & A_n^r \\ A_n & 0 \end{bmatrix}, \quad b = \begin{bmatrix} b_n \\ 0 \end{bmatrix}, \quad x = \begin{bmatrix} 0 \\ x_n \end{bmatrix}$$

۱- حالت کوانتومی $|\psi_1\rangle = |\psi\rangle = \sum_{i=1}^n \beta_i |v_i\rangle$ را با استفاده از روش [۲۱] آماده کنید.

۲- بر روی A و $A + \alpha I$ روش SVE انجام شود. بنابراین حالت سیستم در این مرحله به صورت زیر خواهد بود.

$$|\psi_2\rangle = \sum_i \beta_i |v_i\rangle_B \|\tilde{\sigma}_i + \alpha\|_C$$

۳- یک رجیستر کمکی اضافه نمایید و در صورتی که مقدار رجیستر B بزرگتر از مقدار رجیستر C بود آن را ۱ کرده و گیت فاز شرطی را اجرا نمایید. بنابراین داریم:

$$|\psi_3\rangle = \sum_i (-1)^{v_i} \beta_i |v_i\rangle_B \|\tilde{\sigma}_i + \alpha\|_C |v_i\rangle_D$$

۴- فرآیند عدم محاسبه را بر روی رجیسترهای C,D انجام دهید. اگر مقدار رجیستر B بزرگتر از T بود یک کیوبیت کمکی با مقدار $|0\rangle$ اضافه کرده و چرخش کنترلی به حالت $(\frac{1}{2\kappa\tilde{\sigma}_i}|1\rangle + \sqrt{1 - \frac{1}{4\kappa^2\tilde{\sigma}_i^2}}|0\rangle)$ انجام گردد. بنابراین حالت سیستم در این مرحله به صورت زیر است.

$$|\psi_4\rangle = \sum_{i=1}^k (-1)^{v_i} \beta_i |v_i\rangle_B \|\tilde{\sigma}_i\|_C \left(\frac{1}{2\kappa\tilde{\sigma}_i}|1\rangle + \sqrt{1 - \frac{1}{4\kappa^2\tilde{\sigma}_i^2}}|0\rangle \right)$$

۵- فرآیند عدم محاسبه برای رجیستر B انجام گردد.

$$|\psi_5\rangle = \sum_{i=1}^k (-1)^{v_i} \beta_i |v_i\rangle \left(\frac{1}{2\kappa\tilde{\sigma}_i}|1\rangle + \sqrt{1 - \frac{1}{4\kappa^2\tilde{\sigma}_i^2}}|0\rangle \right)$$

۶- روش تقویت دامنه را بر روی حالت $|1\rangle$ انجام دهید و سپس آن یعنی $|x\rangle = \sum_{i=1}^k (-1)^{v_i} \beta_i \left(\frac{1}{2\kappa\tilde{\sigma}_i} |v_i\rangle \right)$ را اندازه گیری نمایید.

۷- پس از متناسب سازی پارامترهای واقعی برابر با $|x\rangle = \sum_{i=1}^k (-1)^{v_i} \beta_i \left(\frac{1}{\tilde{\sigma}_i} |v_i\rangle \right)$ خروجی $|x\rangle$ (تخمین پارامترها)

از آنجایی که در الگوریتم پیشنهادی ماتریس ورودی با روش بیان شده هرمیتی می‌گردد، بنابراین حالت ورودی الگوریتم برابر با $|b\rangle = \sum_{i=1}^n \beta_i |v_i\rangle$ خواهد بود. در ادامه به تحلیل خطای باند خروجی الگوریتم و پیچیدگی محاسباتی آن خواهیم پرداخت.

ماتریس غیرهرمیشن را دارا خواهد بود. با فرض تجزیه مقادیر تکین $A = \sum_i \sigma_i u_i v_i^t$ ، برای حل مسئله حداقل مربعات داریم: $e = \|Ax - b\|^2$. با فرض نمایش بردارهای x و b بر حسب پایه‌های $\{u_i\}$ و $\{v_i\}$ داریم:

$$e = \|Ax - b\|^2 = \left\| \sum_{i=1}^r \sigma_i u_i v_i^t \sum_{i=1}^n \alpha_i v_i - \sum_{i=1}^m \beta_i u_i \right\|^2 \quad (14)$$

$$= \sum_{i=1}^r (\sigma_i \alpha_i - \beta_i)^2 - \sum_{i=r+1}^m \beta_i^2$$

برای حداقل شدن رابطه (۱۶) باید $\alpha_i = \frac{\beta_i}{\sigma_i}$ باشد. بنابراین $x = \sum_{i=1}^r \frac{\beta_i}{\sigma_i} v_i$ خواهد بود. طبق فرمول بندی بیان شده در (۱۶)، برای رسیدن به تخمین بهینه در روش حداقل مربعات باید حالت کوانتومی $|x\rangle = \sum_{i=1}^r \beta_i \sigma_i^{-1} |v_i\rangle$ محاسبه شود. بنابراین در این قسمت هدف تبدیل $\sum_{i=1}^r \beta_i u_i$ به $\sum_{i=1}^r \frac{\beta_i}{\sigma_i} v_i$ می‌باشد. یکی از مشکلات مهم در قضیه QSVE عدم تعیین علامت مقادیر تکین تخمین زده شده می‌باشد. همچنین با توجه به هرمیتی بودن ماتریس ورودی داریم: $|\tilde{\sigma}_i| = |\lambda_i|$. در این مقاله به منظور حل مسئله GLS برای ماتریس‌های کلی باید علامت مقادیر تکین مشخص باشد. به همین دلیل با الهام از ایده مطرح شده در مرجع [۲۲]، به منظور تعیین علامت مقادیر تکین از روش زیر استفاده خواهیم کرد. ابتدا روش QSVE بر روی دو ماتریس A و $A' = A + \alpha I$ (که α یک اسکالر مثبت می‌باشد) اجرا می‌گردد. لازم به ذکر است که بردارهای ویژه ماتریس A' همان بردارهای ویژه ماتریس A هستند. اما مقادیر ویژه آن برابر با $\sigma_i + \alpha$ خواهد بود. برای هر $\sigma_i \geq 0$ داریم: $|\sigma_i + \alpha| = |\sigma_i| + |\alpha| \geq |\sigma_i|$ و اگر $\sigma_i \leq -\frac{\alpha}{2}$ داریم: $|\sigma_i + \alpha| \leq |\sigma_i|$. بنابراین با مقایسه دو مقدار تکین بدست آمده در این مرحله می‌توانیم علامت مقدار تکین تخمین زده شده را تعیین نماییم. الگوریتم تمام کوانتومی این مقاله به صورت زیر می‌باشد.

