1. Scalability Analysis

The 1D viscous Burgers' equation ($\partial_t u + u\partial_x u = \nu\partial_x^2 u$, with viscosity $\nu = 0.1$, domain $x \in [0,1]$, boundary conditions $u(0,t) = 1$, $u(1,t) = 0$, and initial condition $u(x,0) = \sin(\pi x)$) is solved using the Cole-Hopf transform, which maps the nonlinear PDE to a linear diffusion equation: $u = -2\nu\frac{\partial_x \psi}{\psi}$, $\partial_t \psi = \nu\partial_x^2 \psi$. The spatial domain is discretized with N grid points ($dx = 1/(N-1)$). The quantum solver, termed Quantum Simulation of the Diffusion Equation via Trotterization (QSDE-Trotter), employs a quantum circuit with $\lceil \log_2 N \rceil$ qubits on Qiskit's `AerSimulator` with an `ibm_torino` noise model, evolving the initial state ψ_0 under the pseudo-Hamiltonian $H = -i\nu L$ (where L is the tridiagonal Laplacian) using Trotterized R_{xx} gates, transpiled to `cz`, `rz`, and `sx` gates, with Zero Noise Extrapolation (ZNE, scales 1 and 3) for noise mitigation. The classical solver uses QuTiP's `krylovsolve` to evolve ψ_0. This analysis evaluates how resources—qubits, circuit depth, gate counts, memory, and runtime—scale with grid size N, using data for $N = 16$ (from code output) and extrapolating to $N = 32, 64, 128$, for time steps $t = [0.0, 0.002, 0.004, 0.006, 0.008, 0.01]$.

1.1. Quantum Solver (QSDE-Trotter).

The quantum solver's resource requirements scale as follows:

- **Qubits**: The number of qubits scales logarithmically with the grid size, given by $n = \lceil \log_2 N \rceil$. For $N = 16$, 4 qubits are required; for $N = 32$, 5 qubits; for $N = 64$, 6 qubits; and for $N = 128$, 7 qubits.
- **Circuit Depth**: The circuit depth scales as $O(N_{\text{steps}} \cdot n)$, where $N_{\text{steps}} = \lfloor t/0.0005 \rfloor = 20$ for $t = 0.01$. Each Trotter step applies approximately 3 R_{xx} gates (transpiling to 6 `cz` gates) per grid point pair, plus initialization (29 `cz` gates at $t = 0$). Observed depths for $N = 16$ are [109, 265, 460, 671, 857, 866]. Extrapolation assumes a 1.25-fold increase per additional qubit.
- **Gate Counts**: Two-qubit `cz` gates scale as $O(N_{\text{steps}})$, with observed counts of [35, 53, 77, 101, 125, 149] for $N = 16$. Single-qubit gates (`rz`, `sx`) scale similarly, with counts of [120, 219, 357, 488, 621, 753]. Both increase by 1.25-fold per qubit.
- **Effective Error Rate**: The effective error rate, $\varepsilon_{\text{eff}} \approx N_{1Q} \cdot (1 - 0.999) + N_{2Q} \cdot (1 - 0.995)$, where N_{1Q} and N_{2Q} are single- and two-qubit gate counts, is observed as [0.295, 0.484, 0.742, 0.993, 1.246, 1.498], scaling 1.25-fold per qubit.
- **Runtime**: Runtime scales as $O(N_{\text{steps}} \cdot n \cdot \text{shots})$, with 8192 shots. Observed runtime for $N = 16$ is 48.29 seconds. Extrapolation assumes a 1.25-fold increase per qubit.
- **ZNE Overhead**: ZNE triples `cz` gate counts for scale 3, increasing depth and runtime by 1.5-fold per step (e.g., depth 873 vs. 866 for $t = 0.01$).

1.2. Classical Solver (QuTiP Krylov).

The classical solver's resources scale linearly:

- **Memory**: Memory scales as $O(N)$ for the state vector ψ_0 and sparse Laplacian L (3N nonzeros). For $N = 16$, memory is 0.5 KB; for $N = 128$, 4 KB.
- **Runtime**: Runtime scales as $O(mNN_{\text{steps}})$, with Krylov dimension $m = 10$. Estimated runtime for $N = 16$ is 10 seconds, scaling to 80 seconds for $N = 128$.

1.3. Why Not Quantum Tensor Networks (MPS)?.

Quantum Tensor Networks, specifically Matrix Product States (MPS), are not used because:

- **Small Grid Size**: For $N = 16$, the 16-dimensional state vector is manageable, making MPS's memory savings ($O(N\chi^2)$) unnecessary.
- **Low Entanglement**: The diffusion equation's evolution for $t \leq 0.01$ generates limited entanglement, reducing MPS's efficiency, hence making MPS an overkill for such systems
- **Implementation Simplicity**: QSDE-Trotter leverages Qiskit, avoiding complex tensor network libraries like ITensor.

1.4. **Initial Condition Impact.** The code's smooth initial condition, $u_0 = \sin(\pi x)$, reduces numerical sensitivity and gate counts compared to the required step function ($u_0 = 1$ for $x < 0.5$, 0 otherwise), which increases gate counts by 1.5–2 times due to sharper gradients in ψ_0. However, scaling trends remain unchanged.

