Molecular Geometry Generation Processes through Hybrid Quantum-Classical Generative Adversarial Networks and Python-Based Self-Consistent Field Molecular Calculations

Max Cui*
Sir Winston Churchill Secondary School
Vancouver, Canada
max.mcui@gmail.com

Adelina Chau*
Archbishop Mitty High School
San Jose, USA
adel123ballet@gmail.com

Michelle Pan*
The Quarry Lane School
Dublin, USA
michelle.pan.0987@gmail.com

Vaibhav Vaiyakarnam*
The Quarry Lane School
Dublin, USA
vaibhav.vaiyakarnam@gmail.com

Larry McMahan, Ph.D
Aspiring Scholars Directed Research Program
Fremont, USA
larry.mcmahan@asdrp.org

*These authors contributed equally.

***Abstract*—Current compound synthesis and reaction pathway determination are implemented with computational chemistry methods, but methodological improvements are needed to increase precision, accuracy, and efficiency. We are implementing a molecular geometry generation process using Hybrid Quantum-Classical Graph Generative Adversarial Networks (QNetGAN), STO-nG basis set, Hartree-Fock (HF) and Post-Hartree-Fock (PHF) approximations of Schrödinger's equation for molecular energy calculations using PySCF. QNetGAN generates molecular graphs, ensuring atom connectedness, druglikeness, and Octet Rule satisfaction, at a fast runtime and great efficiency. Although a work in progress, QNetGAN has demonstrated its capabilities with an impressive training time of 10.164 minutes and 47% success rate, generating 141/300 structurally valid molecular geometries that fulfill Lipinski's Rule of Five. Meanwhile, we have developed a Python-based molecular energy and properties program with proven accuracy. By testing pre-existing organic molecules using HF (and later, PHF) and comparing our experimental results with accepted values from the CCCBDB government database, we are verifying the reliability of our program and preparing to determine the stability and feasibility of future QNetGAN-generated molecules. Currently, the QNetGAN and the HF/PHF molecular geometry and molecular energy calculations are separate; future work involves improving the accuracy of QNetGAN before integrating these components together and the current classical GAN/Density Functional Theory (DFT) systems. The ultimate goal is to increase efficiency and lower costs in R&D processes, accelerating the development of new medicines and treatments for currently incurable diseases.**

I. QNetGAN

The Generative Adversarial Network (GAN) is a machine learning algorithm with two neural networks, a Generator and a Discriminator, that compete to optimize the production of an object. In the past, GANs like MolGAN have shown great potential in computational drug discovery [1]. Building off of existing classical GANs, researchers have integrated quantum computing to develop hybrid quantum-classical GANs [2]. In our project, we combined hybrid quantum-classical GANs with the graph-based NetGAN to design a quantum GAN for generating molecular graphs (QNetGAN) [3].

QNetGAN is trained on the PyTorch Geometric QM9 dataset containing graphical representations of 134,000 small, organic molecules [4-5]. QM9 molecules are inputted to the Quantum Generator, which uses Quantum Long Short-Term Memory cells (QLSTM) to process sequential data recurrently and better suited for generative applications. The QLSTMs use Variational Quantum Circuits (VQCs) to map qubits from a randomly initialized state to an output state via rotation gates that perform the random walks. The VQC, run on Pennylane's default.qubit.torch simulator with no noise, contains 14 qubits and has a depth of 2, resulting in a total of 28 gate operations. The final output is an adjacency matrix which is inputted to the Discriminator to determine whether the graph constructed from the matrix is realistic w.r.t the training examples. The Discriminator returns this information to the Generator, calculating its loss and updating the rotation angles via Parameter-Shift optimization. Similarly, the Discriminator determines its prediction accuracy by calculating the transition score matrix and updates its parameters using the ADAM algorithm. Training continues until the QNetGAN has converged. Afterwards, the generator is run to acquire sample adjacency matrices to be inputted to the Moleculizer.

In the Moleculizer, the initially unlabelled graph is labeled using an algorithm that assigns atom symbols to vertices based on vertex degree. Each undirected edge between two vertices represents a chemical bond between two atoms. After the graph is labeled, the resultant molecule is checked against Lipinski's Rule of Five to determine if it may be a successful

orally active drug.

Though the current molecules are valid, they can still be improved. Currently, generated graphs containing vertices with degree > 4 are not supported since no QM9 molecule forms expanded octets. The molecules also lack multiple bonds and have too few hydrogens compared to heavy atoms, both of which aren't representative of organic molecules. Nevertheless, though improvements are needed, QNetGAN has an optimistic success rate of $141/300 = 47\%$ and has shown a promising path towards more efficient and cost effective drug development processes.

