Quantum-Inspired Simulation of 1D Burger's Equation

Objective

The main objective of this program code is to solve and analyze the 1D viscous Burgers' equation using a classical finite-difference solver and Hydrodynamic Schrodinger Equation (HSE) framework inspired quantum algorithm. Then the quantum circuit performance was evaluated with and without noise modeling.

Algorithm

Classical solver: `burgers_fd(u0, nu, dx, dt, nt)`

This function solves burgers' equation using explicit finite differences.

nt is the number of steps

Initial condition setup: `riemann_initial(nx)`

This function creates a Riemann problem initial condition where $u_{left} = 1.0$ and $u_{right} = 0.0$

Quantum-Inspired Evolution:

`kinetic_operator(qc, qubits, theta)`

This function applies applies Rx gates which mimics a kinetic energy operator

`potential_operator(qc, qubits, u_field)`

This function applies Rz gates which mimics potential from Burgers' field u

`Trotter_step(qc, qubits, dt, u_field)`

This function combines kinetic and potential operators (1^{st}-order Trotterization)

Circuit: `build_hse_circuit(u0, n_qubits, dt)`

This function initializes a quantum circuit with Hadamard superposition over all qubits and then a single trotter step using u0 values. Finally, it takes the full measurement of all qubits.

L2 error metric: `compute_l2_error(u1, u2)`

This function computes normalized L2 norm to quantify error between quantum and classical results.

Simulation pipeline: `run_simulation( )`

Grid size: nx=16

Time steps: nt=3

Viscosity: nu=0.01

This function runs classical FD solver to get reference solution `u_classical`. Moreover, build and run quantum circuit names as `qasm_simulator` to obtain bitstring counts. Then it postprocess bitstrings into a normalized output vector. After that it computed L2 error between quantum and classical solutions. Then plot results for visual comparison.