Quantum Computing for Materials Simulations

quantum computer as another HPC accelerator

Karim Elgammal

QAS2024

December 4, 2024

Notes

- I came from Density Functional Theory (DFT) background
- Learning quantum computing then I can run DFT on another accelerator (QPU)
- The effort here is still in-the-making and comments are welcomed
- The quantum part here is very small! but I thought it can be good extension of DFT people to try exploring running on quantum simulations
- Priliminary Manuscript is here [1] and Implementation available on GitHub is here [2]. Comments and pull requests are welcomed!

Quantum Chemistry Simulations using Quantum Computing Algorithms

There is numerous literature work on Quantum chemistry calculations as a promising early applications of quantum computers, an example is simulating small molecules as indicated in [3]

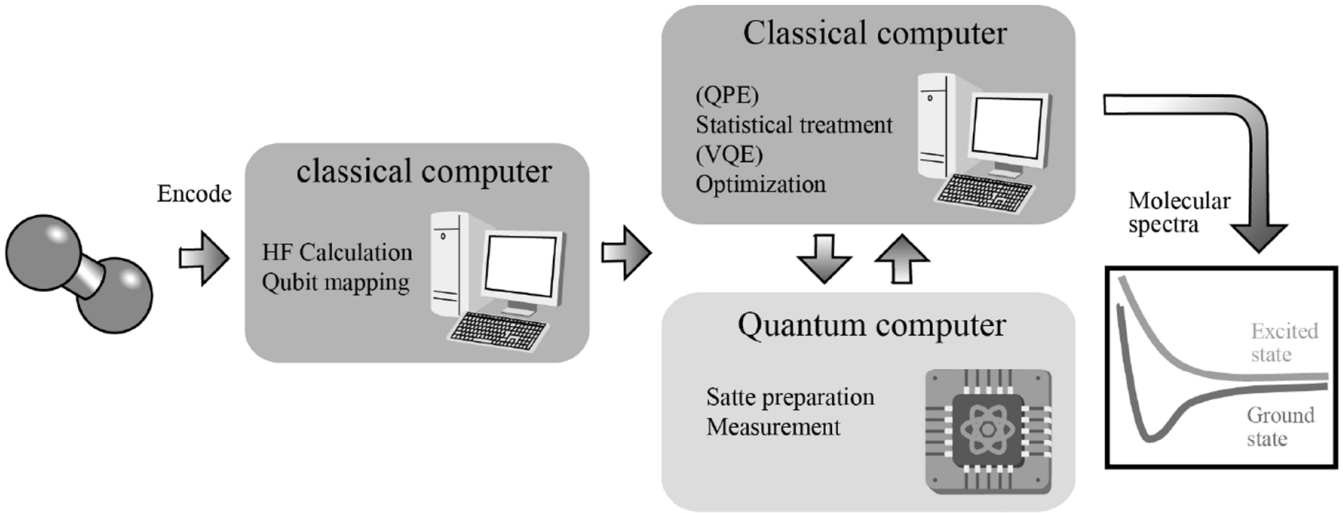


Figure: From [4]:Typical flow of quantum computing algorithm for quantum chemistry calculation

Note:

Applying this workflow beyond small molecules to larger molecules or periodic systems can be difficult at the moment due to large number of qubits needed

Materials Systems in Different Dimensions

Focus on materials science simulations like solid-state systems is picking up using quantum computing algorithms, those systems are often modeled as periodic systems and simulation cell can have hundreds of atoms and system can be 1D, 2D or 3D

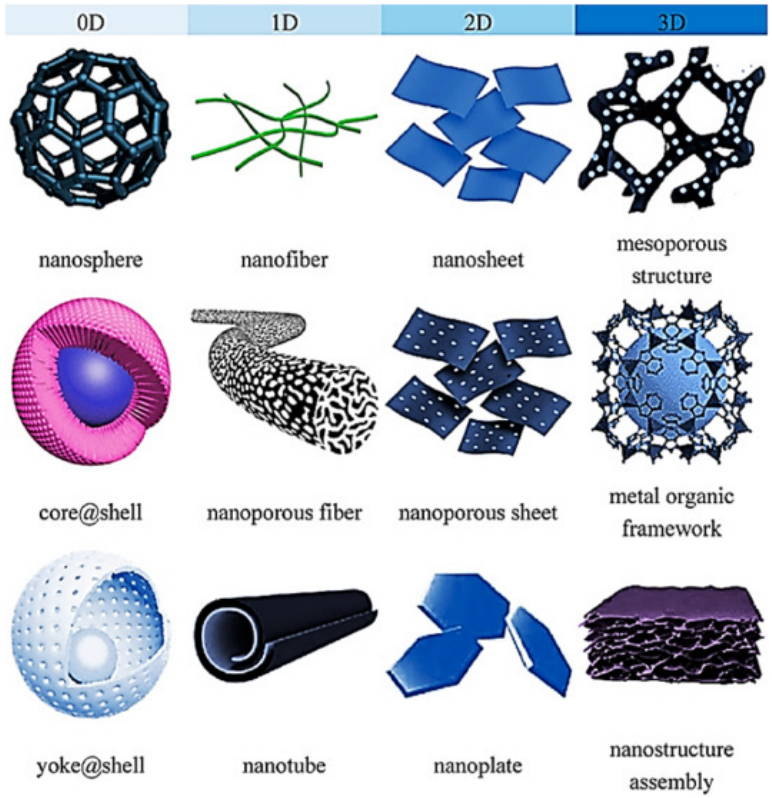


Figure: from [5]: Examples of materials systems in 1D, 2D and 3D

Computational materials science in a nutshell

- Modeling and computing materials properties is solving the electronic structure problem for the system of interest to predict a physical variable of interest
- Electronic structure of molecules and solids is the starting point in a computational materials science workflow
- Models and embedding help simplify complex quantum calculations

