Quantum simulation of negative hydrogen ion using variational quantum eigensolver on IBM quantum computer

Shubham Kumar,¹,* Rahul Pratap Singh,²† Bikash K. Behera,³,† and Prasanta K. Panigrahi³,‡

¹Department of Physics, Central University of Jharkhand, Brambe, Ranchi, India
²Indian Institute of Science Education and Research Kolkata, Mohanpur 741246, West Bengal, India
³Department of Physical Sciences, Indian Institute of Science Education and Research Kolkata, Mohanpur 741246, West Bengal, India

The negative hydrogen ion is the first three body quantum problem whose ground state energy is theoretically calculated using the “Chandrasekhar Wavefunction” that accounts for the electron-electron correlation¹. The best value of ground state energy is obtained by photodetachment experiment using lasers in the laboratory. Solving multi-body systems is a daunting task in quantum mechanics as it includes choosing a trial wavefunction and the calculation of integrals for the system that becomes almost impossible for systems with three or more particles. This difficulty can be addressed by quantum computers. They have emerged as a tool to address different electronic structure problems with remarkable efficiency. Here, we show the quantum simulation of $H^-$ ion to calculate its ground state energy in IBM quantum computer. We observe that the quantum computer is efficient in preparing the correlated wavefunction of $H^-$ and shows it as a bound entity as the ground state energy is found to be lower than that of Hydrogen atom. We use a recently developed algorithm known as “Variational Quantum Eigensolver”²,³ and implement it in IBM’s 5-qubit quantum chips “ibmqx2” and “ibmqx4”. An optimization routine is performed on a classical computer by running quantum chemistry program and codes in QISKit to converge the energy to the minimum. We also present a comparison of different optimization routines and encoding methods used to converge the energy value to the minimum. The circuit is parametrized by 12 arbitrary angles and is thus used to create different trial wavefunctions by varying the parameters. The technique can be used to solve various many body problems³ with great efficiency.

I. INTRODUCTION

Quantum simulation⁵ is a fast-growing field which promises to have profound applications in the field of condensed-matter physics, nuclear physics, quantum cosmology, quantum chemistry, and quantum biology. According to Feynman⁶, a quantum computer can deal with exponential amount of information without using exponential amount of resources. He pointed out that physical systems can be studied on quantum computers. A decade later, Lloyd proved that a quantum computer can actually be used as a universal quantum simulator⁷. Simulations of quantum systems have always been a tough job even for present generation supercomputers, due to the exponential explosion when the system size increases. Hence, the use of quantum computer for quantum simulation is essential for near term future applications. A wide range of problems in condensed-matter physics e.g., quantum phase transitions⁸, quantum magnetism⁹, in quantum chemistry e.g., calculating molecular energy values¹⁰, in nuclear physics e.g., studying atomic nucleus dynamics¹¹, in quantum biology e.g., analyzing the structure of protein and DNA¹², in quantum cosmology e.g., structuring space-time curves¹³,¹⁴ can be addressed using quantum computers. A detailed review and future applications and implications of quantum simulation can be found from these Refs.¹,⁵,¹⁵–¹⁷. Different architectures such as optical lattice,¹⁸,¹⁹ trapped ions,²⁰, nuclear spins,²¹, superconducting qubit,²² based quantum computers have been extensively used in the past for simulation of quantum systems.

Since 2016, IBM provides the composer on its website which is a cloud-based quantum computing platform²³. Any user can give a quantum circuit on the five-, and sixteen-qubit devices for a real run or simulation which is available with the help of QISKit Terra and use it by changing backend to perform a run or simulation. IBM Q Experience has now been used to perform a number of real experiments on the quantum chips. The real experiments include quantum simulation¹,²⁴–³², developing quantum algorithms³³–³⁹, testing of quantum information theoretical tasks⁴⁰,⁴¹, quantum cryptography⁴²–⁴⁵, quantum error correction⁴⁶–⁴⁹, quantum applications⁵⁰,⁵¹ to name a few.

Quantum chemistry has witnessed an upthrust in the application of quantum computers. Solving molecular problems using quantum mechanical laws makes it difficult because the interactions between large number of particles such as electrons cannot be handled by even powerful computers⁵². Recent demonstrations of molecular simulations³ have paved the way for the study of complex molecules. Simulations are limited to small molecules due to hardware limitations. Various algorithms have been developed to calculate ground and excited states and mitigation of errors. In this paper, we demonstrate the use of one such algorithm known as Variational Quantum Eigensolver (VQE) for the simulation of $H^-$ ion. Initially developed by Peruzzo and McClean², VQE is a hybrid quantum-classical algorithm that uses both quantum and classical resources to solve the eigen-
value problem.

The negative hydrogen ion is a three body system of a proton and two electrons where one electron is weakly attached to the nucleus. The structure of \( H^- \) ion is shown in Fig. 1. One of the electrons is closer to the nucleus due to repulsion from another electron. Thus, the "inner" electron feels more charge than unity and the "outer" electron feels less charge than unity. None of the electrons can remain at the same radial distance from the nucleus due to Coloumb repulsion. It has an early history of theoretical research. It attracted attention as an application of quantum mechanics to a two electron system during the early development of quantum mechanics. It has an astrophysical importance as it is found in stars because of the presence of hydrogen and low energy electrons that can form a bound structure when an electron comes in the vicinity of the hydrogen atom\(^5\). The main source of opacity in the atmosphere of the Sun at red and infrared wavelengths was predicted due to the absorption by \( H^- \) ion\(^5\). At that time it was surprising that a system such as \( H^- \), earlier believed to be unstable before the discovery of it’s bound state could be a part of the solar spectrum, essential to sustain life on earth. It was discovered by G. Patrick Flanagan in 1983 to be present in the living fluids of all living organisms. There is very little dissociated hydrogen on earth and atmosphere. It’s spectrum corresponds to infrared and visible wavelengths. Attempts were made to calculate the ground state energy of \( H^- \) using perturbation and variational methods but these methods failed. The energy was calculated to be -0.375 Hartree which is greater than the ground state energy of hydrogen atom (-0.5 Hartree). To prove that it is a bound state the energy must be less than that of hydrogen atom. The dynamics of the system is such that it breaks up into proton + electron + electron at infinity when it reaches 2-3 eV above the threshold energy\(^5\). Thus, it becomes important to consider the electron-electron correlations. The wavefunction should describe the correlations efficiently in order to get a good measure of ground state energy.