ورودی‌ها: ماتریس A_i ، b_i و $W = R_e^{-1}$

۱،۱،۳ تحلیل باند خطای خروجی

$$p = \sum_{i=1}^k \frac{\beta_i^2}{4\kappa^2 \tilde{\sigma}_i^2} = \frac{1}{4\kappa^2} \sum_{i=1}^k \frac{\beta_i^2}{\tilde{\sigma}_i^2} = \frac{1}{4\kappa^2} \eta \quad (18)$$

همچنین مقدار بدست آمده برای باند خطا در قسمت قبل $p \geq \frac{\varepsilon^2}{4\kappa^4}$ باشد آنگاه $\eta \geq \frac{\varepsilon^2}{\kappa^2}$ اگر $\varepsilon^2 = 4\delta^2 \|A\|_F^2 \kappa^2$ می‌باشد. بنابراین طبق مرجع [۲۰] برای تقویت دامنه و رساندن احتمال به یک مقدار ثابت (۹۹ درصد) نیاز است که مراحل قبلی به تعداد $\sqrt{1/p} = O(\kappa^2/\varepsilon)$ تکرار گردد. بنابراین پیچیدگی محاسباتی کلی الگوریتم از مرتبه لگاریتمی $O((\text{polylog}(n^2)/\delta)(\kappa^2/\varepsilon)) = O(\frac{\kappa \text{polylog}(n^2)}{\delta \|A\|_F})$ خواهد بود.

۳،۱،۳ مقایسه بین روش‌های مختلف

در جدول ۱، پیچیدگی محاسباتی (سرعت الگوریتم)، محدودیت‌های مختلف، نحوه پیاده‌سازی و کاربرد زمان واقعی روش‌های پیشنهادی مقاله با چند روش رایج کلاسیک و کوانتومی مقایسه شده است. روش کلاسیک-کوانتومی پیشنهادی به علت انجام محاسبات کلاسیک در مرحله ۱ دارای پیچیدگی نسبتاً زیادی می‌باشد. اما در مقایسه با روش‌های کلاسیک همچون تجزیه چولسکی دارای پیچیدگی کمتری است. روش تمام کوانتومی پیشنهادی این مقاله در مقایسه با روش‌های دیگر دارای پیچیدگی مناسب و شامل محدودیت‌های کمتری می‌باشد. همچنین با توجه به استفاده از قضیه ۱ در ساخت ورودی الگوریتم، پارامترهای تخمین زده شده در هر دو روش پیشنهادی دارای بایاس نخواهند بود. طبق جدول ۱، هر دو روش پیشنهادی از نظر پیاده‌سازی آسان و کاربرد زمان واقعی دارای عملکرد قابل قبولی در مقایسه با دیگر روش‌ها می‌باشند. با توجه به توضیحات داده شده روش‌های پیشنهادی این مقاله در حالی که محدودیت‌های کمتری نسبت به سایر روش‌ها دارند، پیچیدگی محاسباتی آنها قابل قبول بوده و از لحاظ پیاده‌سازی آسان و کاربرد زمان واقعی دارای عملکرد مناسبی می‌باشند.

در مرحله ۶ وقتی $|1\rangle$ مشاهده می‌گردد، حالت خروجی برابر با $|\tilde{x}\rangle = \sum_{i=1}^k \frac{\beta_i}{\tilde{\sigma}_i} |v_i\rangle$ می‌باشد. طبق توضیحات اول این بخش حالت واقعی برابر با $|x^*\rangle = \sum_{i=1}^k \frac{\beta_i}{\sigma_i} |v_i\rangle$ خواهد بود. بنابراین داریم:

$$\begin{aligned} \|\tilde{x} - x^*\|_2^2 &= \sum_{i=1}^k \left(\frac{\beta_i}{\tilde{\sigma}_i} - \frac{\beta_i}{\sigma_i} \right)^2 = \sum_{i=1}^k \frac{\beta_i^2}{\sigma_i^2} \left(1 - \frac{\tilde{\sigma}_i}{\sigma_i} \right)^2 \\ &= \sum_{i=1}^k \frac{\beta_i^2}{\sigma_i^2} \left(\frac{(\tilde{\sigma}_i - \sigma_i)^2}{\sigma_i^2} \right). \end{aligned} \quad (15)$$

با توجه به قضیه ۲، $\|\tilde{\sigma}_i - \sigma_i\| \leq \delta \|A\|_F$ خواهد بود. همچنین با توجه به شرط مرحله ۴ الگوریتم، $\tilde{\sigma}_i > T$ می‌باشند. بنابراین یک کران بالا برای رابطه فوق به صورت زیر بدست می‌آید:

$$\sum_{i=1}^k \frac{\beta_i^2}{\sigma_i^2} \left(\frac{(\tilde{\sigma}_i - \sigma_i)^2}{\sigma_i^2} \right) \leq \left(\frac{\delta \|A\|_F}{T} \right)^2. \quad (16)$$

طبق مرجع [۲۰]، یکی از انتخاب‌های مناسب $T = \frac{1}{2\kappa}$ می‌باشد. با این انتخاب داریم:

$$\|\tilde{x} - x^*\|_2^2 \leq \frac{4\delta^2 \|A\|_F^2 \kappa^2}{\varepsilon^2} \|x^*\|_2^2. \quad (17)$$

بنابراین با انتخاب مناسب پارامتر دقت δ می‌توان خطای باند خروجی را در محدوده مناسبی قرار داد.