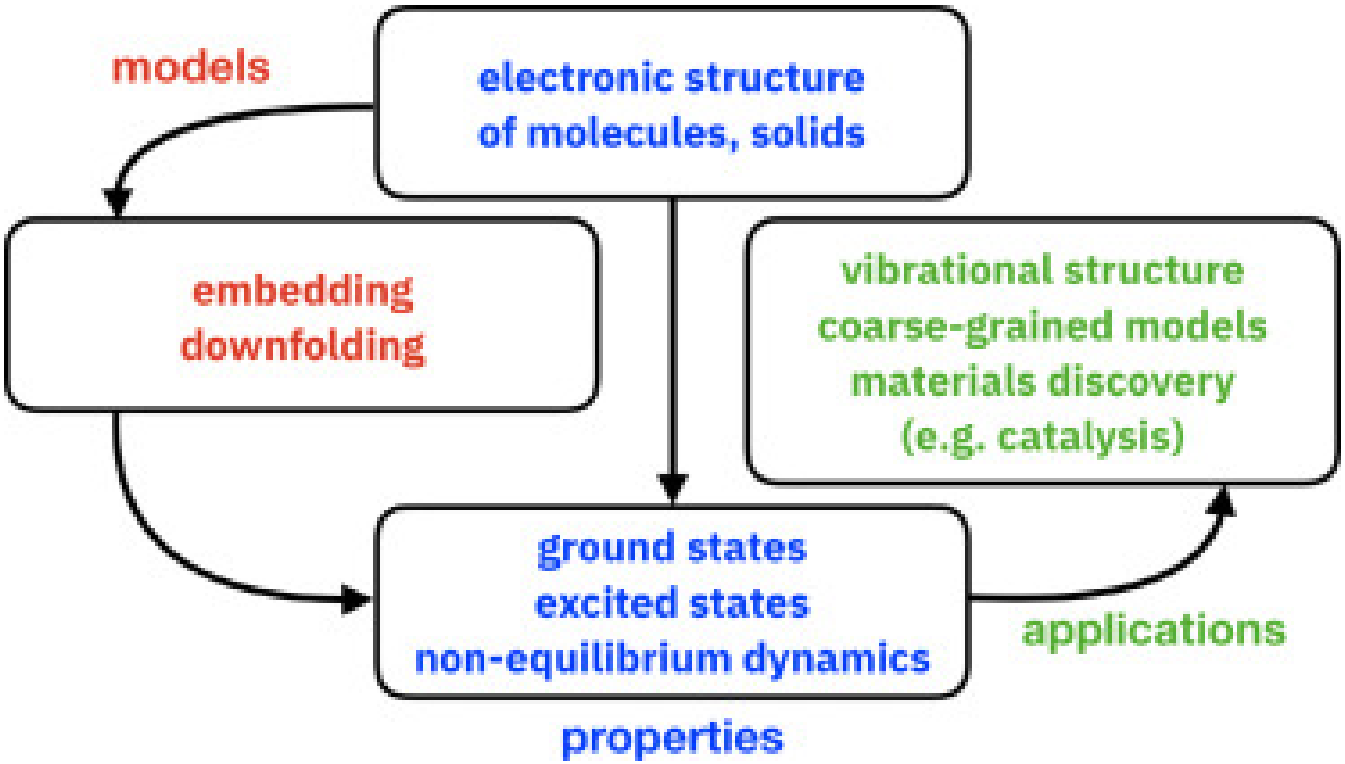
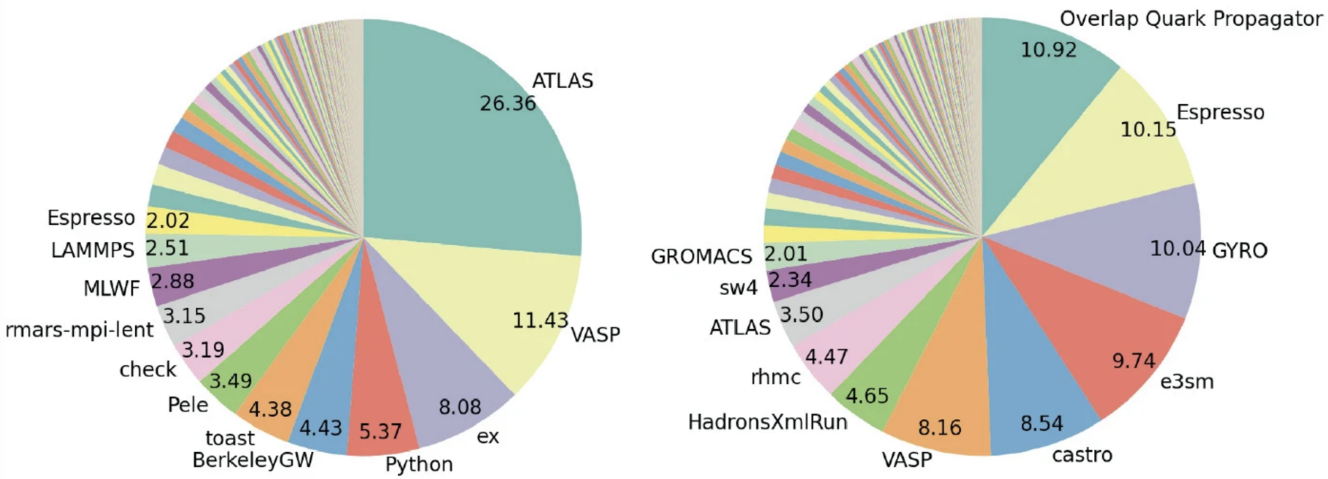


Figure: From [6]

Connection to HPC resources

- Typically those systems require many cores and nodes and fill-up big portion of HPC time, for example through Density Functional Theory (DFT) calculations (by famous codes like VASP, Quantum ESPRESSO, ABINIT, etc.)

From: Analyzing Resource Utilization in an HPC System: A Case Study of NERSC's Perlmutter



(a) CPU-only nodes. (b) GPU-accelerated nodes.

Decomposition of node-hours by applications. Infrequent applications are not labeled.

Figure: From [7]

Connection to HPC resources

- Typically those systems require many cores and nodes and fill-up big portion of HPC time, for example through Density Functional Theory (DFT) calculations (by famous codes like VASP, Quantum ESPRESSO, ABINIT, etc.)

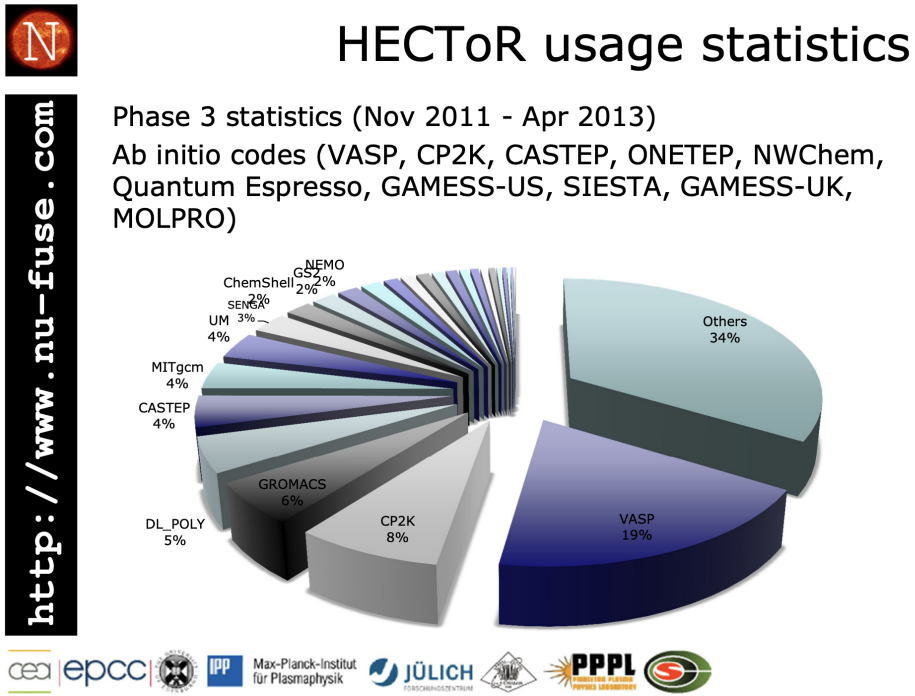


Figure: From [8]