\( H^- \) was proved as a bound entity by Bethe in 1929 using Hylleraas wavefunction\(^5\). The ground state energy was further improved by Chandrasekhar\(^1\) by introducing a wavefunction of the following form,

\[
\psi(r_1, r_2) = e^{-ar_1-br_2} + e^{-br_1-ar_2} \tag{1}
\]

where \( a \) and \( b \) are variational parameters, representing the effective nuclear charges of the electrons. This wavefunction considers electron-electron correlation implicitly. The energy was found to be -0.51330 Hartree. A much better value was calculated by Chandrasekhar when he introduced electron-electron correlation explicitly in the improved wavefunction,

\[
\psi = \psi(r_1, r_2)(1 + cr_{12}) \tag{2}
\]

where \( c \) is the new variational parameter. The energy was found to be -0.52592 Hartree. Solving the Schrodinger using trial wavefunctions is a tedious task and becomes very complicated for multi-body systems because of the difficulty to guess the wavefunction that describes the system exactly. This difficulty can be addressed by quantum computers, which have been used for the simulation of various physical systems. The largest molecule to be simulated is \( H_2O\(^5\)). This was done by using variational quantum eigensolver that is used to find eigenvalues of a matrix\(^2,57,58\). VQE is better than other quantum algorithms such as phase estimation due to its high fidelity, robustness to errors and less resource requirement. It is based on the variational principle,

\[
\frac{|\langle \psi |H| \psi \rangle|}{|\langle \psi |\psi \rangle|} \geq E \tag{3}
\]

where \( E \) is the minimum energy eigenvalue. The variational principle ensures that this expectation value is always greater than the smallest eigenvalue of \( H \). Here, we report the calculation of the ground state energy of \( H^- \) ion using VQE.

## II. RESULTS

The graphical representations of the results using various optimization methods obtained using Qiskit are shown.

The results are better than that calculated from the Hartree fock method (-0.375 Hartree). They converge to -0.468070601028 for Cobyla (simulator), -0.407087502741 for Cobyla (real processor, ibmqx2), -0.46513997401 using Powell (simulator) and -0.467324316239 for Nelder-Mead (simulator). The results do not confirm the stability of \( H^- \) as they are still greater than the ground state energy of hydrogen atom (-0.5 Hartree). We run the VQE circuit in Qiskit and calculate the expectation value of \( Z_0 I \) term. We use the same parameters of state preparation at which \( Z_0 I \) converged to calculate the expectation.
value of $IZ_1$ and $Z_0Z_1$. The probability of finding the electron in the first state is maximum. Thus, the lowest eigenvalue should be found using the set of parameters for which $Z_0J$ converges. Using Bravyi-Kitaev and Jordan-Wigner encoding the energy converged to -0.49971186 and -0.5339355468 (simulator) respectively much close to the theoretical value with an error of 0.8376%. The energy converged to a lower value than the theoretical value but it is within the error bound. The details of the QISKit codes with results and experimental data are provided in the Supplementary information.

Run results obtained from ibmqx2 and ibmqx4 using
different sets of parameters are given below.

![Image](run_results_z0l_z021_i21_ibmqx2.png)

**FIG. 4.** Initial rotations of the parameters of 1st and 2nd Qubit are set to $\pi/2$ and final rotations are set to 0. Gate depth is 1. Number of shots = 8192

![Image](run_results_z0l_z021_i21_ibmqx4.png)

**FIG. 5.** Initial and final rotations of the parameters in 1st and 2nd Qubit are set to $\pi$. Gate depth is 1. Number of shots = 8192

![Image](run_results_z0l_z021_i21_ibmqx2_2.png)

**FIG. 6.** Initial rotations of the parameters of 1st and 2nd Qubit are set to $\pi$ and final rotations are set to 0. Gate depth is 1. Number of shots = 8192

The run results (Fig.4) are obtained using ibmqx2. Initial rotations were set to $\pi/2$ and final rotations to 0 and we get the energy value -0.381156 Hartree. Setting initial and final rotations in both the qubits to $\pi$ (Fig.5), we get the energy value -0.396531 Hartree. Setting initial rotations to $\pi$ and final rotations to 0 in both qubits (Fig.6), gives the energy value -0.507891 Hartree. Using ibmqx4, when the initial and final parameters in 1st and 2nd qubit are set to $\pi$ and 0 respectively we get the energy value -0.450297.

The energy value -0.507891 Hartree is lower than that of Hydrogen atom and thus confirms the bound state of $H^-$ ion.

**A. Variance**

We calculate the variance for the run result -0.507891 Hartree (ibmqx2)

$$\text{Var}(H) = |\langle \psi | H^2 | \psi \rangle | - |\langle \psi | H | \psi \rangle |^2$$  \hspace{1cm} (4)
We first measure the variance of individual Pauli operators.

- The variance of the individual Pauli operators are then plucked in the above equation to obtain the variance.