۲،۱،۳ تحلیل پیچیدگی محاسباتی

در مرحله ۱ می‌توان با بکارگیری مراحل مرجع [۲۱]، حالت کوانتومی $|\psi_1\rangle = \sum_{i=1}^k \beta_i |v_i\rangle = |b\rangle$ را ایجاد نمود. پیچیدگی محاسباتی این مرحله از مرتبه $O(\log(n))$ می‌باشد. در مرحله ۲ با توجه به مرجع [۱۹]، پیچیدگی محاسباتی روش QSVE از مرتبه $O(\text{polylog}(n^2)/\delta)$ خواهد بود. در مرحله ۶، با انتخاب $T = \frac{1}{2\kappa}$ احتمال اصلی مشاهده حالت $|1\rangle$ برابر با $p = \sum_{i=1}^k \frac{\beta_i^2}{4\kappa^2 \tilde{\sigma}_i^2}$ می‌باشد. با در نظر گرفتن $\eta = \sum_{i=1}^k \frac{\beta_i^2}{\tilde{\sigma}_i^2}$ داریم:

۴. شبیه‌سازی

جدول ۱. مقایسه روش‌های مختلف

کاربرد زمان واقعی	پیاده‌سازی	محدودیت‌ها	پیچیدگی (سرعت اجرا بر اساس پارامترهای ورودی)	روش
دارد	متوسط	هرمیتی بودن خوش حالت بودن تنک بودن	$O(\frac{\kappa^2 s^2 \log(n)}{\epsilon})$	روش [۱۲]
دارد	متوسط	هرمیتی بودن خوش حالت بودن تنک بودن	$O(\frac{\kappa^3 s^2 \log(n)}{\epsilon^2})$	روش [۱۱]
دارد	متوسط	ندارد	$O(\frac{\kappa^2 \sqrt{r} \text{polylog}(n^2)}{\epsilon})$	روش مرجع [۱۴]
ندارد	آسان	ندارد	$O(n^3)$	روش تجزیه چولسکی (کلاسیک) (سیک)
ندارد	آسان	خوش حالت بودن تنک بودن	$O(\sqrt{\kappa s n})$	روش CG (کلاسیک)
ندارد	متوسط	ندارد	$O(n^2(\log(n) + r))$	روش پیشنهادی کلاسیک- کوانتومی
دارد	متوسط	ندارد	$O(\frac{\kappa \text{polylog}(n^2)}{\delta^2 \ A\ _F})$	روش پیشنهادی تمام کوانتومی

در این قسمت دو نمونه مثال از شبیه‌سازی الگوریتم کلاسیک - کوانتومی پیشنهادی ارائه خواهد شد. برای حل این مثال‌ها از محیط گوگل کلاب^{۱۶} استفاده می‌کنیم. برای شبیه‌سازی الگوریتم‌های کوانتومی در این محیط باید ابتدا پکیج QISKIT در این محیط نصب گردد. QISKIT یک نرم افزار منبع باز برای کار با کامپیوترهای کوانتومی در سطح مدارها، پالس‌ها و الگوریتم‌ها است. این نرم‌افزار ابزارهایی برای ساخت و دستکاری برنامه‌های کوانتومی و اجرای آن‌ها بر روی نمونه‌های اولیه دستگاه‌های کوانتومی (IBM Quantum Experience) یا شبیه‌سازهای یک رایانه کوانتومی ارائه می‌کند. این نرم‌افزار از مدل مداری برای محاسبات کوانتومی پیروی می‌کند و می‌تواند برای هر سخت افزار کوانتومی (در حال حاضر از کیوبیت‌های ابررسانا و یون‌های به دام افتاده پشتیبانی می‌کند) استفاده شود. برای شبیه‌سازی و آنالیز مدار در این نرم‌افزار باید از زبان پایتون استفاده کرد. ابتدا به عنوان مثال یک مسئله ساده که نیاز به ۴ کیوبیت دارد را در نظر می‌گیریم. فرض می‌کنیم که ماتریس‌های داده به الگوریتم کوانتومی به صورت زیر خواهند بود:

$$A = \begin{bmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{bmatrix}, \quad |b\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (19)$$

در این مثال از یک کیوبیت برای نمایش $|b\rangle$ و در نهایت برای نمایش $|x\rangle$ و از دو کیوبیت برای ذخیره‌سازی دو مقدار ویژه و از یک کیوبیت به عنوان کیوبیت کمکی استفاده خواهیم کرد. ابتدا مقادیر ویژه را محاسبه می‌کنیم تا بتوانیم یک نمایش دودویی دقیق از مقادیر ویژه مقیاس شده در رجیستر را انتخاب کنیم. با این حال، می‌دانیم که برای اجرای الگوریتم نیازی به دانش قبلی از مقادیر ویژه نیست. بنابراین داریم:

$$\lambda_1 = \frac{2}{3}, \lambda_2 = \frac{4}{3}. \quad (20)$$

در مرحله تخمین فاز خروجی به صورت دودویی و به فرم $\frac{\lambda_j t}{2\pi}$ خواهد بود. بنابراین با تنظیم $t = 2\pi \cdot \frac{3}{8}$ تخمین فاز دارای خروجی $\frac{\lambda_1 t}{2\pi} = \frac{1}{4}, \frac{\lambda_2 t}{2\pi} = \frac{1}{2}$ خواهد بود که به ترتیب به فرم $|10\rangle$ و $|01\rangle$ نوشته می‌شود. بردارهای ویژه نیز برابر با $|u_1\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}, |u_2\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ نیز در پایه ویژه ماتریس A به صورت $|b\rangle = \sum_{j=1}^2 \frac{1}{\sqrt{2}} |u_j\rangle$ نشان داده می‌شود. بنابراین حالت سیستم در مراحل مختلف الگوریتم به صورت زیر خواهد بود.

