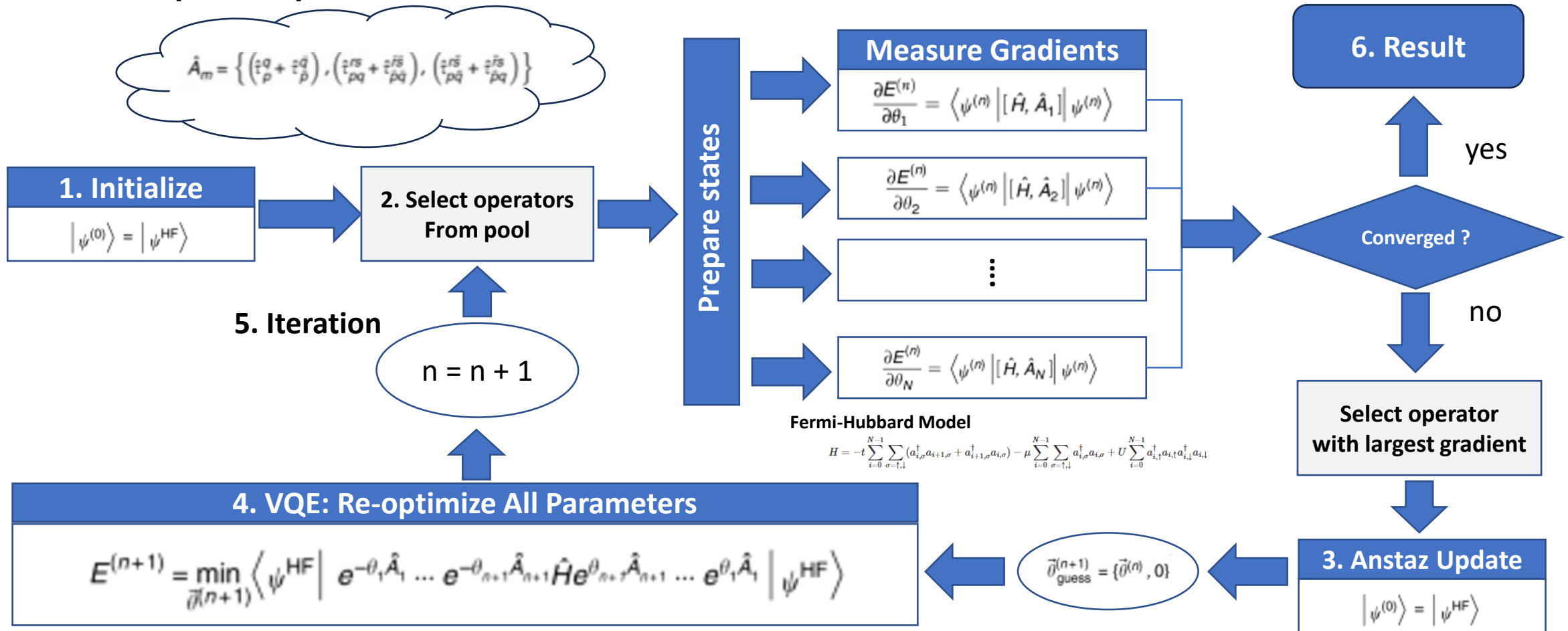Process Flow of Adapt-VQE For Fermi-Hubbard Model

Team: QuanInt

Quantum Algorithm Grand Challenege 23'

Step 1: Initialization:

➢ The algorithm starts by initializing the Hamiltonian and the number of qubits.

➢ The fermion pool is initialized based on the number of orbitals (which is half the number of qubits). The fermion pool consists of single and double excitation operators.

➢ The fermion operators are then transformed into qubit operators using the Jordan-Wigner transformation. This forms the qubit pool.

➢ The initial state is set to a computational basis state, and the initial ansatz circuit is constructed from the Hartree-Fock state.

Step 2: Operator Selection:

➢ For each operator in the qubit pool, the algorithm calculates the gradient of the operator with respect to the current state.
➢ The operator with the largest absolute gradient is selected and added to the ansatz.

Step 3: Ansatz Update:

➢ The ansatz circuit is updated by appending the selected operator.
➢ The new parameter associated with the selected operator is initialized to zero.

Step 4: Parameter Optimization:

➢ The parameters of the ansatz circuit are optimized using the Adam optimizer.

➢ The cost function is the expectation value of the Hamiltonian, and the gradient is calculated using the parameter-shift rule.

➢ The optimization process updates the parameters to minimize the cost function.

Step 5: Iteration:

➢ Steps 2-4 are repeated until a stopping condition is met. In this case, the stopping condition is when the total quantum circuit time exceeds a certain limit.

Result:

➢ The final result is the minimum expectation value of the Hamiltonian, which represents the estimated ground state energy of the system.

Initialise state, ansatz and pool
$n = 0, \rho_0 = \rho^{\text{HF}}, U_0 = \mathbb{I}, \mathcal{P} = \{A_\alpha\}$

Start

Apply ansatz
$\rho_n(\vec{\theta'}) = U_n(\vec{\theta'})\rho_0 U_n^\dagger(\vec{\theta'})$

Trial state
$\rho_n(\vec{\theta'})$

Measure expectation
$E_n(\vec{\theta'}) = \text{Tr}[H\rho_n(\vec{\theta'})]$

Optimise parameters
$\vec{\theta} \longrightarrow \vec{\theta'}$

Converged?

Yes

No   N→N+1

Append to ansatz
$U_n(\vec{\theta}) = A_n(\theta_n)\cdots A_1(\theta_1)$

Ansatz Element Pool

Choose best element
$A_n(\theta) = A_\alpha(\theta) = e^{\theta T_\alpha}$

Decision Rule

Max energy reduction in subset $\mathcal{S}$
$\underset{\alpha:A_\alpha\in\mathcal{S}}{\text{argmin}} \left(\underset{\theta_n}{\min} \text{Tr}\left[HA_\alpha(\theta_n)\rho_{n-1}A_\alpha^\dagger(\theta_n)\right]\right)$

Or

Max gradient
$\underset{\alpha:A_\alpha\in\mathcal{P}}{\text{argmax}} |\text{Tr}\{[H, T_\alpha]\rho_{n-1}\}|$

END

```
Running algorithm(1)..
iteration 1
-7.817853896126992
iteration 2
-7.42228258172591
iteration 3
-7.456232891619905
iteration 4
-7.939863832387813
iteration 5
-7.643200256038577
iteration 6
-7.959757180457868

############## Result ##############
Resulting energy = -7.959757180457868
Circuit time = 1004.1084
###################################
```