Quantum Circuit Transformation For Neutral atom computation

Dingchao Gao

ISCAS

24 de agosto de 2025

Main reference

- Computational Capabilities and Compiler Development for Neutral Atom Quantum Processors: Connecting Tool Developers and Hardware Experts
- DasAtom: A Divide-and-Shuttle Atom Approach to Quantum Circuit Transformation,Yunqi Huang and Dingchao Gao and Shenggang Ying and Sanjiang Li

contents

Recent Developments

- **Microsoft and Atom Computing Collaboration**
  microsoft.com/en-us/research/blog/microsoft-and-atom-computing-collaborate-on-quantum-supercomputer
- **Modular Quantum System-on-Chip Development**
  nature.com/articles/s41586-024-03876-9
- **Fault-Tolerant Quantum Computation with High-Rate qLDPC Codes**
  arxiv.org/abs/2408.12345

Neutral Atom Quantum Computing Devices

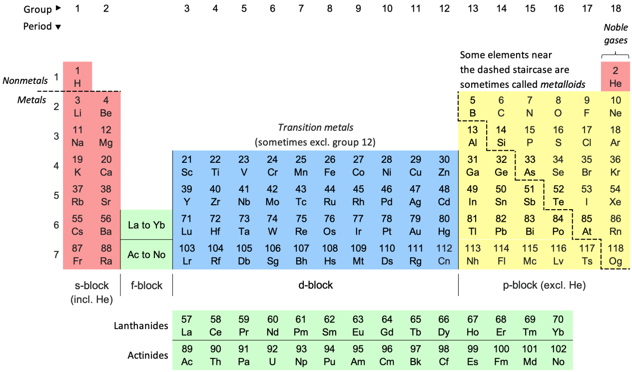
- **Qubits:** Individual neutral atoms (e.g., rubidium, cesium)
- **Trapping Methods:** Optical tweezers, optical lattices
- **Encoding:** Internal atomic states (hyperfine ground states)
- **Interactions:** Mediated via Rydberg state excitation

Key Features

- **Scalability:** Potential for large qubit arrays
- **Long Coherence Times:** Use of clock states reduces sensitivity to magnetic fields
- **Controlled Interactions:** Tunable via Rydberg excitations

Common Atoms Used

- **Alkali Atoms** Rubidium (Rb), Cesium (Cs)
- **Alkaline-Earth-Like Atoms** Strontium (Sr),Ytterbium (Yb)



Workflow

1. Metal
2. Atomic beam
3. Cooling
4. Optical lattice
5. Rearrange

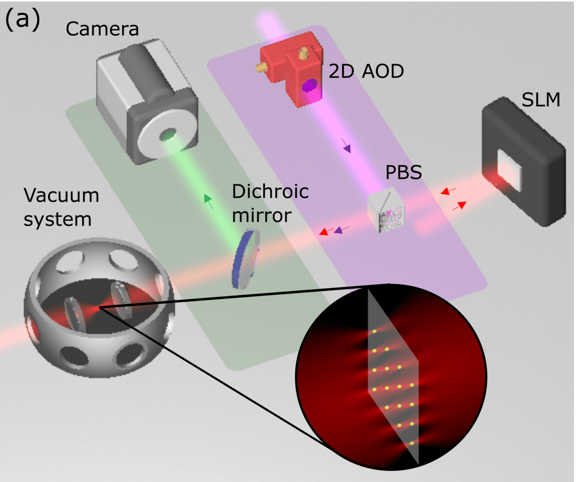


Figura: Overview of the main hardware components constituting a quantum processor

Optical Traps and Tweezers

- **Atom Trapping Methods**
  - Optical dipole traps
  - Optical tweezers created by lasers
- **Array Configurations**
  - One-, two-, or three-dimensional setups
- **Dynamic Control**
  - Reloading and rearrangement of atoms
  - Enhances scalability and qubit adjustment

- **Single-Qubit Operations**
  - Laser pulses drive Rabi oscillations
  - Controlled via Rabi frequency and detuning
- Implemented by driving Rabi oscillations between qubit states $|0\rangle$ and $|1\rangle$.
- Hamiltonian:

$$\frac{H_1(t)}{\hbar} = \frac{\Omega(t)}{2}|0\rangle\langle 1| + \frac{\Omega^*(t)}{2}|1\rangle\langle 0| - \Delta(t)|1\rangle\langle 1|$$

  where $\Omega(t)$ and $\Delta(t)$ are Rabi frequency and detuning.
- Single- or two-photon transitions used, with options for individual or global laser addressing.