The variance of individual Pauli operators is

$$|\langle \psi | \Delta P_i^2 | \psi \rangle| = \sum_i (|\langle \psi | P_i^2 | \psi \rangle| - |\langle \psi | P_i | \psi \rangle|^2)$$  \hspace{1cm} (5)

where $P_i$'s are individual Pauli operators. Eq.4 becomes

$$Var(H) = \sum_{ij} h_{ij}^2 (\Delta P_i^2) = 0.0870538$$  \hspace{1cm} (6)

B. Conclusion

Optimization of the VQE circuit using SciPy (Python module) produced good run results but insufficient to account for the bound state of $H^-$. This can be attributed to the possibility of energy convergence at a local minima. Run result of the VQE circuit in the IBM Q experience gave the energy value -0.507891 Hartree with an error of 3.3 % from the theoretical value obtained using Chandrasekhar wavefunction. The entangled state of two electron system (correlated wavefunction of $H^-$) is efficiently realized in the IBM Quantum Computer.

MATERIALS AND METHOD

III. THE SECOND-QUANTIZED HAMILTONIAN

The second quantized Hamiltonian for fermions is given by,

$$H = \sum_{i,j} h_{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{i,j,k,l} h_{ijkl} a_i^\dagger a_j^\dagger a_k a_l$$  \hspace{1cm} (7)

where $h_{ij}$ and $h_{ijkl}$ are one and two electron integrals given by

$$h_{ij} = \int dr_1 \chi_i^* (r_1) \left( -\frac{\nabla^2_1}{2} - \sum_{\sigma} \frac{Z}{|r_1 - R_\sigma|} \right) \chi_j (r_1)$$  \hspace{1cm} (8)

and

$$h_{ijkl} = \int dr_1 dr_2 dr_3 dr_4 \chi_i^* (r_1) \chi_j^* (r_2) \chi_k (r_3) \chi_l (r_4) \frac{1}{|r_1 - r_2|}$$  \hspace{1cm} (9)

where $\chi_i (r_1)$ is the $i$th spin orbital, $Z$ is the nuclear charge, $r_1$ is the position of the $i$th electron, $r_{12}$ is the distance between the two points $r_1$ and $r_2$, $R_\sigma$ is the position of the nucleus. $a_i^\dagger$ and $a_j$ are fermionic creation and annihilation operators that follow the anti-commutation relations,

$$\{a_i^\dagger, a_j\} = \delta_{ij}$$  \hspace{1cm} (10)

$$\{a_i, a_j\} = 0$$  \hspace{1cm} (11)

The fermionic creation operator increases the occupational number of an orbital by one and the annihilation operator decreases it by one. Using Jordan-Wigner or Bravyi-Kitaev transform, the fermionic Hamiltonian can be mapped to spin type Hamiltonian.

IV. THE JORDAN-WIGNER TRANSFORM

Jordan-Wigner transform is a second-quantized encoding method to encode fermions into qubits. The mapping\(^{59}\) is given by,

$$a_i = Q_i \otimes Z_{i-1} \otimes Z_{i-2} \ldots \otimes Z_0$$  \hspace{1cm} (12)

$$a_i^\dagger = Q_i^\dagger \otimes Z_{i-1} \otimes Z_{i-2} \ldots \otimes Z_0$$  \hspace{1cm} (13)

where,

$$Q = \frac{X + iY}{2}$$  \hspace{1cm} (14)

$$Q^\dagger = \frac{X - iY}{2}$$  \hspace{1cm} (15)

Each qubit stores the occupation number of the orbital.

V. THE PARITY TRANSFORMATION

In the parity basis the qubit stores the parity of all occupied orbitals\(^{59}\). The mapping is given by, $p_i = \sum_j [\pi_n]_{ij} f_j$. This changes the occupation number basis state to it’s corresponding parity basis state. The matrix form of $[\pi_n]_{ij}$ is given by,

$$\begin{bmatrix}
1 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 \\
1 & 1 & 1 & 1
\end{bmatrix}$$

VI. THE BRAVYI KITAEV TRANSFORM

The Bravyi Kitaev Transform is a midway between the Jordan Wigner and parity encoding. The orbitals store partial sums of occupation numbers\(^{58}\). The qubit stores the parity of the set of occupation numbers corresponding to that set of orbitals. The qubits store occupation numbers when indices are even and parity when indices are
odd. The transformation\textsuperscript{59} is given by, \( b_i = \sum_j [\beta_n]_{ij} f_j \), the matrix \([\beta_n]_{ij}\) for a 4 qubit system is,

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 1 & 1 & 1
\end{bmatrix}
\]

The relation of creation and annihilation operators\textsuperscript{59} is given by,

\[ a_i^+ = X_U(i) Q_i^\dagger \otimes Z_{\rho(i)} \quad (16) \]
\[ a_i = X_U(i) Q_i \otimes Z_{\rho(i)} \quad (17) \]

VII. HAMILTONIAN OF THE H\textsuperscript{−} ION

The \( H\textsuperscript{−} \) ion consists of two electrons, one of which is closer to the nucleus (Fig. 1). Thus, it has two states with one electron each. Thus the Hamiltonian given in Eq. (7) can be expanded using the Jordan-Wigner transform as\textsuperscript{58},

\[
H = h_{00}(a_0^\dagger a_0) + h_{11}(a_1^\dagger a_1) + \frac{1}{2} h_{0101}(a_0^\dagger a_1 a_1^\dagger a_0 a_0^\dagger) + \frac{1}{2} h_{1001}(a_1^\dagger a_1 a_0 a_0^\dagger) + \frac{1}{2} h_{1010}(a_1^\dagger a_1 a_1^\dagger a_0 a_0^\dagger) \quad (18)
\]

Using the anti-commutation relations (10) and (11), the Hamiltonian becomes,

\[
H = h_{00}(a_0^\dagger a_0) + h_{11}(a_1^\dagger a_1) + h_{0101}(a_0^\dagger a_1 a_0 a_0^\dagger) + h_{0110}(a_1^\dagger a_0 a_1 a_0^\dagger) \quad (19)
\]

where we have used the fact that, \( h_{0101} = h_{1010} \) and \( h_{0110} = h_{1010} \). Using J-W transform (13), we have,

\[
a_0^\dagger a_0 = Q_0^\dagger Q_0,
\]
\[
a_1^\dagger a_1 = (Q_1^\dagger \otimes Z_0)(Q_1 \otimes Z_0),
\]
\[
a_0^\dagger a_1 a_0 a_0^\dagger a_0 = (Q_0^\dagger)^2(Q_1^\dagger \otimes Z_0)(Q_0)(Q_1 \otimes Z_0),
\]
\[
a_1^\dagger a_1 a_0 a_0^\dagger a_0 = (Q_1^\dagger \otimes Z_0)(Z_0)(Z_0)(Q_1 \otimes Z_0) \quad (20)
\]