۱. آماده سازی حالت‌های کوانتومی: $|b\rangle = |0\rangle$.

¹⁶ Google Colab

$$\left\{-\frac{1}{8} + \frac{1}{8\sqrt{2}}, -\sqrt{2}, \frac{1}{\sqrt{2}}, -\frac{1}{2}\right\}, \left\{\frac{3}{8} - \frac{3}{8\sqrt{2}}, -\sqrt{2}, -\frac{1}{\sqrt{2}}, \frac{1}{2}\right\},$$

$$\left\{-\frac{1}{8} - \frac{1}{8\sqrt{2}}, \sqrt{2}, -\frac{1}{\sqrt{2}}, -\frac{1}{2}\right\}, \left\{\frac{3}{8} + \frac{3}{8\sqrt{2}}, \sqrt{2}, \frac{1}{\sqrt{2}}, \frac{1}{2}\right\}. \quad (22)$$

با توجه به داده‌های مسئله سیستم خطی به فرم زیر نمایش داده می‌شود.

$$\beta_0 - \sqrt{2}\beta_1 + \frac{1}{\sqrt{2}}\beta_2 - \frac{1}{2}\beta_3 = -\frac{1}{8} + \frac{1}{8\sqrt{2}}$$

$$\beta_0 - \sqrt{2}\beta_1 - \frac{1}{\sqrt{2}}\beta_2 + \frac{1}{2}\beta_3 = \frac{3}{8} - \frac{3}{8\sqrt{2}} \quad (23)$$

$$\beta_0 + \sqrt{2}\beta_1 - \frac{1}{\sqrt{2}}\beta_2 - \frac{1}{2}\beta_3 = -\frac{1}{8} - \frac{1}{8\sqrt{2}}$$

$$\beta_0 + \sqrt{2}\beta_1 + \frac{1}{\sqrt{2}}\beta_2 + \frac{1}{2}\beta_3 = \frac{3}{8} + \frac{3}{8\sqrt{2}}$$

سیستم فوق به فرم ماتریسی دارای نمایش زیر است:

$$\begin{bmatrix} -\sqrt{2} & 1 & \frac{1}{\sqrt{2}} & -\frac{1}{2} \\ -\sqrt{2} & 1 & -\frac{1}{\sqrt{2}} & \frac{1}{2} \\ -\sqrt{2} & -1 & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \sqrt{2} & 1 & \frac{1}{\sqrt{2}} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} -\frac{1}{8} + \frac{1}{8\sqrt{2}} \\ \frac{3}{8} - \frac{3}{8\sqrt{2}} \\ -\frac{1}{8} - \frac{1}{8\sqrt{2}} \\ \frac{3}{8} + \frac{3}{8\sqrt{2}} \end{bmatrix}. \quad (24)$$

این مسئله در حالت کلی به صورت $Ax = b$ نشان داده می‌شود. برای ایجاد ماتریس‌های ورودی به الگوریتم (شرط هرمیتی بودن)، رابطه $A^T Ax = A^T b$ را در نظر می‌گیریم. بنابراین $A^T A$ و $A^T b$ به عنوان ورودی‌های الگوریتم در نظر گرفته می‌شوند و ورودی‌های الگوریتم به صورت زیر خواهند بود.

$$A^T A = \frac{1}{4} \begin{bmatrix} 15 & 9 & 5 & -3 \\ 9 & 15 & 3 & -5 \\ 5 & 3 & 15 & -9 \\ -3 & -5 & -9 & 15 \end{bmatrix}, \quad A^T b = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} \quad (25)$$

۲. پس از تخمین فاز داریم: $\frac{1}{\sqrt{2}}|01\rangle|u_1\rangle + \frac{1}{\sqrt{2}}|10\rangle|u_2\rangle$

۳. در مرحله چرخش کنترل با تنظیم $C = \frac{1}{8}$ داریم:

$$\frac{1}{\sqrt{2}}|01\rangle|u_1\rangle \left(\sqrt{1 - \frac{(1/8)^2}{(1/4)^2}}|0\rangle + \frac{1/8}{1/4}|1\rangle \right) + \frac{1}{\sqrt{2}}|10\rangle|u_2\rangle \left(\sqrt{1 - \frac{(1/8)^2}{(1/2)^2}}|0\rangle + \frac{1/8}{1/2}|1\rangle \right) =$$

$$\frac{1}{\sqrt{2}}|01\rangle|u_1\rangle \left(\sqrt{1 - \frac{1}{4}}|0\rangle + \frac{1}{2}|1\rangle \right) + \frac{1}{\sqrt{2}}|10\rangle|u_2\rangle \left(\sqrt{1 - \frac{1}{16}}|0\rangle + \frac{1}{4}|1\rangle \right).$$

۴. بعد از اعمال عکس تخمین فاز داریم:

$$\frac{1}{\sqrt{2}}|00\rangle|u_1\rangle \left(\sqrt{1 - \frac{1}{4}}|0\rangle + \frac{1}{2}|1\rangle \right) + \frac{1}{\sqrt{2}}|00\rangle|u_2\rangle \left(\sqrt{1 - \frac{1}{16}}|0\rangle + \frac{1}{4}|1\rangle \right).$$