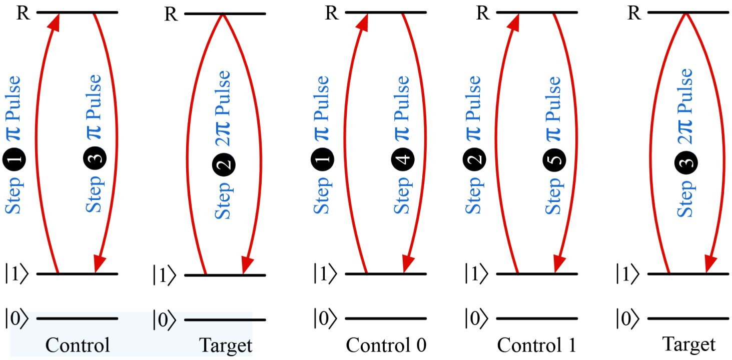
Two-Qubit Gates via Rydberg Blockade:

- Rydberg blockade prevents simultaneous excitation of two nearby atoms to Rydberg states (one or more electrons that have a very high principal quantum number, n).
- Blockade radius r_b defines the interaction range:

$$r_b \simeq \left(\frac{C_6}{\hbar \Omega_r} \right)^{1/6}$$

- Enables two-qubit gates for atoms beyond nearest neighbors, increasing connectivity.

Multi-qubit gates Gates with Rydberg Atoms



(a) CZ Rydberg Pulse **(b) CCZ Rydberg Pulse**

Shutting Atoms

- **Shuttling Process**
  - Qubits transferred between static (SLM) and dynamic (AOD) traps.
  - Controlled AOD frequency moves qubits, then releases to static trap.
- **Dynamically Field-programmable Qubit Arrays (DPQA)**
  - Shuttling-only processor design with entangling, measuring, and storage zones.
  - Balances routing overhead and operation optimization.
- **Benefits of Shuttling in NAQC**
  - Dynamically moves qubits without breaking entanglement.
  - Replaces virtual SWAP (3 CX gates) for flexible qubit connections.

Shuttling Process

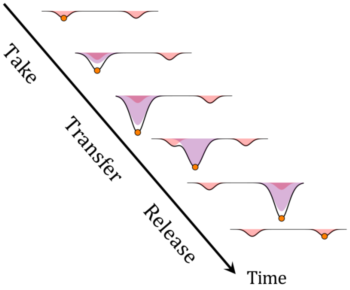


Figura: Moving a single atom from one site to another in the register

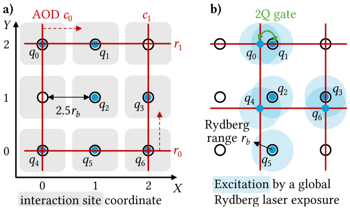


Figura: DPQA

Scalability Potential

- **Flexible Arrangement**
  - Atoms arranged in customizable geometries
- **Long-Range Interactions**
  - Rydberg blockade enables distant entanglement
- **Atom Shuttling**
  - Physical movement enhances connectivity

Challenges in Scaling Up

- **Cross-Talk**
  - Limited precision in laser focus (Stark effect)
  - Unintended interactions during Rydberg transitions
- **Maintaining Coherence**
  - Coherence times may decrease with system size
- **Reliable Atom Control**
  - Preventing atom loss and decoherence during trapping and shuttling

Differences in Noise Sources Compared to Other Platforms

Superconducting Qubits

- Sensitive to electromagnetic noise and material defects
- Decoherence due to charge and flux noise

Trapped Ion Qubits

- Susceptible to electric field fluctuations
- Motional heating and laser noise

Photonic Qubits

- Affected by atom loss and detector inefficiencies
- Immune to thermal noise affecting matter-based qubits

Common Noise Sources in Neutral Atom Quantum Computing

- **Laser Noise:** Intensity fluctuations, frequency/phase noise
- **Atom Loss:** Background gas collisions, photon scattering
- **Decoherence:** Spontaneous emission, blackbody radiation
- **Motional Decoherence:** Thermal motion, trap fluctuations
- **Control Errors:** Beam imperfections, timing jitter
- **Rydberg Decay:** Finite lifetimes, state mixing

Mitigation Strategies

- **Laser Stabilization:** High-quality lasers with active stabilization
- **Vacuum Improvements:** Enhanced vacuum conditions
- **Cooling Techniques:** Raman sideband cooling
- **Error Correction:** Quantum error correction protocols
- **Optimal Control:** Robust control pulse design

contents

Key Compilation Subroutines during Compilation

- **Platform-independent Compilation:**General optimizations, e.g., loop unrolling and gate commutation.
- **Platform-dependent Compilation:**Adapts quantum circuits to specific hardware constraints and capabilities.
- **Hardware-dependent Compilation:**Translates abstract operations into hardware-executable instructions.

primary objectives

- **Synthesis:** Decompose gates into native hardware-compatible operations.
- **Mapping:** Assign logical qubits to physical qubits, introducing SWAP/MOVE operations.
- **Scheduling:** Optimize gate execution timing to maximize parallelism and minimize errors.

attention:

- NAQC-specific capabilities like atom shuttling and Rydberg interactions affect all steps.
- Compilation steps are often optimized collectively for hardware constraints.

SC Compilation Challenges for NAQC

- **Synthesis:** SC only supports one- and two-qubit gates, making multi-qubit gates challenging.
- **Mapping:** SC relies on virtual swaps for connectivity, while NAQC defines coupling using r_{int}.
- **Scheduling:** NAQC adds constraints from the restriction radius, requiring tailored adjustments.