Making use of the tensor product relation \((A \otimes B)(C \otimes D) = (AC) \otimes (BD)\), The Hamiltonian simplifies to,

\[
H = \frac{1}{2} h_{00}(1 - Z_0) + \frac{1}{2} h_{11}(1 - Z_1) + \frac{1}{8} h_{0110}(1 - Z_0 - Z_1 + Z_0 Z_1) \quad (21)
\]

Similarly using Bravyi-Kitaev transform the Hamiltonian for \( H\textsuperscript{−} \) is

\[
H = \frac{1}{2} h_{00}(1 - Z_0) + \frac{1}{2} h_{11}(1 - Z_1 Z_0) + \frac{1}{8} h_{0110}(1 - Z_0 + Z_1 - Z_0 Z_1) \quad (22)
\]

VIII. METHOD OF SIMULATION

The method of simulation consists of 2 parts,

Quantum

- Prepare the state \( \psi \), also known as the ansatz.
- Measure the expectation value \(|\langle \psi | H | \psi \rangle|\) using algorithms such as phase estimation or variational quantum eigensolver.

Classical

- Use a classical optimizer such as Nelder-Mead, Powell, Coybala and gradient descent methods for optimization.
- Iterate until the energy converges.

A. Variational Quantum Eigensolver

The quantum part of simulation is performed using variational quantum eigensolver as the algorithm to run the quantum subroutine. VQE was first demonstrated in 2014\textsuperscript{2}. It has also been demonstrated in the Refs.\textsuperscript{62–64}. The state preparation is done by entangling the qubits using various single qubit rotation gates to produce a complex state. VQE is capable of finding the ground state energies of small molecules using low depth circuits. It is based on the Rayleigh-Ritz variational principle,

\[
|\langle \psi(\theta) | H | \psi(\theta) \rangle| \geq E_0, \quad (23)
\]

where \( E_0 \) is the ground state energy. The wave function is taken to be normalized. The circuit diagram of VQE\textsuperscript{56} for gate depth 1 is shown in Fig. 8. The circuit consists of 2 parts,

Initial state preparation (ansatz) -

- The two qubit system is given initial rotations using the gates \( U_0 \) and \( U_1 \) followed by \( CNOT \)s. The unitary gates \( U_i \) are of the form \( R_z(\theta_i)R_y(\theta_i)R_z(\theta_i) \). The parameters \( \theta_i \) have to be adjusted. This completes one layer of entanglement. The layers can
be increased to produce more complex state as required. For an n-qubit system, the number of parameters for initial rotations would be 3n(n-1) and the number of unitary gates required is given n(n-1)\(^5\). For a two-qubit system, we thus need six parameters and two unitary gates. It is then followed by final rotations \(U_2\) and \(U_3\).

**Measurement**

- The expectation values of each term in the Hamiltonian is measured one by one by introducing each Pauli term in the circuit after the state preparation as shown in Fig. 8.

**B. Optimization for Energy Convergence**

Optimization (classical part) is the last step for finding the minimum (ground state) energy. There are various optimization methods classified into two categories\(^5\).

**Direct Search Method**

Direct search algorithms do not make use of the gradient of the objective function e.g., particle swarm optimization, Nelder-Mead, Powell and Cobyia. These methods have been proven to be much better than gradient-based method. Nelder Mead is a good method for optimization. It is used in the fields of chemistry, medicine, science and technology. The method is derivative free and is used in systems where the functions are noisy and discontinuous such as parameter estimation and statistical problems. Powell method requires repeated line search minimization, which may be carried out using univariate gradient free, or gradient-based procedures. Cobyla constructs successive linear approximations of the objective function and constraints via a simplex of n+1 points (in n dimensions), and optimizes these approximations in a trust region at each step.

**Gradient-based Method**

Gradient-based methods use the gradient of the objective function. Examples are simultaneous perturbation stochastic approximation (SPSA) algorithm, and L-BFGS-B. SPSA calculates the gradient by,

\[
G(\theta) = \frac{1}{2} \left( (\psi(\theta_i + \pi/2)|H|\psi(\theta_i + \pi/2)) - (\psi(\theta_i - \pi/2)|H|\psi(\theta_i - \pi/2)) \right)
\]

(24)

where \(G(\theta)\) is the gradient. The parameters are then updated according to whether the gradient decreases or increases. L-BFGS-B minimizes a differentiable scalar function \(f(x)\) over unconstrained values of the real-vector “x”\(^6\). Gradient-based optimization converges to a local minima, thus giving poor results. They are not used for functions with large number of parameters. Direct methods are much efficient and provide good values. But they are dependent on the system. Results vary with the number of qubits and gate depth. Here, we use Nelder-Mead, Cobyla and Powell methods and compare the results to depict which method is best suited for the calculation of ground state energy of \(H^-\).

**C. Implementation**

In order to determine the ground state energy, we need the “h” values \(h_{00}, h_{11}\) and \(h_{0110}\) given in Eq. (VII). The “h” values are calculated by solving the one electron integrals (Eqs. (8) and (9)). They have been calculated in lecture 18 of the lecture series titled “Quantum Principles”\(^6\). They are given as, \(h_{00} = h_{11} = 0.5, h_{0110} = 0.625, h_{00}\) and \(h_{11}\) are equal as electrons are fermions (identical and indistinguishable particles). The expectation value of Hamiltonian is therefore,

\[
\langle \psi | H | \psi \rangle = \frac{1}{4} \langle \psi | (1 - Z_0) | \psi \rangle + \frac{1}{4} \langle \psi | (1 - Z_1) | \psi \rangle + \frac{1}{4} \langle \psi | (1 - Z_0 - Z_1 - Z_0 \cdot Z_1) | \psi \rangle
\]

(25)

The expectation of \(Z_0I, IZ_1\) and \(Z_0Z_1\) can be found by placing Z gates on qubits \(q_0\) and \(q_1\) after final rotation gates and measuring both the qubits.