Connection with current codes

This approach has the potential to combine the best of both worlds: the higher accuracy you can get from quantum methods and the efficiency of classical methods in accelerated material science

Quantum-centric supercomputing

The Quantum-centric supercomputers efforts taking place at the moment using quantum accelerators with current HPC infrastructures can be used to accelerate those materials science simulations

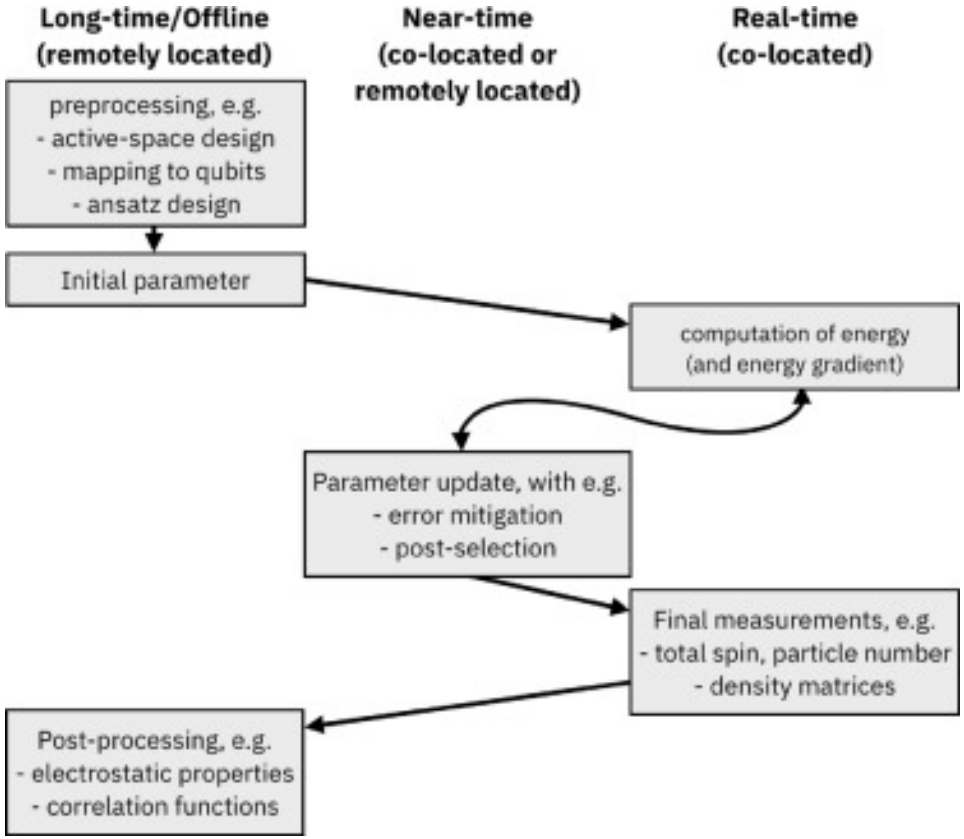


Figure: From [6]:Integration between classical (HPC) and quantum computing resources exemplified by the variational quantum eigensolver. The steps of the calculation are represented by gray blocks, connected by arrows describing the flow of operations and arranged left/center/right for operation that require "long-time/near-time/real-time" interaction between HPC and quantum computers

Workflow for Quantum Simulations of Materials

- Practical-wise, one can use the classical resources to do the calculations as usual and focus the quantum part on the important interactions that can benefit from the quantum algorithms and feed the results back to the classical side

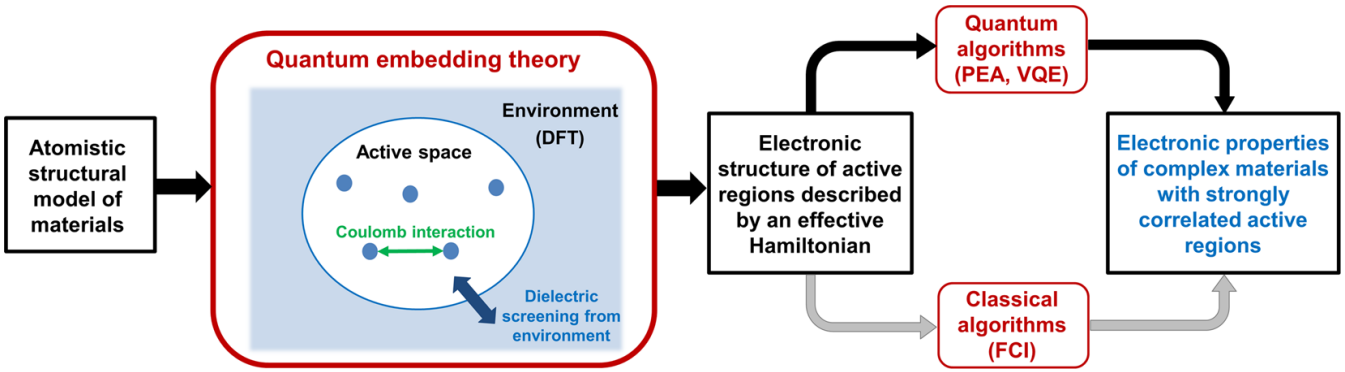


Figure: General strategy for quantum simulations of materials using quantum embedding [9]

quantum embedding and active space

- The figure above shows how to combine the classical and quantum calculations using the quantum embedding method
- Large simulations utilises active space approximation to reduce the resources required to model the system's electronic structure on a quantum workflow

Quantum Embedding Method for the Simulation of Systems on Quantum Computers I

The energy of the active subsystem A embedded into the environment subsystem B by means of an SCF-in-DFT calculation is given by [10]:

$$E_{\text{SCF-in-DFT}}[\gamma^A_{\text{emb}}; \gamma^A, \gamma^B] = E_{\text{SCF}}[\gamma^A_{\text{emb}}] + E_{\text{DFT}}[\gamma^A + \gamma^B] - E_{\text{DFT}}[\gamma^A] + \text{tr}[(\gamma^A_{\text{emb}} - \gamma^A)\nu_{\text{emb}}[\gamma^A,] + \alpha \text{tr}[\gamma^A_{\text{emb}} \mathbf{P}^B] \quad (1)$$

Brief Description

This relies on the projection-based wave function-in-DFT (WF-in-DFT) embedding method, where the total KS density matrix, γ, of the molecular system obtained from KS-DFT is partitioned into an active and environment subsystem, γ_A and γ_B