۵. با اندازه‌گیری کیوبیت کمکی در حالت $|1\rangle$ داریم:

$$\frac{\frac{1}{\sqrt{2}}|00\rangle|u_1\rangle \left(\frac{1}{2}|1\rangle\right) + \frac{1}{\sqrt{2}}|00\rangle|u_2\rangle \left(\frac{1}{4}|1\rangle\right)}{\sqrt{\frac{5}{32}}}$$

با دقت در رابطه بالا داریم:

$$\frac{\frac{1}{2\sqrt{2}}|u_1\rangle + \frac{1}{4\sqrt{2}}|u_2\rangle}{\sqrt{5/32}} = \frac{|x\rangle}{\|x\|}$$

بنابراین بدون استفاده از گیت اضافی می‌توان نرم $|x\rangle$ را بدست آورد که احتمال مشاهده ۱ در کیوبیت کمکی می‌باشد.

$$p(|1\rangle) = \left(\frac{1}{2\sqrt{2}}\right)^2 + \left(\frac{1}{4\sqrt{2}}\right)^2 = \frac{5}{32} = \|x\|^2$$

لازم به ذکر است که خروجی نهایی الگوریتم (جواب $|x\rangle$) برابر

$$\left(\frac{\frac{1}{2\sqrt{2}}|u_1\rangle + \frac{1}{4\sqrt{2}}|u_2\rangle}{\sqrt{5/32}}\right) \cdot \left(\frac{1}{8} \times \frac{3}{8}\right) = (1.125, 0.375)$$

قسمت ۱,۶ پیوست مقاله به منظور آشنایی بیشتر خوانندگان با نحوه شبیه‌سازی با استفاده از QISKIT، کدهای شبیه‌سازی و

توضیحات مربوطه به زبان پایتون قرار داده شده است. در مثال

دوم یک مسئله رگرسیون خطی با استفاده از روش پیشنهادی

حل می‌گردد. در این قسمت مدل در نظر گرفته شده برای

سیستم به صورت زیر می‌باشد:

$$y_i = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_{ip} + \varepsilon_i = x_i^T \beta + \varepsilon_i. \quad (21)$$

به منظور بررسی الگوریتم پیشنهادی فرض می‌کنیم مجموعه داده

نمونه‌برداری شده (۴ پارامتری) به صورت زیر می‌باشد.

و $\tau = \frac{\pi}{2^t}$ می‌باشد.

در این قسمت برای شبیه‌سازی مدار فوق از QISKIT Aer Backend استفاده شده است. همچنین برای پیاده‌سازی اپراتورهای یکانی $U = \exp(iAt/16)$ از روش GLOA استفاده شده است [۲۳]. این روش سبب می‌شود که اپراتور مورد نظر به گیت‌های یکانی ساده‌تر تجزیه شده و پیاده‌سازی آن ساده‌تر گردد. از آنجایی که الگوریتم کوانتومی مورد نظر احتمالی می‌باشد خروجی بدست آمده در شبیه‌سازی QISKIT به صورت شکل ۲ خواهد بود. در شکل ۲ نمودار آبی رنگ معادل با خروجی اصلی (محاسبات کلاسیک) و نمودار قرمز رنگ معادل با خروجی بدست آمده حاصل از الگوریتم کوانتومی پیشنهادی می‌باشد. همانطور که مشخص است. اعداد بدست آمده نسبت به اعداد واقعی دارای مقدار خطا می‌باشند که علت آن استفاده از روش GLOA در تخمین اپراتورهای یکانی مرحله تخمین فاز می‌باشد.

۵. نتیجه‌گیری

در این مقاله از محاسبات کوانتومی برای حل مسئله حداقل مربعات تعمیم داده شده GLS در شناسایی سیستم استفاده گردید. در روش‌های پیشنهادی برخلاف روش پایه حداقل مربعات HHL، نیاز نیست که ماتریس داده هرمیتی و خوش حالت باشد. همچنین با بکارگیری این الگوریتم‌ها می‌توان در حضور نویز رنگی (به شرط داشتن ماتریس همبستگی) به تخمین بدون بایاس نویز دست یافت. روش پیشنهادی ترکیبی مقاله دارای پیچیدگی محاسباتی از مرتبه $O(n^2 \log(n))$ و روش تمام کوانتومی پیشنهادی از مرتبه $O(\text{polylog}(n))$ نسبت به داده‌های ورودی می‌باشد. این در حالی است که روش‌های معمول حل حداقل مربعات از مرتبه $O(n^3)$ می‌باشند. در این مقاله روش پیشنهادی ترکیبی به منظور شبیه‌سازی و تست عملی بر روی محیط QISKIT شرکت IBM پیاده‌سازی شده است.

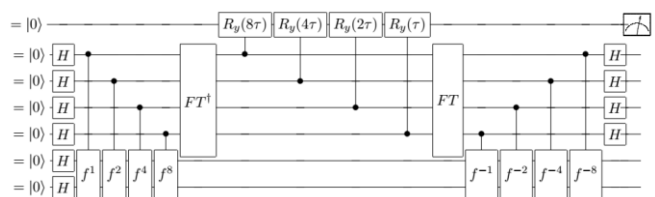
ماتریس فوق یک ماتریس هرمیتی با مقادیر ویژه $\lambda_1 = 1, \lambda_2 = 2, \lambda_3 = 4, \lambda_4 = 8$ می‌باشد. بردارهای ویژه متناظر آن‌ها به صورت زیر می‌باشد.

$$\begin{aligned} |u_1\rangle &= -|00\rangle - |01\rangle - |10\rangle + |11\rangle \\ |u_2\rangle &= |00\rangle + |01\rangle - |10\rangle + |11\rangle \\ |u_3\rangle &= |00\rangle - |01\rangle + |10\rangle + |11\rangle \\ |u_4\rangle &= -|00\rangle + |01\rangle + |10\rangle + |11\rangle \end{aligned} \quad (26)$$