In matrix form, \(Z_0I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}\)

the eigenvalues are

- +1 for \(|00\rangle\) and \(|01\rangle\)
- -1 for \(|10\rangle\) and \(|11\rangle\)

Expectation value is, \(\langle \psi | Z_0I | \psi \rangle = P_{00} + P_{01} - P_{10} - P_{11}\), where the eigenvalues have been multiplied as coefficients with respective eigenvectors. Similarly,

\(IZ_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}\)

the eigenvalues are...
the eigenvalues are

- +1 for |00⟩ and |11⟩
- -1 for |01⟩ and |10⟩

\begin{equation}
\langle \psi | IZ_1 | \psi \rangle = P_{00} - P_{01} + P_{10} - P_{11}
\end{equation}

and

\begin{equation}
Z_0 Z_1 = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & -1 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\end{equation}

The initial parameters may be chosen at random. It can be adjusted according to the measured expectation value. We depict a comparison between various optimization methods. The optimization was performed by running codes in QISKit and running program in QISKit Aqua.

**DATA AVAILABILITY**

Data are available to any reader upon reasonable request.

---

*shubhamkumar.kumar@gmail.com
†singhprataprahul97@gmail.com
‡bkb18r6025@iiserkol.ac.in
§pprasanta@iiserkol.ac.in

1. S. Chandrasekhar, Some Remarks on the Negative Hydrogen Ion and Its Absorption Coefficient, Astrophys. J. 100, 176 (1944).
2. A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O’Brien, A variational eigenvalue solver on a quantum processor, Nat. Commun. 5, 4213 (2014).
3. A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Hardware-efficient Variational Quantum Eigensolver for Small Molecules and Quantum Magnets, Nature 549, 242 (2017).
4. S. Wiesner, Simulations of Many-Body Quantum Systems by a Quantum Computer, arXiv:quant-ph/9603028.
5. I. M. Georgescu, S. Ashhab, and F. Nori, Rev. Mod. Phys. 86, 153 (2014).
6. R. P. Feynman, Simulating Physics with Computers, Int. J. Theor. Phys. 21, 467 (1982).
7. S. Lloyd, Universal Quantum Simulators, Science 273, 1073 (1996).
8. S. Sachdev, Quantum Phase Transitions, Cambridge University Press, Cambridge, U.K. (1999).
9. S. Sachdev, Quantum magnetism and criticality, Nat. Phys. 4, 173 (2008).
10. B. P. Lanyon, J. D. Whitfield, G. G. Gillett, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, A. Aspuru-Guzik, and A. G. White, Towards quantum chemistry on a quantum computer, Nat. Phys. 2, 110 (2010).
11. E. F. Dumitrescu, A. J. McCaskey, G. Hagen, G. R. Jansen, T. D. Morris, T. Papenbrock, R. C. Pooser, D. J. Dean, and P. Lougovski, Phys. Rev. Lett. 120, 210501 (2018).
12. N. Lambert, Y.-N. Chen, Y.-C. Cheng, C.-M. Li, G.-Y. Chen, and F. Nori, Quantum biology, Nat. Phys. 9, 10 (2013).
13. S. Lloyd, A theory of quantum gravity based on quantum computation, arXiv:quant-ph/0501135.
D. Ghosh, P. Agarwal, P. Pandey, B. K. Behera, and P. K. Panigrahi, Automated Error Correction in IBM Quantum Computer and Explicit Generalization, Quantum Inf. Process. 17, 153 (2018).

J. Roffe, D. Headley, N. Chancellor, D. Horsman, and V. Kendon, Protecting quantum memories using coherent parity check codes, Quantum Sci. Technol. 3 035010 (2018).

S. Satyajit, K. Srinivasan, B. K. Behera, and P. K. Panigrahi, Nondestructive discrimination of a new family of highly entangled states in IBM quantum computer, Quantum Inf. Process. 17, 212 (2018).

R. Harper and S. Flammia, Fault tolerance in the IBM Q Experience, arXiv:1806.02359.

A. Dash, S. Rout, B. K. Behera, and P. K. Panigrahi, A Verification Algorithm and Its Application to Quantum Locker in IBM Quantum Computer, arXiv:1710.05196.

U. Alvarez-Rodriguez, M. Sanz, L. Lamata, and E. Solano, Quantum Artificial Life in an IBM Quantum Computer, arXiv:1711.09442.

P. A. M. Dirac and R. H. Fowler, Quantum mechanics of many-electron systems, Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character 123, (1929).

A. R. P. Rau, The Negative Ion of Hydrogen, J. Astrophys. Astron. 17, 113 (1996).

R. Wildt, Negative ion of hydrogen and the opacity of stellar atmospheres, Astrophys. J. 90, 611 (1939).

H. Bethe, Berechnung der Elektronenaffinität des Wasserstoffs, Z phys. 57, 815 (1929).

T. Bian, D. Murphy, R. Xia, A. Daskin, and S. Kais, Quantum computing methods for electronic states of the water molecule, arXiv:1804.05453.

J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, The theory of variational hybrid quantum-classical algorithms, New J. Phys. 18, 023023 (2016).

S. McArdle, S. Endo, A. Aspuru-Guzik, S. Benjamin, and X. Yuan, Quantum computational chemistry, arXiv:1808.10402.

J. T. Seeley, M. J. Richard, and P. J. Love, The Bravyi-Kitaev transformation for quantum computation of electronic structure, J. Chem. Phys. 137, 224109 (2012).