II. Self-Consistent Field (SCF) Molecular Energy and Vibrational Frequency Calculations

Via the Python-C++ library PySCF, we find the molecular energies of various pre-existing molecules such as hydrocarbons and alcohols using SCF methods. These methods result in calculated molecular energy values that are then compared to official values found on the CCCBDB government database to ensure our program's reliability [6]. We have achieved an optimistic result in the percentage error of all our molecular energies calculated, at below 0.001%. After verifying the accuracy of our program, our final goal is to use our Python-based program to test the QNetGAN-generated molecules, by determining vibrational frequency, bond lengths and energies, molecular energy, electronegativity and electron density. In doing so, we can ascertain the stability of the generated molecule, and whether it is feasible for further progress in drug discovery.

Typical molecular energy and stability approximations are calculated with DFT, using electron densities instead of precise wavefunctions. However, we are aiming to obtain data with greater accuracy and precision with the recursive PHF methods, which calculate each electron's wavefunction independently, then within the field of one another, while considering correlation energy. Before implementing PHF, however, we opted to first use HF in order to test the feasibility of its runtime. If our PySCF program is able to calculate the energies using HF in a reasonably short time, then moving onto the more complicated PHF would likely be feasible. Thus, though the typical use of DFT may outperform our current HF program, we are working towards implementing PHF in our near future. Currently, the runtime is longer using HF/PHF than DFT, but it is worth compromising for two reasons: higher accuracy and the amount of resources that quantum computing saves compared to classical. HF, an extension of the Schrödinger equation for multielectron systems, begins with an initial guess (basis set) for the molecular orbital wavefunctions. The two main types of basis sets are Gaussian-Type (GTO), with a shorter runtime but lower accuracy, and Slater-Type (STO), with a longer runtime but greater accuracy [7]. In our research, we use a cc-PVTZ size 3 GTO basis set to optimize accuracy and resource efficiency. This way, we can simplify calculations by modeling an STO basis set in summing 3 GTOs for each electron. Firstly, the many-electron wavefunction is approximated as a product of each electron's orbital wavefunction in the multielectron system. Incorporating the Pauli exclusion principle, the spin states are considered by calculating the Slater determinant of the wavefunctions [8]. The analog (HF eq.) to Schrödinger's equation for a single electron spin orbital is converted into a matrix equation via the Roothaan Hall eq., enabling recursive calculations to find the converged SCF energy.

We are currently using HF because of its shorter runtime compared to PHF. However, because HF does not take electron correlation energy into consideration and because the HF short runtimes indicate a plausible implementation of PHF, we plan on continuing our research using coupled cluster correlation energy, second-order Møller-Plesset perturbation, and configuration interaction. By comparing accuracy and runtime using HF and PHF, we can obtain a comprehensive view of the SCF methods, and determine which is to be used to achieve our goal of testing the generated molecules.

III. Future Work

We will adapt the Moleculizer to account for implicit hydrogens, enable multiple bonds, and optimize generated molecules' geometries to account for electron-pair repulsion. Quantum chemistry calculations will move on to testing various PHF methods for greater accuracy and a comprehensive view of the SCF methods. Furthermore, more chemical properties will be tested with organic compounds to determine the accuracy of our program and find trends that indicate the stability of a molecule. We aim to integrate the two processes into a pipeline where scientists can input chemical properties and computational factors deemed necessary to develop a drug targeting a specific disease and generate a list of chemically-feasible molecules meeting their criteria. To make our program more user-friendly, we will also develop a GUI application. Ultimately, our research aims to reduce the time and costs needed in the drug discovery process.

References

[1] De Cao, Nicola, and Thomas Kipf. "MolGAN: An implicit generative model for small molecular graphs." arXiv preprint, 30 May 2018, arXiv:1805.11973

[2] Li, Junde, et al. "Quantum Generative Models for Small Molecule Drug Discovery." ArXiv.org, 23 Aug. 2021, arxiv.org/abs/2101.03438

[3] Bojchevski, Aleksandar, et al. "NetGAN: Generating Graphs via Random Walks." arXiv.org, 2 Mar. 2018, arxiv.org/abs/1803.00816v2.

[4] L. Ruddigkeit, R. van Deursen, L. C. Blum, J.-L. Reymond, Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17, J. Chem. Inf. Model. 52, 2864–2875, 2012.

[5] R. Ramakrishnan, P. O. Dral, M. Rupp, O. A. von Lilienfeld, Quantum chemistry structures and properties of 134 kilo molecules, Scientific Data 1, 140022, 2014.Anderson, Scott. "Visualizing Molecules with GOpenMol." Tutorial - Department of Chemistry - The University of Utah, n.d.

[6] Sun, Qiming, et al. "PySCF: the Python-based simulations of chemistry framework." Wiley Interdisciplinary Reviews: Computational Molecular Science 8.1 (2018): e1340.

[7] Magalhaes, Alexandre L. "Gaussian-type orbitals versus Slater-type orbitals: a comparison." Journal of Chemical Education 91.12 (2014): 2124-2127.

[8] Fock, Vladimir. "Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems." Zeitschrift für Physik 61.1 (1930): 126-148. Translated: Fock, Vladimir. "Approximate Method for Solving the Quantum Mechanical Multibody Problem." Journal of Physics 61.1 (1930): 126-148