Key Components

- E_{SCF} is the energy of the embedded subsystem A at SCF level
- $\mathbf{P}^B = S\gamma^B S$ is a projector for orbital orthogonality
- α is a scaling parameter
- ν_{emb} includes all two-electron interactions

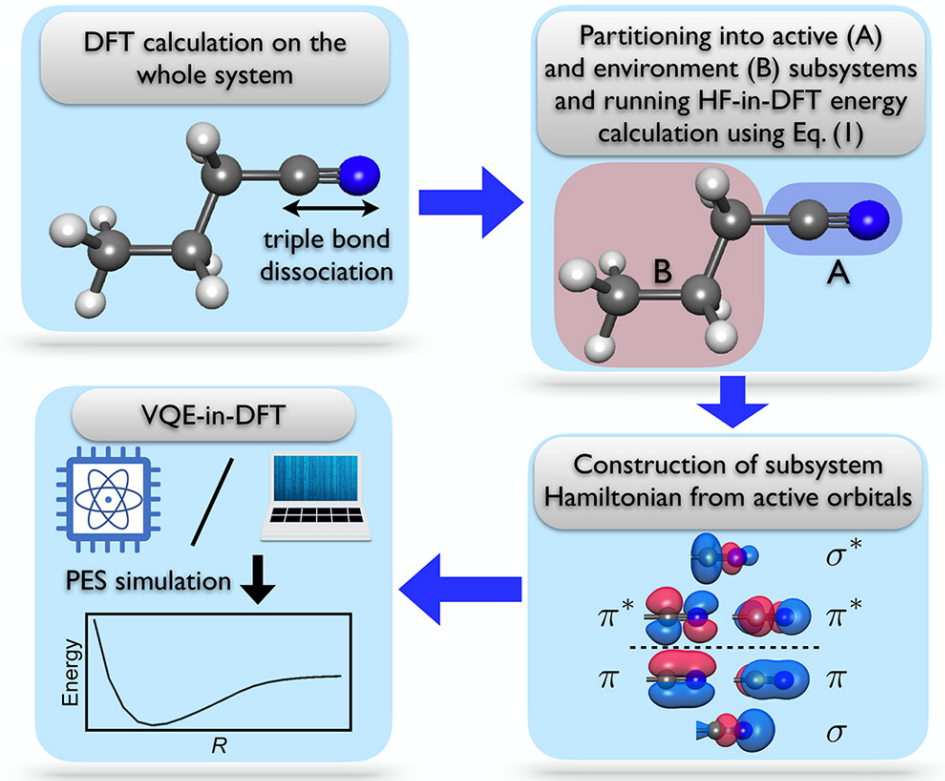


Figure: Quantum embedding in action [10]

Quantum Embedding Method for the Simulation of Systems on Quantum Computers II

ADAPT-VQE at a glance

Adaptive Derivative-Assembled Pseudo-Trotter VQE (ADAPT-VQE) is a solver technique that builds an ansatz iteratively from a predefined operator pool, which can more efficiently converge to predict the ground state energy [11]

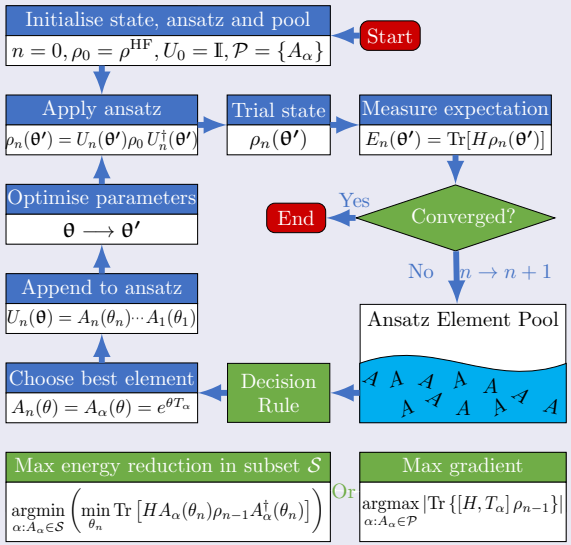


Figure: From [12]:ADAPT-VQE procedure

Then different pool of ansatz offered by UCCSD can be chosen then initial Hartree-Fock state can be formed then build the Hamiltonian

A practical usecase on Quantum Computing for Materials?

- A practical example of how periodic systems can be important to study is studying corrosion inhibition through examining the adhesion power of few inhibitor molecules on top of metal alloys surfaces that form the body of a car or aeroplane. This can enhance the lifespan and efficiency of the car or aeroplane body, there is research around shifting towards eco-friendly alternatives like smart coatings and organic inhibitors due to environmental concerns, for more information please see [1]
- Such inhibitor molecules can benefit from accurate quantum calculations to study their interactions with the metal surface. Then those calculations can correct for energies computed by the Density Functional Theory (DFT)

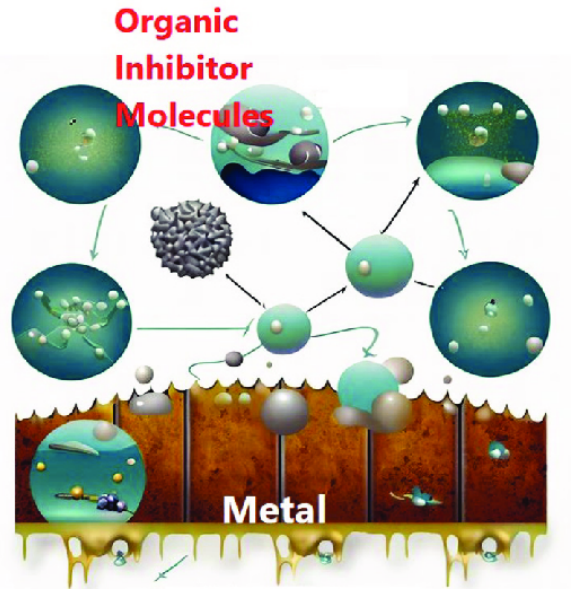


Figure: From [13]: Corrosion inhibition mechanisms of organic inhibitors

How to perform the quantum-centric supercomputing

- Build the DFT simulation workflow
- Identify the part of important interactions to study with the quantum algorithm (e.g. surface-adsorbate interactions where we choose orbitals around the Fermi level)
- Build Hybrid quantum-classical computational framework where calculations from DFT (implemented in CP2K) can talk to quantum algorithm calculation (adaptVQE implemented in Qiskit)

Integration between CP2K and Qiskit

- Socket-based communication between CP2K and Qiskit
- FCIDUMP format for integral transfer

most important article here!