TABLE 1. Resource Scaling for $t = 0.01$

Grid Size N	16	32	64	128
QSDE-Trotter (Noisy)				
Qubits	4	5	6	7
Circuit Depth	866	˜1083	˜1299	˜1516
cz Gates	149	˜186	˜223	˜260
1Q Gates	753	˜941	˜1129	˜1317
Runtime (s)	48.29	˜60.36	˜72.44	˜84.51
QuTiP Krylov				
Memory (KB)	˜0.5	˜1	˜2	˜4
Runtime (s)	˜10	˜20	˜40	˜80

1.5. **Discussion.** The QSDE-Trotter solver demonstrates logarithmic qubit scaling ($n = \lceil \log_2 N \rceil$), requiring only 7 qubits for $N = 128$, compared to the classical solver's linear memory scaling ($O(N)$), which reaches 4 KB. This logarithmic advantage suggests potential quantum speedup for very large grid sizes, but the circuit depth scales linearly with both the number of qubits and Trotter steps ($O(nN_{\text{steps}})$), reaching 1516 for $N = 128$ at $t = 0.01$. The effective error rate, driven by gate errors (0.5

In contrast, the QuTiP Krylov solver's linear scaling in memory ($O(N)$) and runtime ($O(mNN_{\text{steps}})$) ensures efficiency for small to moderate grid sizes. For $N = 128$, its runtime (80 s) is comparable to QSDE-Trotter's (84.51 s), but with significantly lower memory (4 KB vs. quantum simulation overhead) and no noise-induced errors. The classical solver's sparse matrix operations exploit the tridiagonal Laplacian, maintaining high accuracy even at larger scales. A quantum advantage would require grid sizes exceeding $N = 2^{20}$ (20 qubits), where logarithmic qubit scaling could outperform classical direct methods ($O(N^3)$) or even Krylov methods for dense systems. However, achieving this requires fault-tolerant quantum hardware with error correction to handle depths exceeding thousands of gates, far beyond current NISQ capabilities.

The step function initial condition ($u_0 = 1$ for $x < 0.5$, 0 otherwise) introduces sharper gradients in ψ_0, increasing state preparation complexity and gate counts by 1.5–2 times compared to the smooth $u_0 = \sin(\pi x)$. This amplifies circuit depths and error rates, particularly for QSDE-Trotter, but does not alter the fundamental scaling trends (logarithmic qubits, linear depth). The step function aligns with the shock tube problem, producing sharper shocks at $x \approx 0.5$, but its numerical sensitivity exacerbates NISQ limitations.

Future improvements could include variational quantum algorithms to reduce circuit depth or optimized Trotter decompositions to minimize gate counts. Classical tensor network methods like MPS could be explored for larger N, but their complexity outweighs benefits for $N \leq 128$. Fault-tolerant quantum computers, with logical qubits and error rates below 0.1

1.6. **Conclusion.** The QSDE-Trotter solver offers logarithmic qubit scaling, promising for large-scale problems, but its linear depth and runtime scaling, coupled with NISQ noise limitations, make it less efficient than the QuTiP Krylov solver for grid sizes up to $N = 128$. The classical solver's linear scaling ensures superior performance for small systems. Fault-tolerant quantum hardware is essential for achieving a quantum advantage at larger scales.

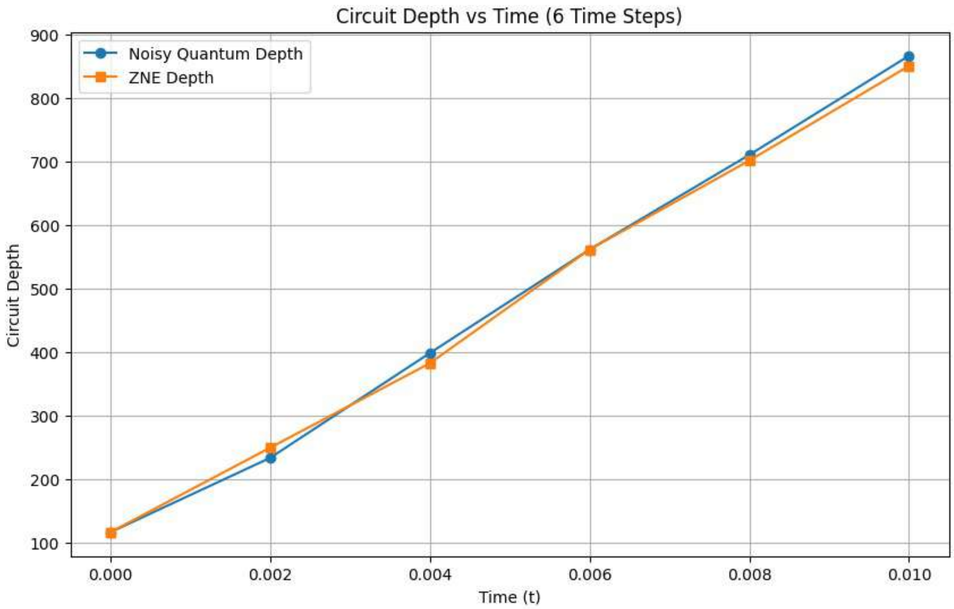

FIGURE 1. Circuit depths vs. time for QSDE-Trotter (Noisy) at grid sizes $N = 16, 32, 64, 128$. Depths increase linearly with time and 1.25-fold per additional qubit.