همچنین $b = \left[\frac{1}{2} \quad \frac{1}{2} \quad \frac{1}{2} \quad \frac{1}{2} \right]^T$ را می‌توان به صورت $\sum_{j=1}^4 \beta_j |u_j\rangle$ با ضریب $\beta_j = \frac{1}{2}$ نوشت. بنابراین ورودی b به صورت $|b\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle$ بیان می‌گردد. مدار پیشنهادی برای حل مسئله فوق در شکل ۱ نشان داده شده است. مرحله تخمین فاز با $U = \exp(iAt/16)$ انجام می‌شود که خروجی آن به صورت رابطه (۲۹) می‌باشد.

$$\begin{aligned} & \frac{1}{2}|0001\rangle^c |u_1\rangle' + \frac{1}{2}|0010\rangle^c |u_2\rangle' + \\ & \frac{1}{2}|0100\rangle^c |u_3\rangle' + \frac{1}{2}|1000\rangle^c |u_4\rangle' \end{aligned} \quad (27)$$

این رابطه معادل با $\left| \frac{\tilde{\lambda}_j t_0}{2\pi} \right\rangle$ (با فرض $t_0 = 2\pi$) می‌باشد. لازم به ذکر است که در این مثال به علت ۴ کیوبیتی بودن رجیستر ساعت، $|\tilde{\lambda}_j\rangle = |\lambda_j\rangle$ می‌باشد. با انتخاب $C = \frac{8\pi}{2^t}$ ، پس از مرحله چرخش کنترلی خروجی به صورت $\frac{1}{\sqrt{340}}(-|00\rangle + 7|01\rangle + 11|10\rangle + 13|11\rangle)$ خواهد بود. این خروجی متناسب با جواب اصلی مسئله یعنی $\frac{1}{32}[-1 \ 7 \ 11 \ 13]^T$ می‌باشد زیرا ضرب داخلی آنها برابر با یک می‌باشد.



شکل ۱. مدار پیشنهادی برای حل مسئله. در این شکل $f = \exp(iAt/16)$

مقایسه بین روش‌های پیشنهادی و روش‌های رایج کلاسیک و کوانتومی نشان داد که روش‌های پیشنهادی مقاله علاوه بر داشتن کمترین محدودیت‌ها سبب کاهش بار محاسباتی و افزایش سرعت در حل مسائل شناسایی مبتنی بر حداقل مربعات خطا می‌گردند.

Qiskit Simulation_Sample Output:

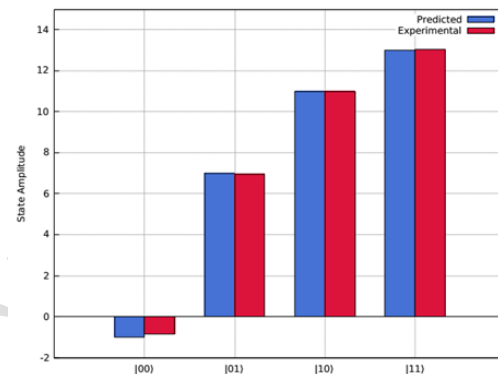
Predicted solution:

[-1 7 11 13]

Simulated experiment solution:

[-0.8425 6.9604 10.9980 13.0341]

Error: 0.1660



شکل ۲. خروجی بدست آمده در شبیه‌سازی با QISKIT

۶. پیوست‌ها

۱.۶. پیاده‌سازی مسئله در شبیه‌ساز کوانتومی QISKIT

برای حل مسئله اول بیان شده در قسمت ۴ در شبیه‌ساز کوانتومی، ابتدا پکیج QISKIT بر روی محیط گوگل کلاب نصب می‌شود. سپس ورودی‌های مسئله و نتیجه الگوریتم به صورت زیر تعریف می‌گردد.

```
import numpy as np
from linear_solvers import NumPyLinearSolver, HHL
matrix = np.array( [1,-1/3], [-1/3,1] )
vector = np.array ( [1,0] )
naive_hhl_solution = HHL().solve(matrix, vector)
```

شکل ۳. نحوه تعریف ورودی الگوریتم

با استفاده از دستور زیر ابتدا مسئله فوق را به صورت کلاسیک حل می‌کنیم.

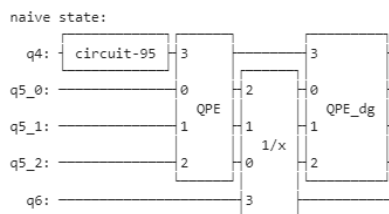
```
classical_solution = NumPyLinearSolver().solve(matrix, vector)
print('classical state:', classical_solution.state)
classical state: [1.125, 0.375]
```

شکل ۴. نحوه بدست آوردن حل کلاسیک مسئله

یکی از مشکلات الگوریتم کوانتومی HHL این است که خروجی مسئله به صورت یک حالت کوانتومی است و برای بدست آوردن تک تک المان‌های حالت باید به تعداد N پرس‌وجو انجام داد. بنابراین پیچیدگی محاسباتی الگوریتم به مقدار $O(N)$ افزایش خواهد یافت که ویژگی افزایش سرعت با این کار از بین می‌رود. بنابراین معمولاً خروجی الگوریتم به صورت تابعی از حالت $|x\rangle$ خواهد بود (برای مثال $\langle x^T | M | x \rangle$ محاسبه می‌گردد). بنابراین خروجی الگوریتم به صورت ساده^{۱۷} و بدون تغییر در نظر گرفته می‌شود. با دستور شکل ۵ خروجی ساده الگوریتم مشاهده می‌گردد.

```
print ('naive state:')
```

```
print (naive_hhl_solution.state)
```