This figure is from [14]

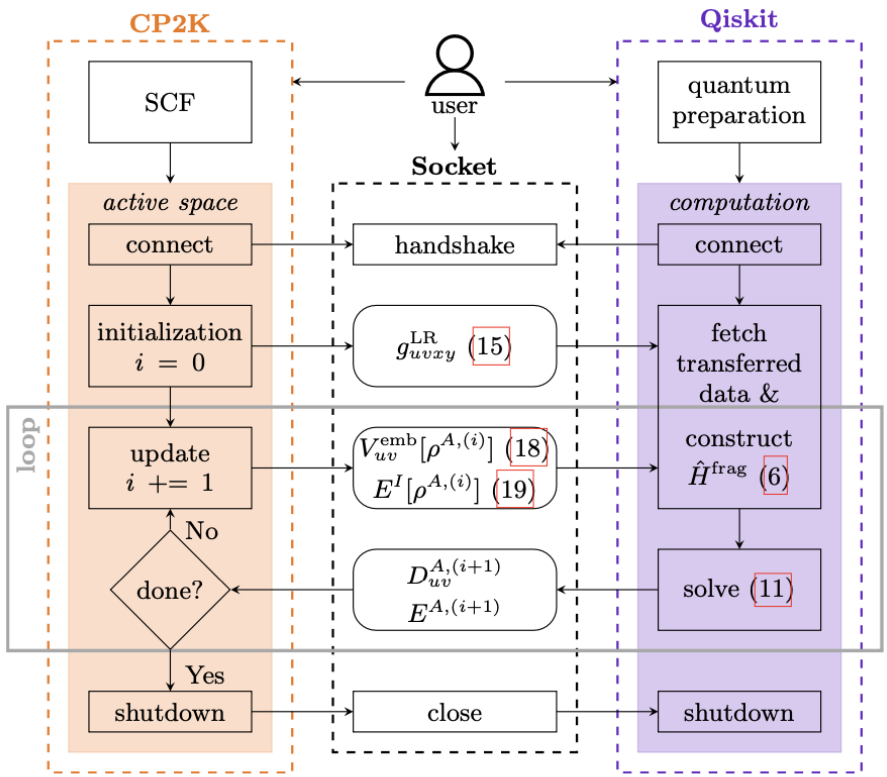


FIG. 2. Workflow diagram depicting the interaction of CP2K and Qiskit Nature. The user configures the two classical processes and the socket for the IPC. Each process then follows the computational steps (rectangular boxes) outlined inside of their respective frames. The data that gets computed and transferred is indicated by the rounded boxes. Numbers in parentheses refer to the respective equations in this manuscript. The self-consistent embedding requires a loop which is highlighted by the gray box. This loop is terminated based on the decision (diamond shape) taken by the CP2K process.

Case Study: Calculational System Simplification

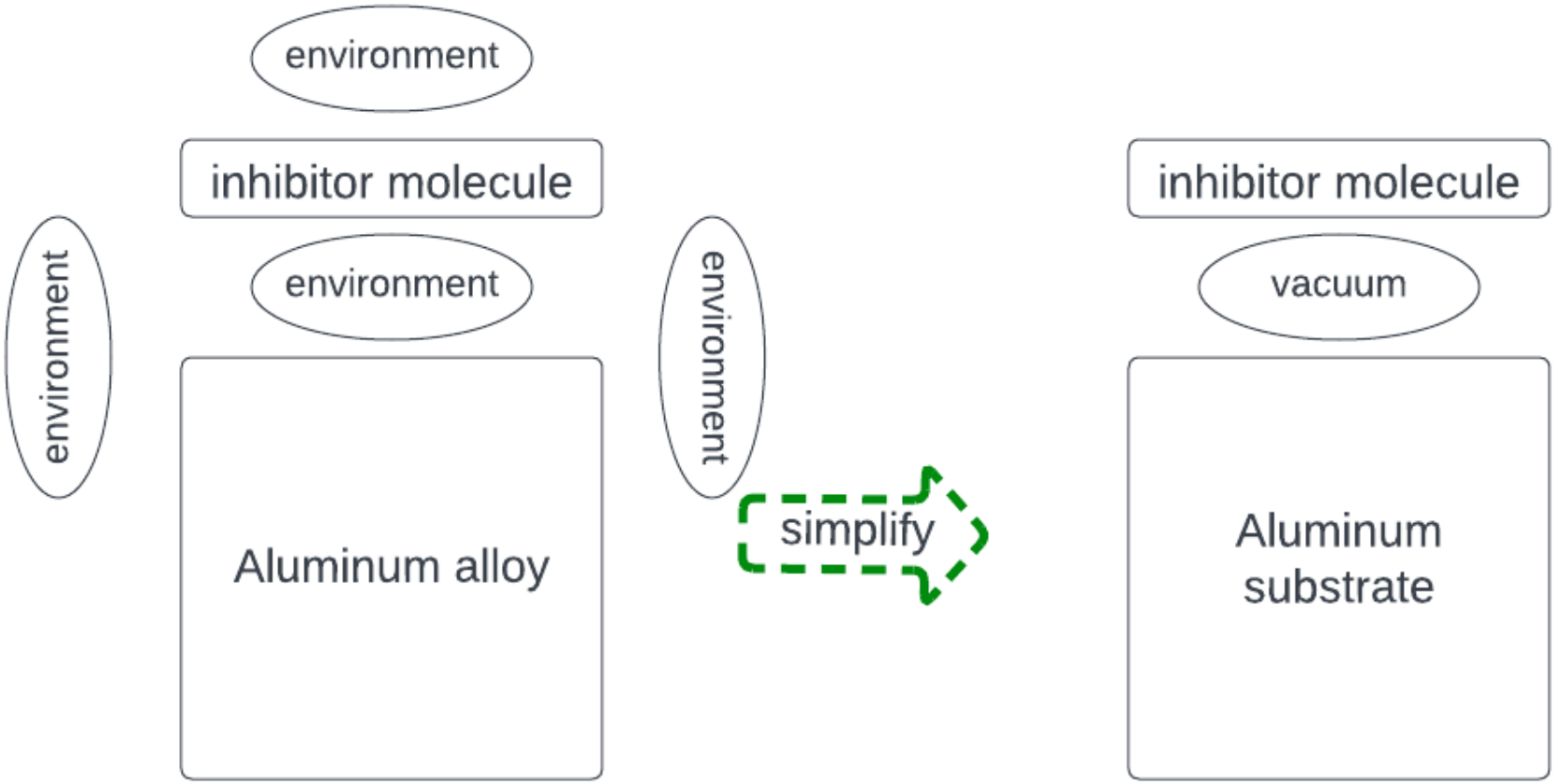
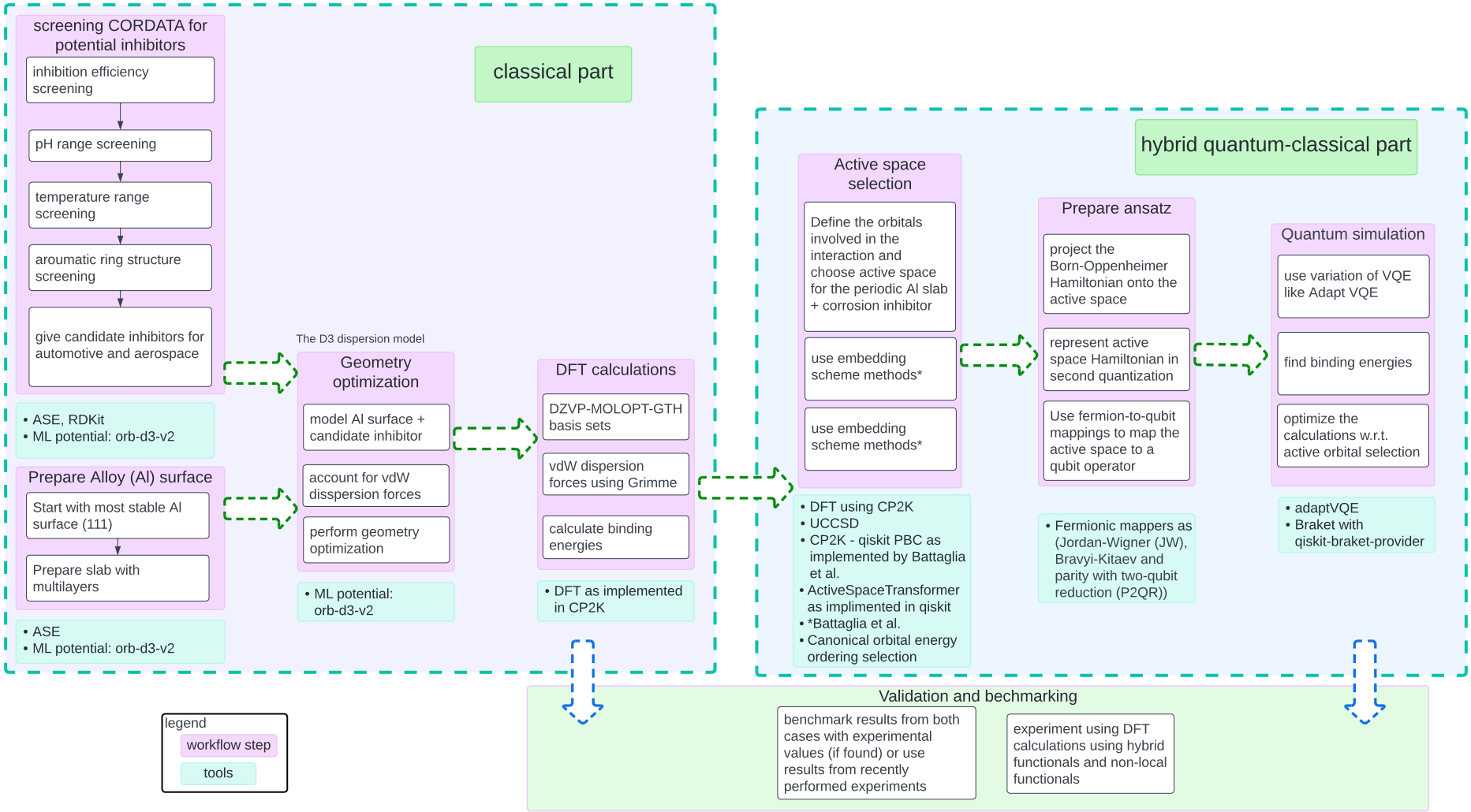


