# Quantum Algorithm for Linear Systems of Equations

Alfred Nguyen
*Department of Computer Science*
*Technical University of Munich*
05748 Garching, Bavaria
alfred.nguyen@outlook.com

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—component, formatting, style, styling, insert**

## I. Ease of Use

### A. Maintaining the Integrity of the Specifications

The IEEEtran class file is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

## II. Prepare Your Paper Before Styling

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not number text heads—LaTeX will do that for you.

### A. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

### B. Units

- Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as “3.5-inch disk drive”.
- Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.
- Do not mix complete spellings and abbreviations of units: “Wb/m$^2$” or “webers per square meter”, not “webers/m$^2$”. Spell out units when they appear in text: “. . . a few henries”, not “. . . a few H”.
- Use a zero before decimal points: “0.25”, not “.25”. Use “cm$^3$”, not “cc”.)

### C. Equations

Number equations consecutively. To make your equations more compact, you may use the solidus ( / ), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in:

$$a + b = \gamma \tag{1}$$

Be sure that the symbols in your equation have been defined before or immediately following

### D. LaTeX-Specific Advice

Please use “soft” (e.g., `\eqref{Eq}`) cross references instead of “hard” references (e.g., `(1)`). That will make it possible to combine sections, add equations, or change the order of figures or citations without having to go through the file line by line.

Please don’t use the `{eqnarray}` equation environment. Use `{align}` or `{IEEEeqnarray}` instead. The `{eqnarray}` environment leaves unsightly spaces around relation symbols.

## III. Template

Please note that the `{subequations}` environment in LaTeX will increment the main equation counter even when there are no equation numbers displayed. If you forget that, you might write an article in which the equation numbers skip from (17) to (20), causing the copy editors to wonder if you’ve discovered a new method of counting.

BibTeX does not work by magic. It doesn’t get the bibliographic data from thin air but from .bib files. If you use BibTeX to produce a bibliography you must send the .bib files.

## IV. Outlook

In this section we will take a look at the future outlooks of the HHL algorithm We will explore potential applications given the constraints set of the HHL algorithm. Then we will delve into different variations and optimizations which can improve the performance. Lastly, we will discuss some broad perspective, that the HHL algorithm offers in the field of quantum computing.

### *A. Applications*

The main problem wih the HHL algorithm is, that it does not output the full solution vector, but rather the solutino state $|x\rangle$. Not to be forgotten, is that the matrix $A$ has to be sparse for the HHL algorithm to work efficiently. But in some cases these constraints are not a problem and can therefore be solved by the HHL algorithm. We will now discuss some examples that utilize the algorithm.

*1) Analyzing Large Sparse Electrical Networks:* The HHL algorithm can be applied to analyze large sparse electrical networks, where we are interested in an estimate of a specific feature. Electrical networks consist of vast numbers of interconnected components, like generators, transformers, transmission lines, etc. Though, these networks have a high number of components, they have a relative low amount of connections between the each other. Thus, systems can be modeld as a graph which correspond to large sparse matrices. By being able to calculate the inverse of a matrix $A^{-1}$ efficienly, we can deduct many network properties, such as resistance. This can help us to further optimize the electrical powersystem. Conventionally, we would have to rely on iterative algorithms, which are as we saw earlier, exponentially slower in this specific case.

*2) Machine Learning (Least-Square Estimation):* In the field of machine learning the HHL algorithm can also help improve calculations. The HHL algorithm also plays a significant role in the field of machine learning, as it can improve computational intensive calculations. For example in least-square estimation, the HHL algorithm can be used to estimates of inverse matrices. This can be very helpful in data fitting tasks, enabling efficient parameter estimation and model optimization.

All in all, finding more applications for the HHL algorithm is crucial, given its specific constraints and promising capabilities.

### *B. Variations and Optimizations*

Current research and developments have been able to find variations and optimizations to the HHL algorithm. These changes have brought more efficient ways to perform the HHL algorithm and expand its applicability. We will now discuss some examples of improvements done to the HHl algorithm.

*1) Ancilla Bit Requirement:* Certain variants allow the HHL algorithm to operate without an ancilla bit. This eliminates the probability of failure in the rotation step of the eigenvalues, which leads to a faster walkthrough, as one must not repeat the process mutliple times at failure. Additionally, this increases the reliability of the algorithm.

*2) QRAM:* Quantum Random Access Memory (QRAM) is the quantum computer equivalent of the classical ram in a computer. It enables the the quantum computer to directly access quantum states for computation without having to encode the input. This would provide us with a very efficient way to load our state $|b\rangle$ into our quantum circuit.

Exploring these variations and identifying novel optimizations can contribute to the advancement and practical implementation of the HHL algorithm.

### *C. IT Security*

Although, the HHL algorithm is primarily designed for solving linear system, there are potential applications in the field of IT security. We will now discuss some examples how one could utilize the algorithm IT security.

*1) Solving Large-scale Linear Systems:* The HHL algorithm can contribute to the efficient calculation of large-scale linear systems. These are required in many IT security protocols, such as secure multi-party computation or zero-knowledge proofs.

Multi-party computation is as protocol that enables multiple different machines to computer a function, whilst not revealing their data to each other. The protocol achieves this by using linear systems as a subroutine to achieve secrecy.

Zero knowledge proofs are methods that enable a party to demonstrate certain features to another party without revealing any other information except the validity of that feature. This can be usefull to verify your identity without giving out any personal data. Oftentimes these zero knowledge proofs can be modeled in linear systems, which then can be processed by the HHL algorithm.

*2) Pattern recognition for fraud dection:* Pattern recognition for fraud detection is also a possible application for the HHL algorithm. Again, this plays in the field of machine learning. Here various big data analysis methods are used to detect fraudulant activites, which can be accelerated by the HHL algorithm.

### *D. Perspectives*

The HHL algorithm has made significant contributions, especially in the field of quantum machine learning. Though, there are no ground breaking applications so far, like the Shor's algorithm that has the ability to break the RSA encryption. Nonetheless, the ongoing discussions and research on the algorithm enable us to discover new optimizations and applications. The advances in the quantum algorithms, including the HHL algorithm, have laid the foundation for future developments in the field. But because of its generality of the HHL algorithm, it has the potential to be used as a subroutine in various domains.

While experiments have demonstrated that the HHL algorithm works on a small quantum computers with high fidelity, more advancements have to be done in the hardware. Better error correction and scaling is need to unlock the full potential of quantum computing

## V. End

test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy test wasy