شکل ۵. خروجی ساده الگوریتم

با استفاده از دستور زیر نرم‌جواب‌های کلاسیک و کوانتومی بدست می‌آید.

```
print ('classical Euclidean norm:', classical_solution.euclidean_norm)
```

```
print ('naive Euclidean norm: ', naive_hhl_solution.euclidean_norm)
```

```
classical Euclidean norm: 1.1858
```

```
naive Euclidean norm: 1.1858
```

شکل ۶. محاسبه نرم کلاسیک و کوانتومی

برای رسیدن به جواب نهایی (جواب غیرساده همچون جواب کلاسیک) از دستورات زیر استفاده می‌کنیم.

```
from qiskit.quantum_info import Statevector
```

¹⁷ Naive

۷. مراجع

[1] P. Eykhoff, "Identification theory: Practical implications and limitations," Measurement, vol. 2, no. 2, pp. 75–85, Apr. 1984.

doi: [https://doi.org/10.1016/0263-2241\(84\)90036-8](https://doi.org/10.1016/0263-2241(84)90036-8).

[2] L. Ljung, "System Identification," Signal Analysis and Prediction, pp. 163–173, 1998.

doi: https://doi.org/10.1007/978-1-4612-1768-8_11.

[۳] ع. قنبری سرخی، ح. حسن پور، م. فاتح، «انتخاب

ناحیه‌های کاندید در سیستم‌های تشخیص و شناسایی اشیاء»،
مجله علمی محاسبات نرم، جلد ۵۱، شماره ۲، ص. ۳۴–۴۷،
۱۳۹۵.

doi: 20.1001.1.23223707.1395.5.2.4.1

[4] J. Jaidev Vyas, Balamurugan Gopalsamy, and H. Joshi, "System Identification," SpringerBriefs in applied sciences and technology, pp. 47–51, Sep. 2018.

doi: https://doi.org/10.1007/978-981-13-2547-2_4.

[۵] م. کراری، «شناسایی سیستم». ویرایش چهارم، انتشارات
دانشگاه صنعتی امیرکبیر، ۱۳۹۲.

[۶] م. ص. محقق، «ارائه الگوریتمی جهت تسریع روش

تکرار سیاست در واریسی مدل فرآیندهای تصمیم مارکوف با
استفاده از یادگیری ماشین»،
مجله علمی محاسبات نرم، پذیرفته
شده، ۱۴۰۲.

doi: 10.22052/SCJ.2023.243360.1029

[7] S. Kalantari and M. J. Abdollahifard, "Optimization-based multiple-point geostatistics: A sparse way," Computers & Geosciences, vol. 95, pp. 85–98, Oct. 2016. doi: <https://doi.org/10.1016/j.cageo.2016.07.006>

[8] S. Kalantari, A. Madadi, M. Ramezani, and A. Hajati, "Controlling the Ground Particle Size and Ball Mill Load Based on Acoustic Signal, Quantum Computation Basis, and Least Squares Regression, Case Study: Lakan Lead-Zinc Processing Plant," International Journal of Industrial Electronics Control and Optimization, vol. 6, no. 3, pp. 205–218, Sep. 2023. doi: <https://doi.org/10.22111/ieco.2023.45981.1488>

[9] S. Kalantari, A. Madadi, and M. Ramezani, "Reconstruction of geological images based on an adaptive spatial domain filter: an example to introduce quantum computation to geosciences," vol. 57, no. 2, pp. 183–194, 2023.

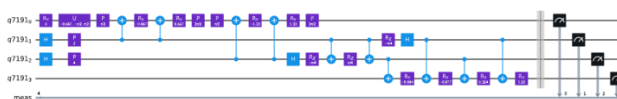
```
naive_sv = Statevector (naive_hhl_solution.state).data
naive_full_vector = np.array ([naive_sv [16], naive_sv [17] ])
print ('naive raw solution vector : ', naive_full_vector)
naive raw solution vector : [0.75+3.01063-16j  0.25+3.3695-17j]
```

```
def get_solution_vector (solution)
solution_vector = Statevector (solution.state).data[16:18].real
norm= solution.euclidean_norm
return norm= *solution_vector / np.linalg.norm(solution_vector)
print ('full naive solution vector:')
get_solution_vector (naive_hhl_solution)
print ('classical state:', classical_solution.state)
```