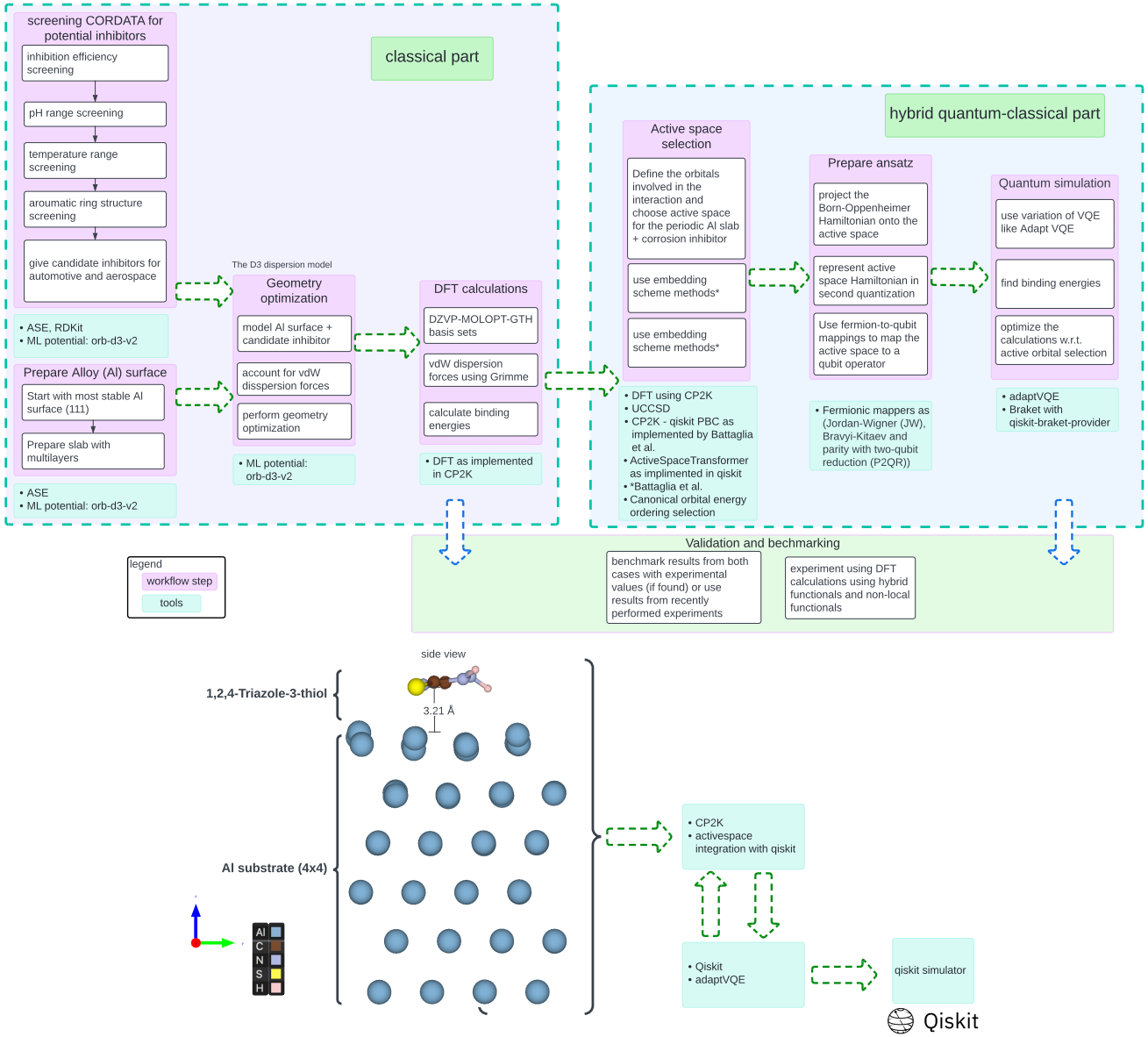
Figure: So, we simplify our system to make the calculations easy to compute and focus on the two parts which are the DFT and the adaptVQE calculations