R. Malouf, A comparison of algorithms for maximum entropy parameter estimation, Proceeding COLING-02 proceedings of the 6th conference on Natural language learning - Volume 20 Pages 1-7 20, (2002).

A. J. Shaka, Lecture 18. The Hydride Ion (Continued): Two-Electron Systems, University of California Irvine (UCI) (2014).

J.-G. Liu, Y.-H. Zhang, Y. Wan, and L. Wang, Variational Quantum Eigensolver with Fewer Qubits, arXiv:1902.02663.

R. M. Parrish, E. G. Hohenstein, P. L. McMahon, and T. J. Martinez, Quantum Computation of Electronic Transitions using a Variational Quantum Eigensolver, arXiv:1901.01234.

J. I. Colless, V. V. Ramasesh, D. Dahlen, M. S. Blok, M. E. Kimchi-Schwartz, J. R. McClean, J. Carter, W. A. de Jong, and I. Siddiqi, Computation of Molecular Spectra on a Quantum Processor with an Error-Resilient Algorithm, Phys. Rev. X 8, 011021 (2018).

M. Sisodia, A. Shukla, and A. Pathak, Experimental realization of nondestructive discrimination of Bell states using a five-qubit quantum computer, Phys. Lett. A 381, 3860 (2017).
ACKNOWLEDGMENTS

S.K. would like to thank Indian Institute of Science Education and Research Kolkata for providing hospitality during the course of the project. B.K.B. acknowledges the support of IISER-K Institute Fellowship. The authors acknowledge the support of IBM Quantum Experience for producing experimental results. The views expressed are those of the authors and do not reflect the official policy or position of IBM or the IBM Quantum Experience team. The authors thank Dheerendra Singh (IPhD Student at IISER Kolkata) for useful discussions.

AUTHOR CONTRIBUTIONS

S.K. has done theoretical analysis and developed the protocol. S.K. and B.K.B. have analyzed and designed the quantum circuits. S.K., R.P.S. and B.K.B. have implemented the circuits on IBM quantum experience platform and performed the experiments. R.P.S. has written the circuit codes, run the optimization routine and quantum chemistry program for energy convergence in QISKit Terra and QISKit Aqua. S.K. and B.K.B. contributed to the composition of the manuscript. B.K.B has supervised the project. P.K.P. thoroughly checked and reviewed the manuscript. S.K., R.P.S. and B.K.B. have completed the project under the guidance of P.K.P.

COMPETING INTERESTS

The authors declare no competing financial as well as non-financial interests.
IX. SUPPLEMENTARY INFORMATION: QISKIT CODES FOR OPTIMIZATION.

For Energy estimation $Z_0 I$, $I Z_1$ and $Z_0 Z_1$ terms of Hamiltonian respectively are coded then optimized through scipy.optimize which contain ‘powell’, ‘nelder-mead’ or ‘cobyla’. The QASM code for the same is as follows:

Code for $Z_1(Z_0 I)$:

```python
# coding: utf-8

from scipy.optimize import minimize
from qiskit import QuantumCircuit, ClassicalRegister, QuantumRegister
import numpy as np

# Create a Classical Register called "c" with 3 bits
c = ClassicalRegister(2)
qc = QuantumCircuit(q,c)

# Create a Quantum Register called "q" with 3 qubits
q = QuantumRegister(2)

# Create a C q = QuantumRegister(2)

qc.u1(theta[0], q[0])
qc.u3(theta[1], -np.pi/2, np.pi/2, q[0])
qc.u1(theta[2], q[0])
qc.cx(q[0], q[1])
qc.u1(theta[3], q[1])
qc.cx(q[1], q[0])
qc.u1(theta[4], -np.pi/2, np.pi/2, q[1])
qc.u3(theta[5], q[1])
qc.cx(q[0], q[1])
qc.u3(theta[6], q[0])
qc.u1(theta[7], q[1])
qc.u3(theta[8], -np.pi/2, np.pi/2, q[0])
qc.u3(theta[9], -np.pi/2, np.pi/2, q[1])
qc.u1(theta[10], q[0])
qc.u1(theta[11], q[1])
qc.z(q[0])
qc.measure(q[0], c[0])
qc.measure(q[1], c[1])

# print(qc)

# print(i)
shots = T # Number of shots to run the program (experiment); maximum is 8192 shots
max_credits = 3 # Maximum number of credits to spend on executions.

job_hpc = execute(qc, backend=backend, shots=shots, max_credits=max_credits)
result_hpc = job_hpc.result()
counts11 = result_hpc.get_counts(qc)

# print(counts11)
Z=0
if '00' in list(counts11):
    Z=Z+counts11['00']/T

# print(Z)
```

Code for $Z_2(I Z_1)$:

```python
# coding: utf-8

from scipy.optimize import minimize
from qiskit import QuantumCircuit, QuantumRegister
import numpy as np

# Create a Quantum Register called "q" with 3 qubits
q = QuantumRegister(2)

c = ClassicalRegister(2)

qc.u1(theta[0], q[0])
qc.u3(theta[1], -np.pi/2, np.pi/2, q[0])
qc.u1(theta[2], q[0])
qc.cx(q[0], q[1])
qc.u1(theta[3], q[1])
qc.cx(q[1], q[0])
qc.u3(theta[4], -np.pi/2, np.pi/2, q[1])
qc.u3(theta[5], q[1])
qc.cx(q[0], q[1])
qc.u3(theta[6], q[0])
qc.u1(theta[7], q[1])
qc.u3(theta[8], -np.pi/2, np.pi/2, q[0])
qc.u3(theta[9], -np.pi/2, np.pi/2, q[1])
qc.u1(theta[10], q[0])
qc.u1(theta[11], q[1])
qc.z(q[0])
qc.measure(q[0], c[0])
qc.measure(q[1], c[1])

# print(qc)
```
def Z3(theta):
    # Create a Quantum Register called "q" with 3 qubits
    q = QuantumRegister(2)
    # Create a Classical Register called "c" with 3 bits
    c = ClassicalRegister(2)
    qc = QuantumCircuit(q, c)
    qc.u1(theta[0], q[0])
    qc.u3(theta[1], -np.pi/2, np.pi/2, q[0])
    qc.u1(theta[2], q[0])
    qc.cx(q[0], q[1])
    qc.u1(theta[3], q[1])
    return Z