شکل ۷. نحوه رسیدن به جواب نهایی

طبق نتایج بدست آمده در شکل ۷ نتیجه خروجی الگوریتم کلاسیک و کوانتومی با یکدیگر برابر شده است. در شکل ۸ نیز کدهای مربوط به شبیه‌سازی مدار کوانتومی برای مثال مورد نظر قرار داده شده است.

```
from qiskit import QuantumRegister, QuantumCircuit
import numpy as np
t = 2
NUM_QUBITS = 4 # Total number of qubits
nb = 1 # Number of qubits representing the solution
nl = 2 # Number of qubits representing the eigenvalues
theta = 0 # Angle defining b>
a = 1 # Matrix diagonal
b = -1/3 # Matrix off-diagonal
# Initialize the quantum and classical registers
qr = QuantumRegister(NUM_QUBITS)
# Create a Quantum Circuit
qc = QuantumCircuit(qr)
qrb = qr[0:nb]
qrl = qr[nb:nb+nl]
qra = qr[nb+nl:nb+nl+1]
# State preparation
qc.rz(2*theta, qrb[0])
# QPE with e^{iAt}
for qu in qrl:
qc.h(qu)
qc.p(a*t, qrl[0])
qc.p(a*t*2, qrl[1])
qc.u(b*t, -np.pi/2, np.pi/2, qrb[0])
# Controlled e^{iAt} on |lambda_{1}>:
params = b*t
qc.p(np.pi/2, qrb[0])
qc.cx(qrl[0], qrb[0])
qc.ry(params, qrb[0])
qc.cx(qrl[0], qrb[0])
qc.ry(-params, qrb[0])
qc.p(3*np.pi/2, qrb[0])
# Controlled e^{2iAt} on |lambda_{2}>:
params = b*t*2
qc.p(np.pi/2, qrb[0])
qc.cx(qrl[1], qrb[0])
qc.ry(params, qrb[0])
qc.cx(qrl[1], qrb[0])
qc.ry(-params, qrb[0])
qc.p(3*np.pi/2, qrb[0])
# Inverse QFT
qc.h(qrl[1])
qc.rz(-np.pi/4, qrl[1])
qc.cx(qrl[0], qrl[1])
qc.rz(np.pi/4, qrl[1])
qc.cx(qrl[0], qrl[1])
qc.rz(-np.pi/4, qrl[0])
qc.h(qrl[0])
# Eigenvalue rotation
t1 = (-np.pi + np.pi/3 - 2*np.arcsin(1/3))/4
t2 = (-np.pi - np.pi/3 + 2*np.arcsin(1/3))/4
t3 = (np.pi - np.pi/3 - 2*np.arcsin(1/3))/4
t4 = (np.pi + np.pi/3 + 2*np.arcsin(1/3))/4
qc.cx(qrl[1], qra[0])
qc.ry(t1, qra[0])
qc.cx(qrl[0], qra[0])
qc.ry(t2, qra[0])
qc.cx(qrl[1], qra[0])
qc.ry(t3, qra[0])
qc.cx(qrl[0], qra[0])
qc.ry(t4, qra[0])
qc.measure_all()
print('Depth: {qc.depth()}')
print('CNOTS: {qc.count_ops()['cx']}')
qc.draw(fold=1)
```



شکل ۸. دستورهایی مورد نیاز برای حل مسئله و رسم مدار کوانتومی

[18] G. H. Golub, A. Hoffman, and G. W. Stewart, "A generalization of the Eckart-Young-Mirsky matrix approximation theorem," *Linear Algebra and its Applications*, vol. 88–89, pp. 317–327, Apr. 1987.

doi: [https://doi.org/10.1016/0024-3795\(87\)90114-5](https://doi.org/10.1016/0024-3795(87)90114-5).

[19] S. A. Goreinov, I. Oseledets, Dmitry Savostyanov, E. E. Тыртышников, and Н. Л. Зама́рашкин, "How to Find a Good Submatrix," *WORLD SCIENTIFIC eBooks*, pp. 247–256, Apr. 2010.

doi: https://doi.org/10.1142/9789812836021_0015

[20] N. Kishore Kumar and J. Schneider, "Literature survey on low rank approximation of matrices," *Linear and Multilinear Algebra*, vol. 65, no. 11, pp. 2212–2244, Dec. 2016.

doi: <https://doi.org/10.1080/03081087.2016.1267104>

[21] L. Grover and T. Rudolph, "Creating superpositions that correspond to efficiently integrable probability distributions," *arXiv:quant-ph/0208112*, Aug. 2002.

doi: <https://arxiv.org/abs/quant-ph/0208112>

[22] L. Wossnig, Z. Zhao, and A. Prakash, "Quantum Linear System Algorithm for Dense Matrices," *Physical Review Letters*, vol. 120, no. 5, Jan. 2018.

doi: <https://doi.org/10.1103/physrevlett.120.050502>

[23] Y. Cao, A. Daskin, S. Frankel, and S. Kais, "Quantum Circuit Design for Solving Linear Systems of Equations," *Molecular Physics*, vol. 110, no. 15–16, pp. 1675–1680, Aug. 2012.

doi: <https://doi.org/10.1080/00268976.2012.668289>

doi:<https://doi.org/10.22059/IJMGE.2023.352048.595007>

[۱۰] م. خسروی، م. ذکری، «مروری بر شبکه های عصبی کوانتومی»، *مجله علمی محاسبات نرم*، جلد ۱، شماره ۱، ص. ۴۶–۵۵، ۱۳۹۱.

doi: 20.1001.1.23223707.1391.1.1.113.0

[11] Y. Liu and S. Zhang, "Fast quantum algorithms for least squares regression and statistic leverage scores," *Theoretical Computer Science*, vol. 657, pp. 38–47, Jan. 2017.

doi: <https://doi.org/10.1016/j.tcs.2016.05.044>

[12] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum Algorithm for Linear Systems of Equations," *Physical Review Letters*, vol. 103, no. 15, Oct. 2009.

doi: <https://doi.org/10.1103/physrevlett.103.150502>

[13] I. Kerenidis and A. Prakash, "Quantum Recommendation Systems," *arXiv:1603.08675*, Sep. 2016, Available: <https://arxiv.org/abs/1603.08675>

[14] K. Li et al., "Quantum Linear System Algorithm for General Matrices in System Identification," *Entropy*, vol. 24, no. 7, p. 893, Jun. 2022, doi: <https://doi.org/10.3390/e24070893>

[15] Nielsen, M. A., & Chuang, I. L, *Quantum computation and quantum information*, Cambridge university, 2010.

doi: <https://doi.org/10.1017/CBO9780511976667>

[16] S. Kalantari, M. Ramezani, and A. Madadi, "Introducing a New Hybrid Adaptive Local Optimal Low Rank Approximation Method for Denoising Images," *International Journal of Industrial Electronics Control and Optimization*, vol. 3, no. 2, pp. 173–185, May 2020.

doi: <https://doi.org/10.22111/ieco.2019.31245.1199>

[17] S. Kalantari, M. Ramezani, A. Madadi, and V. V. Estrela, "Reduction AWGN from Digital Images Using a New Local Optimal Low-Rank Approximation Method," *Smart innovation, systems and technologies*, Dec. 2020. doi: https://doi.org/10.1007/978-3-030-57548-9_5.