Case Study: Workflow Overview



From: [1]

Case Study: Workflow with Example System



From: [1]

Case Study: System Setup and Classical Computational Details

Establish a baseline to benchmark to:

Using the DFT code CP2K to solve the problem classically can be a good way to compare with the hybrid calculation that can use both CP2K and Qiskit later

- System: Al(111) surface with triazole inhibitor and vacuum gap of 25 Å in the z-direction. The supercell is 4×4 size to avoid interactions between repeated cells
- PBE functional with D3 dispersion correction
- Periodic DFT calculations using CP2K
- DZVP-MOLOPT-GTH basis sets
- Binding energy calculation:

$$E_{\text{binding}} = E_{\text{supercell}} - (E_{\text{substrate}} + E_{\text{inhibitor}})$$

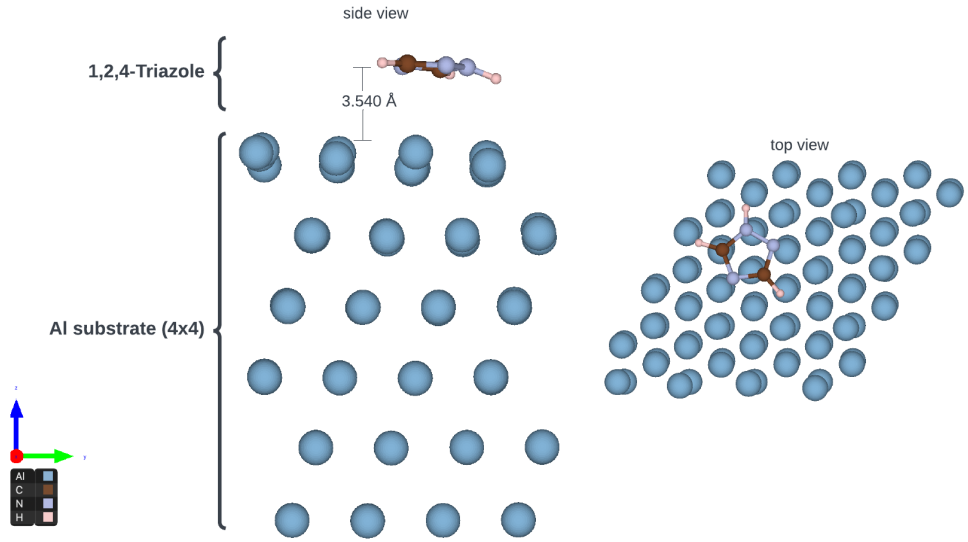


Figure: From [1]

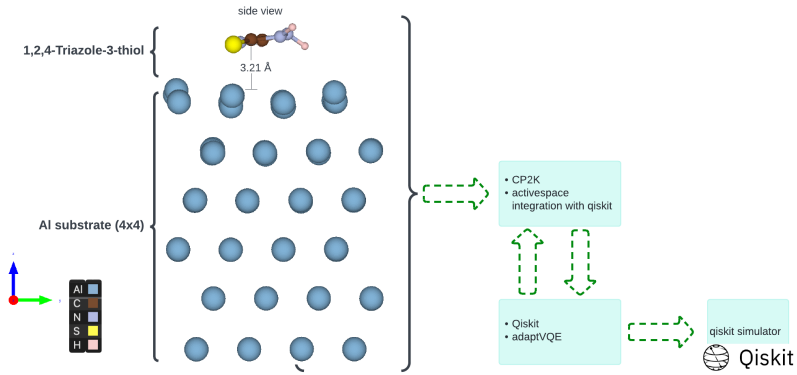
Case Study: Complete Computational Parameters Summary

Method/Component	Details
Classical Calculations	
Geometry Optimization	ASE with orb-d3-v2 model
Dispersion Corrections	Grimme's D3 (integrated into neural network potential)
Surface Model	Al(111) 4×4 supercell
DFT Calculations	
Functional	PBE with GGA implementation
Basis Set	DZVP-MOLOPT-GTH (double-zeta valence polarized)
Method	GPW, plane-wave cutoff: 500 Ry, relative cutoff: 60 Ry
van der Waals	DFT-D3 with PBE reference functional
Vacuum Gap	25 Å (z-direction)
SCF Convergence	1.0E-6 Ha, Broyden mixing ($\alpha = 0.1$, $\beta = 1.5$)
Active Space Parameters	
Configuration	2e, 5o (2 active electrons in 5 orbitals)
Selection Method	ActiveSpaceTransformer (Qiskit implementation), canonical orbital energy ordering selection
Quantum Calculations	
Primary Algorithm	ADAPT-VQE from Qiskit, StatefulAdaptVQE from qiskit-nature-cp2k
Qubit Mapping	Parity with two-qubit reduction
Convergence Criteria	Energy threshold: 1e-6 Hartree, gradient norm: 1e-4
Classical Optimizer	SPSA (learning rate: 0.005, perturbation size: 0.05, max iterations: 1000)
Quantum Hardware & Simulation	
Simulators	Qiskit local and Aer

Case Study: Results - Classical part

Inhibitor	MW (g/mol)	Temp (K)	pH Range	Eff. (%)
1,2,4-Triazole	69.07	298	8-10	90
1,2,4-T-3-thiol	101.13	298	4-10	70-90
Benzotriazole	119.12	298	7-10	90-98
2-MBI	150.2	298	4-10	90
THC	227.24	303	7	91-95
T-methionine	502.70	298	7	95-99

- Shortlisted few inhibitor molecules
- Chosen two variations of Triazole inhibitors
- Geometry-optimized structures consistent with experimental observations



Case Study: Results - Hybrid part

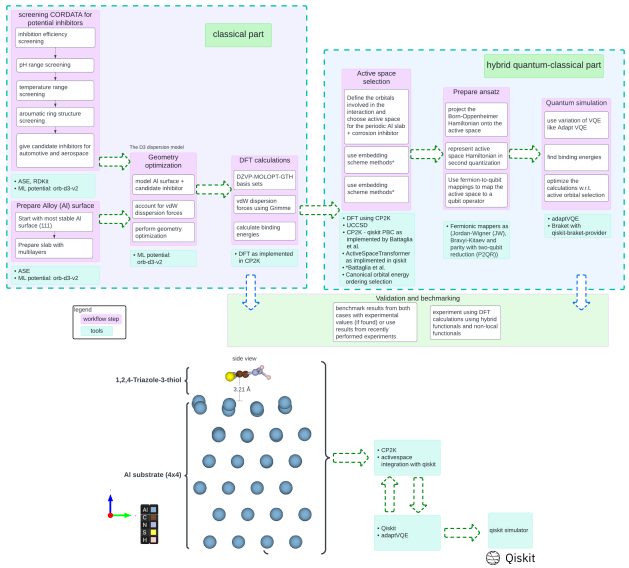
Method	Inhibitor	Binding Energy (eV)	Distance (Å)
Classical DFT	1,2,4-Triazole	-0.385512	3.54
AdaptVQE	1,2,4-Triazole	-0.385508	3.54
vanilla VQE	1,2,4-Triazole	-2.325986	3.54
Classical DFT	1,2,4-Triazole-3-thiol	-1.279063	3.21
AdaptVQE	1,2,4-Triazole-3-thiol	-1.279064	3.21