theta0=[0, np.pi/2, 0, 0, 0, np.pi/2, 0, 0, 0, np.pi/2, 0, 0, 0, np.pi/2, 0]
res = minimize(Z3, theta0, method='nelder-mead', options={'xtol': 1e-6, 'disp': True})
print(res)
print(res.x)
print(res.fun)
print(res.nfev)

# Code for Z3(Z0Z1):

 shots = T
 theta0=[0, np.pi/2, 0, 0, 0, np.pi/2, 0, 0, 0, np.pi/2, 0, 0, 0, np.pi/2, 0]
 res = minimize(Z3, theta0, method='nelder-mead', options={'xtol': 1e-6, 'disp': True})
 print(res)
 print(res.x)
 print(res.fun)
 print(res.nfev)

#num = 3
#n contest = QC('ibmqx4')
#max_credits = 3
#hpc = hpc(result)

The tabulated data of convergence is as shown below.
| Iterations | Theoretical | Experimental |
|------------|-------------|--------------|
| 51         | -0.52052    | -0.404236078 |
| 52         | -0.52052    | -0.4167492066|
| 53         | -0.52052    | -0.4100107529|
| 54         | -0.52052    | -0.4121749407|
| 55         | -0.52052    | -0.4123932755|
| 56         | -0.52052    | -0.4153778284|
| 57         | -0.52052    | -0.4201981629|
| 58         | -0.52052    | -0.4169934128|
| 59         | -0.52052    | -0.4075683802|
| 60         | -0.52052    | -0.4216554432|
| 61         | -0.52052    | -0.4229979668|
| 62         | -0.52052    | -0.4196147123|
| 63         | -0.52052    | -0.4253583764|
| 64         | -0.52052    | -0.4249917301|
| 65         | -0.52052    | -0.4299102527|
| 66         | -0.52052    | -0.4330937475|
| 67         | -0.52052    | -0.4257880472|
| 68         | -0.52052    | -0.4038551852|
| 69         | -0.52052    | -0.4245612122|
| 70         | -0.52052    | -0.4276692965|
| 71         | -0.52052    | -0.4244919108|
| 72         | -0.52052    | -0.4281787243|
| 73         | -0.52052    | -0.4255140557|
| 74         | -0.52052    | -0.4291707672|
| 75         | -0.52052    | -0.4413052833|
| 76         | -0.52052    | -0.4357576628|
| 77         | -0.52052    | -0.4398963891|
| 78         | -0.52052    | -0.4376459041|
| 79         | -0.52052    | -0.4434627664|
| 80         | -0.52052    | -0.4494898276|
| 81         | -0.52052    | -0.4458959721|
| 82         | -0.52052    | -0.4489489295|
| 83         | -0.52052    | -0.4483291378|
| 84         | -0.52052    | -0.4582991199|
| 85         | -0.52052    | -0.4535441657|
| 86         | -0.52052    | -0.4562919823|
| 87         | -0.52052    | -0.4583214830|
| 88         | -0.52052    | -0.4572042224|
| 89         | -0.52052    | -0.4542379097|
| 90         | -0.52052    | -0.4615284535|
| 91         | -0.52052    | -0.4637664447|
| 92         | -0.52052    | -0.4654706567|
| 93         | -0.52052    | -0.4647630364|
| 94         | -0.52052    | -0.4646672379|
| 95         | -0.52052    | -0.4587078335|
| 96         | -0.52052    | -0.4640736331|
| 97         | -0.52052    | -0.4580714646|
| 98         | -0.52052    | -0.4648104928|
| 99         | -0.52052    | -0.4557965244|
|100         | -0.52052    | -0.4643548465|

**TABLE I. For Nelder Mead**

| Iterations | Theoretical | Experimental |
|------------|-------------|--------------|
| 49         | -0.52052    | -0.4068480868|
| 50         | -0.52052    | -0.4075465596|
| Iterations | Theoretical | Experimental |
|------------|-------------|--------------|
| 101        | -0.52952    | -0.4597123945 |
| 102        | -0.52952    | -0.46014880122 |
| 103        | -0.52952    | -0.4631500663 |
| 104        | -0.52952    | -0.4630380665 |
| 105        | -0.52952    | -0.4607966444 |
| 106        | -0.52952    | -0.4640445423 |
| 107        | -0.52952    | -0.4648928415 |
| 108        | -0.52952    | -0.464328522 |
| 109        | -0.52952    | -0.464582708 |
| 110        | -0.52952    | -0.4637653123 |
| 111        | -0.52952    | -0.4627551564 |
| 112        | -0.52952    | -0.4616840943 |
| 113        | -0.52952    | -0.4639483874 |
| 114        | -0.52952    | -0.4658633996 |
| 115        | -0.52952    | -0.4652171301 |
| 116        | -0.52952    | -0.466235862 |
| 117        | -0.52952    | -0.4675969867 |
| 118        | -0.52952    | -0.464460719 |
| 119        | -0.52952    | -0.4655182454 |
| 120        | -0.52952    | -0.4639068364 |
| 121        | -0.52952    | -0.4639023139 |
| 122        | -0.52952    | -0.4643041415 |
| 123        | -0.52952    | -0.4670682595 |