Overview of Related Work on NAQC Compilation

- Neutral atom quantum computing (NAQC) introduces unique challenges and capabilities.
- Related works address these through distinct themes:
  - Long-range interactions
  - Multi-qubit gates
  - Shuttling operations
  - Adaptation of superconducting (SC) compilers

Two Major Implementations:

Baker et al. [33]

- Focus: Mapping & atom loss
- Features parallel SWAP execution
- Uses look-ahead schemes

Q-Tetris [28]

- Greedy heuristic algorithm
- Monte Carlo tree search
- Time-efficient optimization

Geyser [29, 115]

- Novel approach to multi-qubit operations
- Key features:
  - Composes Toffoli gates from two-qubit gates
  - Uses Qiskit for mapping
  - Optimizes laser pulse count
- Evaluation metric: laser pulse count vs. gate count

Shuttling Compilers (1/2)

First Generation:

- **Brandhofer et al. [31]**
  - 1D atom displacements
  - SMT solver optimization
- **Tan et al. [111, 116]**
  - DPQA architecture focus
  - Visual movement animations
  - 2D shuttling capability
- **Schmid et al. [36]**
  - Hybrid compilation scheme
  - SABRE-based heuristic
  - Comprehensive capability support

Future Challenges

- Synthesis improvements needed for:
  - General quantum algorithms
  - Multi-qubit gate integration
- Integration of multiple capabilities
- Hardware-dependent error analysis
- Platform accessibility improvement

Goal: Full exploitation of NAQC platform potential

contents

Subgraph Isomorphism in Quantum Circuit Mapping

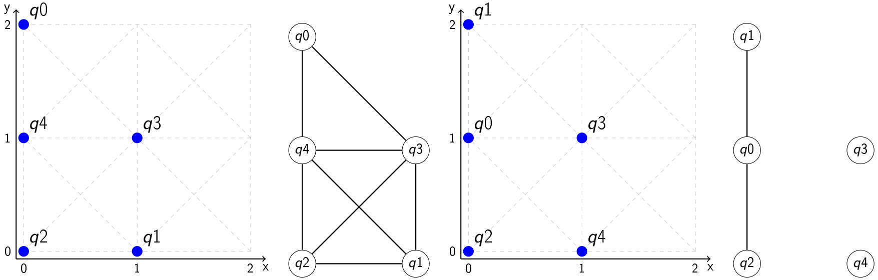
- **DasAtom** uses subgraph isomorphism to map quantum circuits onto neutral atom (NA) architectures.
- Ensures each gate in a subcircuit is directly executable.
- Embedding interaction graphs onto the NA grid enhances fidelity and efficiency.

DasAtom Algorithm Overview

The algorithm operates in four main steps:

1. **Partitioning**
   - Divide the quantum circuit into CZ layers and subcircuits.
   - Ensure each layer is fully contained within a subcircuit.
2. **Embedding**
   - Find optimal qubit mappings for each subcircuit onto the NA grid.
3. **Parallel Execution**
   - Execute gates in each subcircuit in parallel.
   - Consider blockade radius constraints of the NA architecture.
4. **Routing via Atom Shuttling**
   - Use atom shuttling to transition between different qubit mappings.
   - Enable efficient, high-fidelity transformations.

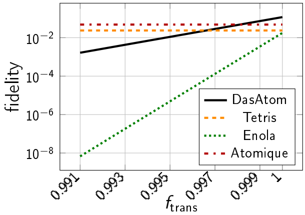
Running example

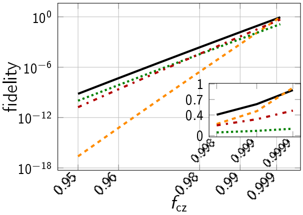


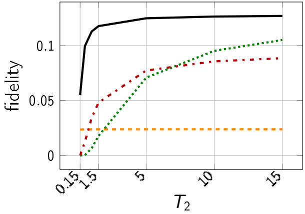
Performance Comparison with Existing Methods

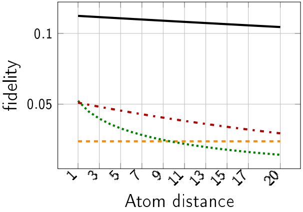
- **Fidelity Improvements**
  - 414x over Enola
  - 10.6x over Tetris
  - 16.1x over Atomique
  - Evaluated on a 30-qubit Quantum Fourier Transform (QFT) circuit
- **Reduced Compilation Time**
  - Significantly shorter runtimes for larger circuits
  - Outperforms existing methods on complex topologies

Ablation studies on QFT-20









Directions for Future Research

- **Approximate Subgraph Isomorphism**
  - Develop algorithms for approximate mappings in large circuits
- **Optimized Embeddings**
  - Design embeddings to minimize atom movement costs
  - Use search-based methods to reduce movement time and distance
- **Integration with Realistic NA Devices**
  - Support native multi-qubit gates like CCZ
  - Incorporate realistic atom movement protocols