- Agreement between classical DFT and AdaptVQE methods
- Stronger binding for 1,2,4-Triazole-3-thiol (-1.279 eV) vs 1,2,4-Triazole (-0.386 eV)
- Shorter binding distance for thiol derivative (3.21Å vs 3.54Å)
- Enhanced surface interaction of thiol derivative supports findings on sulfur's role in surface passivation

Conclusion

- Demostrated hybrid quantum-classical approach for modeling corrosion inhibitors
- Computed a physical property of binding energies with classical and hybrid methods
- AdaptVQE showed better performance over vanilla VQE
- Potential for expanding active space to capture complex interactions
- Priliminary Manuscript is here [1] and Implementation available on GitHub is here [2]. Comments and pull requests are welcomed!

Method	Inh.	B.E. (eV)	Dist. (Å)
Class.	1,2,4-T	-0.39	3.54
AdaptVQE	1,2,4-T	-0.39	3.54
VQE	1,2,4-T	-2.33	3.54
Class.	T-3-t	-1.28	3.21
AdaptVQE	T-3-t	-1.28	3.21



Ongoing Work & Future Directions

- This work is an effort towards building guidance workflow for quantum-classical hybrid materials simulation
- Current results demonstrate proof-of-concept on periodic systems, but there is a huge room for improvement
- The results are yet to be improved and include more influence coming from the embedding methods
- Ongoing developments:
  - Scaling to larger active space
  - Integration of qiskit-nature-cp2k with newest qiskit version 1.x
- Future work will focus on:
  - Benchmarking against pure classical methods (FCI)
  - Implementation of zero-noise extrapolation techniques
  - Optimization of quantum circuit ansätze for hardware constraints
  - Adapt the workflow to the recent work here [15]
  - Extension to broader materials science applications and use cases like metal-oxide frameworks (carbon capture) and catalysis (hydrogen storage)

Useful literature and resources

Learning resources

- For some recent VQE implementations: [16]
- A good review article on the topic of quantum-centric supercomputing: [6]
- For more insights, see the tutorial video lecture [17]
- NVidia blog post on ADAPT-VQE [11]

Qiskit Nature and CP2K resources

- Qiskit-nature-CP2K integrationhttps://github.com/Qiskit/qiskit-nature-cp2k
- Qiskit-nature https://github.com/qiskit-community/qiskit-nature

Quantum Embedding tools

- A Python package for Bootstrap Embedding (BE) method for quantum embedding: [18], the related publication is [15]
- For more on quantum embedding: [19]

References I

[1] K. Elgammal and M. Maußner, "A quantum computing approach to simulating corrosion inhibition," (2024), arXiv:2412.00951 [cond-mat.mtrl-sci], arXiv:2412.00951 [cond-mat.mtrl-sci] .

[2] "Inhibitq project," https://github.com/MarcMaussner/2024_inhibitQ (2024), accessed: 2024-12-03.

[3] NordiQuEst, "Variational quantum eigensolver (vqe)," (2024), nordiQuEst Application Library.

[4] Y. Shikano, H. C. Watanabe, K. M. Nakanishi, and Y.-y. Ohnishi, The European Physical Journal Special Topics **230**, 1037 (2021).

[5] P. S. Goh, K. C. Wong, and A. F. Ismail, Membranes **10**, 297 (2020).

[6] Y. Alexeev and Others, Future Generation Computer Systems **160**, 666 (2024).

[7] J. Li, G. Michelogiannakis, B. Cook, D. Cooray, and Y. Chen, in *High Performance Computing*, edited by A. Bhatele, J. Hammond, M. Baboulin, and C. Kruse (Springer Nature Switzerland, Cham, 2023) pp. 297–316.

[8] I. Jackson, "High performance computing for DFT and MD simulations," https://indico.ph.ed.ac.uk/event/2/contributions/211/attachments/109/119/Jackson_HPC_DFT_nFAME_FeCr_workshop_5June2013.pdf (2013), presentation at nFAME FeCr Workshop, June 5, 2013.

[9] H. Ma, M. Govoni, and G. Galli, npj Computational Materials **6**, 85 (2020).

References II

[10] M. Rossmannek, F. Pavošević, A. Rubio, and I. Tavernelli, The Journal of Physical Chemistry Letters **14**, 3491 (2023).

[11] M. Perricone, "Introducing NVIDIA CUDA-QX libraries for accelerated quantum supercomputing," https://developer.nvidia.com/blog/introducing-nvidia-cuda-qx-libraries-for-accelerated-quantum-supercomputing/ (2024), accessed: 2024-12-04.

[12] K. Dalton, C. K. Long, Y. S. Yordanov, C. G. Smith, C. H. W. Barnes, N. Mertig, and D. R. M. Arvidsson-Shukur, npj Quantum Information **10** (2024), 10.1038/s41534-024-00808-x.

[13] A. Al-Amiery, W. Norsani, w. s. wan nik, M. Sabri, M. Ghazali, E. Yousif, W. al azzawi, M. Fakhratul, R. Zulkifli, W. Daoudi, V. Izionworu, and D. Dhande, Journal of Sustainability Science and Management **19**, 146 (2024).

[14] S. Battaglia, M. Rossmannek, V. V. Rybkin, I. Tavernelli, and J. Hütter, "A general framework for active space embedding methods: applications in quantum computing," (2024), arXiv:2404.18737 [physics.chem-ph] .

[15] L. P. Weisburn, M. Cho, M. Bensberg, O. R. Meitei, M. Reiher, and T. V. Voorhis, "Multiscale embedding for quantum computing," (2024), arXiv:2409.06813 [physics.chem-ph] .

[16] D. Gunlycke, C. S. Hellberg, and J. P. T. Stenger, Phys. Rev. Res. **6**, 013238 (2024).

References III

[17] “”materials science simulations on quantum computers,” andré schleife,” https://www.youtube.com/live/lvdL-gsn3Fg?si=jEgBYEzljM7AYZDP (2024), accessed: 2024-12-03.

[18] “Bootstrap embedding (be) method for quantum embedding,” https://github.com/oimeitei/quemb?tab=readme-ov-file (2024), accessed: 2024-12-03.

[19] M. Nusspickel and G. H. Booth, Phys. Rev. X **12**, 011046 (2022).