**TABLE II. For Cobyla on real Chip**

| Iterations | Theoretical | Experimental |
|------------|-------------|--------------|
| 1          | -0.52952    | -0.3242458015 |
| 2          | -0.52952    | -0.312141158 |
| 3          | -0.52952    | -0.3000890929 |
| 4          | -0.52952    | -0.2012450742 |
| 5          | -0.52952    | -0.3431368385 |
| 6          | -0.52952    | -0.3354519405 |
| 7          | -0.52952    | -0.3605707275 |
| 8          | -0.52952    | -0.3092321975 |
| 9          | -0.52952    | -0.3421126117 |
| 10         | -0.52952    | -0.3823174856 |
| 11         | -0.52952    | -0.3659738851 |
| 12         | -0.52952    | -0.3070652964 |
| 13         | -0.52952    | -0.376163388 |
| 14         | -0.52952    | -0.372308508 |
| 15         | -0.52952    | -0.376021897 |
| 16         | -0.52952    | -0.2783118205 |
| 17         | -0.52952    | -0.376909787 |
| 18         | -0.52952    | -0.3461754311 |
| 19         | -0.52952    | -0.3659191554 |
| 20         | -0.52952    | -0.3757416573 |
| 21         | -0.52952    | -0.3401756666 |
| 22         | -0.52952    | -0.3805747747 |
| 23         | -0.52952    | -0.3638238258 |
| 24         | -0.52952    | -0.3789247876 |
| 25         | -0.52952    | -0.3807488887 |
| 26         | -0.52952    | -0.3915352043 |
| 27         | -0.52952    | -0.377979816 |
| 28         | -0.52952    | -0.3963448298 |
| 29         | -0.52952    | -0.375903137 |
| 30         | -0.52952    | -0.3827794955 |
| 31         | -0.52952    | -0.3805437172 |
| 32         | -0.52952    | -0.3928037149 |
| 33         | -0.52952    | -0.385634926 |
| 34         | -0.52952    | -0.391837581 |
| 35         | -0.52952    | -0.384849156 |
| 36         | -0.52952    | -0.389257216 |
| 37         | -0.52952    | -0.3888326724 |
| 38         | -0.52952    | -0.3732819491 |
| 39         | -0.52952    | -0.381783664 |
| 40         | -0.52952    | -0.3926158442 |
| 41         | -0.52952    | -0.3891585442 |
| 42         | -0.52952    | -0.3859512631 |
| 43         | -0.52952    | -0.3947456284 |
| 44         | -0.52952    | -0.3852036904 |
| 45         | -0.52952    | -0.394294666 |
| 46         | -0.52952    | -0.3850246281 |
| 47         | -0.52952    | -0.3890936958 |
| 48         | -0.52952    | -0.4008480231 |
| 49         | -0.52952    | -0.4013480612 |
| 50         | -0.52952    | -0.3944512178 |
| Iterations | Theoretical | Experimental |
|------------|-------------|--------------|
| 51         | -0.52952    | -0.3884351222 |
| 52         | -0.52952    | -0.3884351222 |
| 53         | -0.52952    | -0.3884351222 |
| 54         | -0.52952    | -0.3907311105 |
| 55         | -0.52952    | -0.3980456027 |
| 56         | -0.52952    | -0.3906448114 |
| 57         | -0.52952    | -0.3865134938 |
| 58         | -0.52952    | -0.4046388436 |
| 59         | -0.52952    | -0.4025665527 |
| 60         | -0.52952    | -0.3938826373 |
| 61         | -0.52952    | -0.3762519993 |
| 62         | -0.52952    | -0.395794006 |
| 63         | -0.52952    | -0.3963339515 |
| 64         | -0.52952    | -0.4041835107 |
| 65         | -0.52952    | -0.4079657663 |
| 66         | -0.52952    | -0.3994302984 |
| 67         | -0.52952    | -0.3969173201 |
| 68         | -0.52952    | -0.4045947654 |
| 69         | -0.52952    | -0.3955795003 |
| 70         | -0.52952    | -0.396464158 |
| 71         | -0.52952    | -0.3954576879 |
| 72         | -0.52952    | -0.3922859202 |
| 73         | -0.52952    | -0.3903440066 |
| 74         | -0.52952    | -0.3997641322 |
| 75         | -0.52952    | -0.3953649947 |
| 76         | -0.52952    | -0.391213919 |
| 77         | -0.52952    | -0.4025246453 |
| 78         | -0.52952    | -0.4028140875 |
| 79         | -0.52952    | -0.3936148616 |
| 80         | -0.52952    | -0.407056244 |
| 81         | -0.52952    | -0.3972980299 |
| 82         | -0.52952    | -0.401852009 |
| 83         | -0.52952    | -0.4013366986 |
| 84         | -0.52952    | -0.4078261972 |
| 85         | -0.52952    | -0.3935236326 |
| 86         | -0.52952    | -0.4035197528 |
| 87         | -0.52952    | -0.4009673514 |
| 88         | -0.52952    | -0.4068310896 |
| 89         | -0.52952    | -0.4011399496 |
| 90         | -0.52952    | -0.4079835523 |
| 91         | -0.52952    | -0.4008302665 |
| 92         | -0.52952    | -0.4089342389 |
| 93         | -0.52952    | -0.404329474 |
| 94         | -0.52952    | -0.4098480158 |
| 95         | -0.52952    | -0.4068971478 |
| 96         | -0.52952    | -0.3970568892 |
| 97         | -0.52952    | -0.405981684 |
| 98         | -0.52952    | -0.4170714243 |
| 99         | -0.52952    | -0.40127125846 |
| 100        | -0.52952    | -0.4028407225 |
| 101        | -0.52952    | -0.397501173 |
| 102        | -0.52952    | -0.4120374641 |
| 103        | -0.52952    | -0.4085077118 |
| 104        | -0.52952    | -0.4095649677 |
| 105        | -0.52952    | -0.4097617167 |
| 106        | -0.52952    | -0.4122036682 |
| 107        | -0.52952    | -0.4182641555 |