Reducing Runtime Overhead via Use-Based Migration in Neutral Atom Quantum Architectures

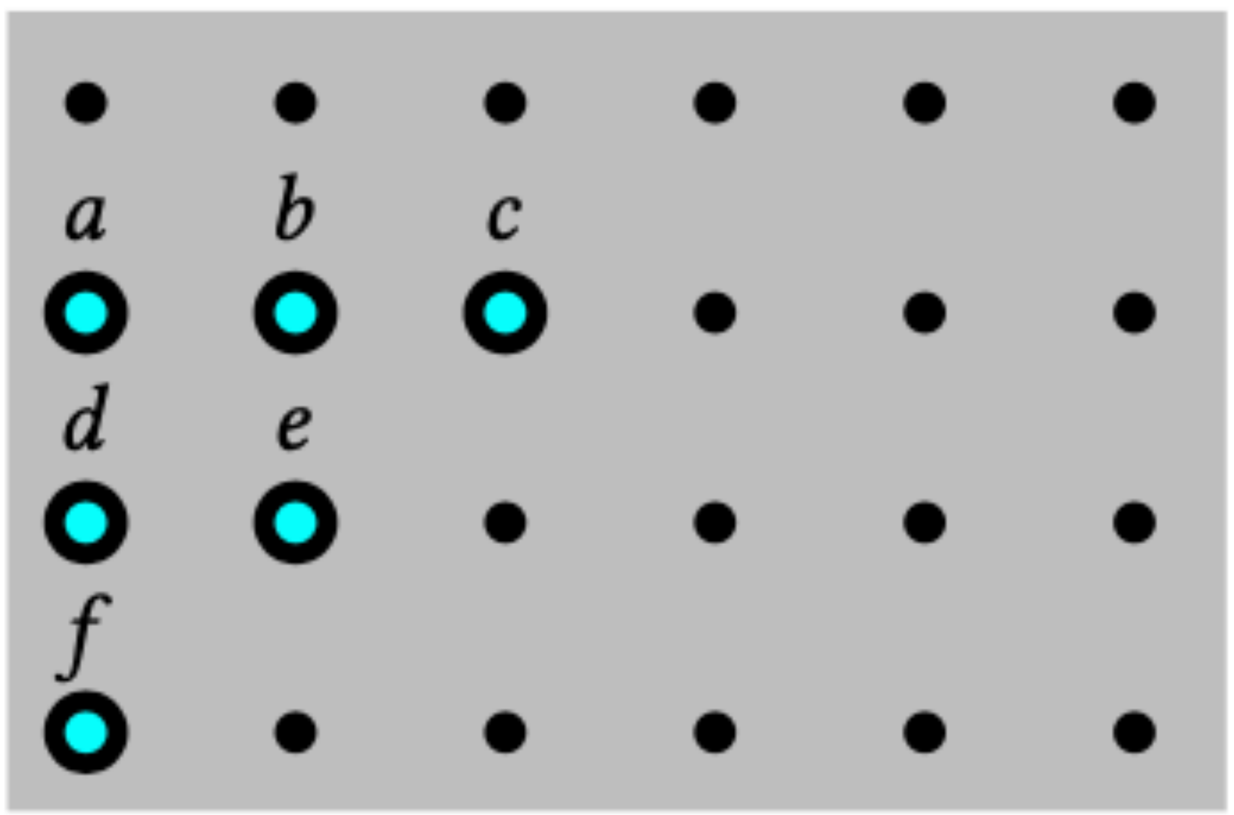
Andrew Litteken, Jonathan M. Baker, Frederick T. Chong
Department of Computer Science
University of Chicago

September 20, 2022 - QCE 2022

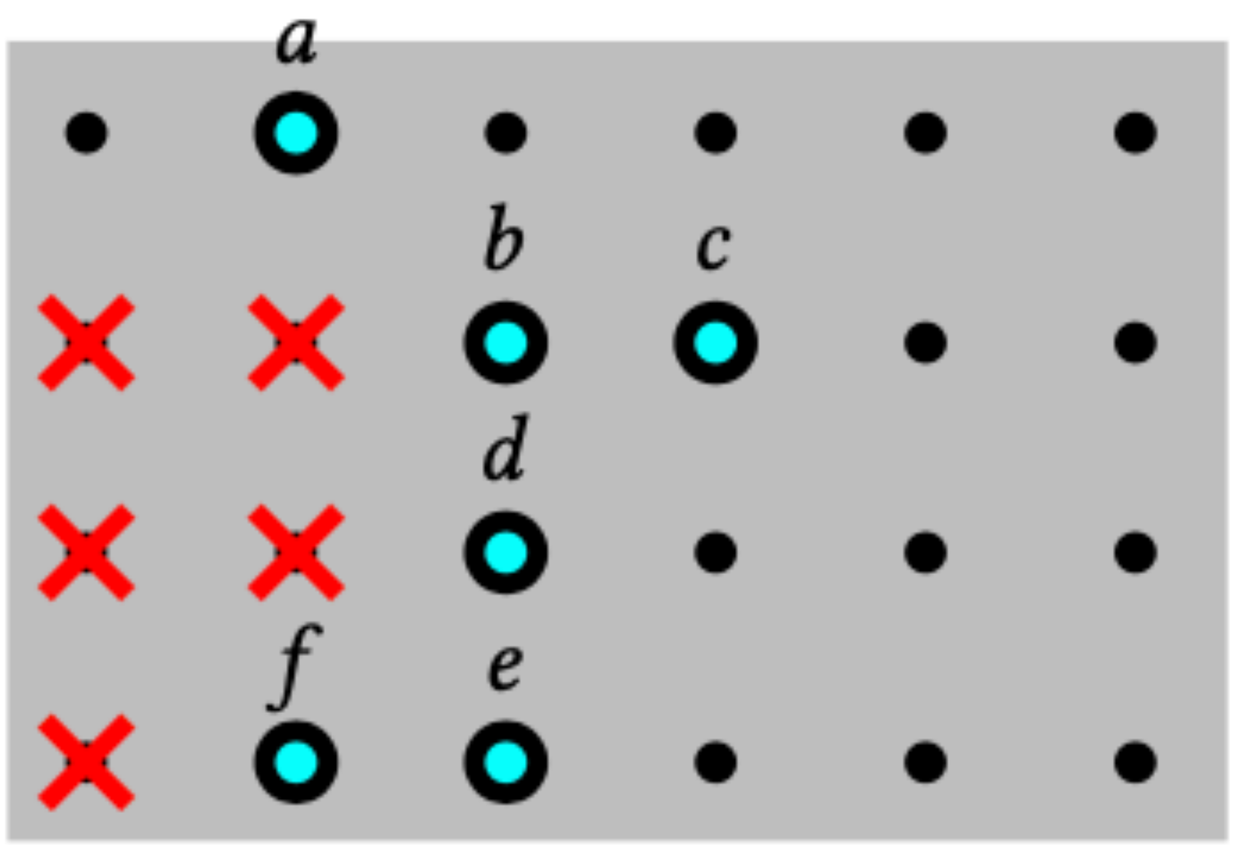
Funded in part by:
NSF Expeditions in Computing 1730449
NSF Phy-1818914/OMA-2016136
DOE DE-SC0020289/0020331/QNEXT

Motivation

Motivation



Initial Mapping

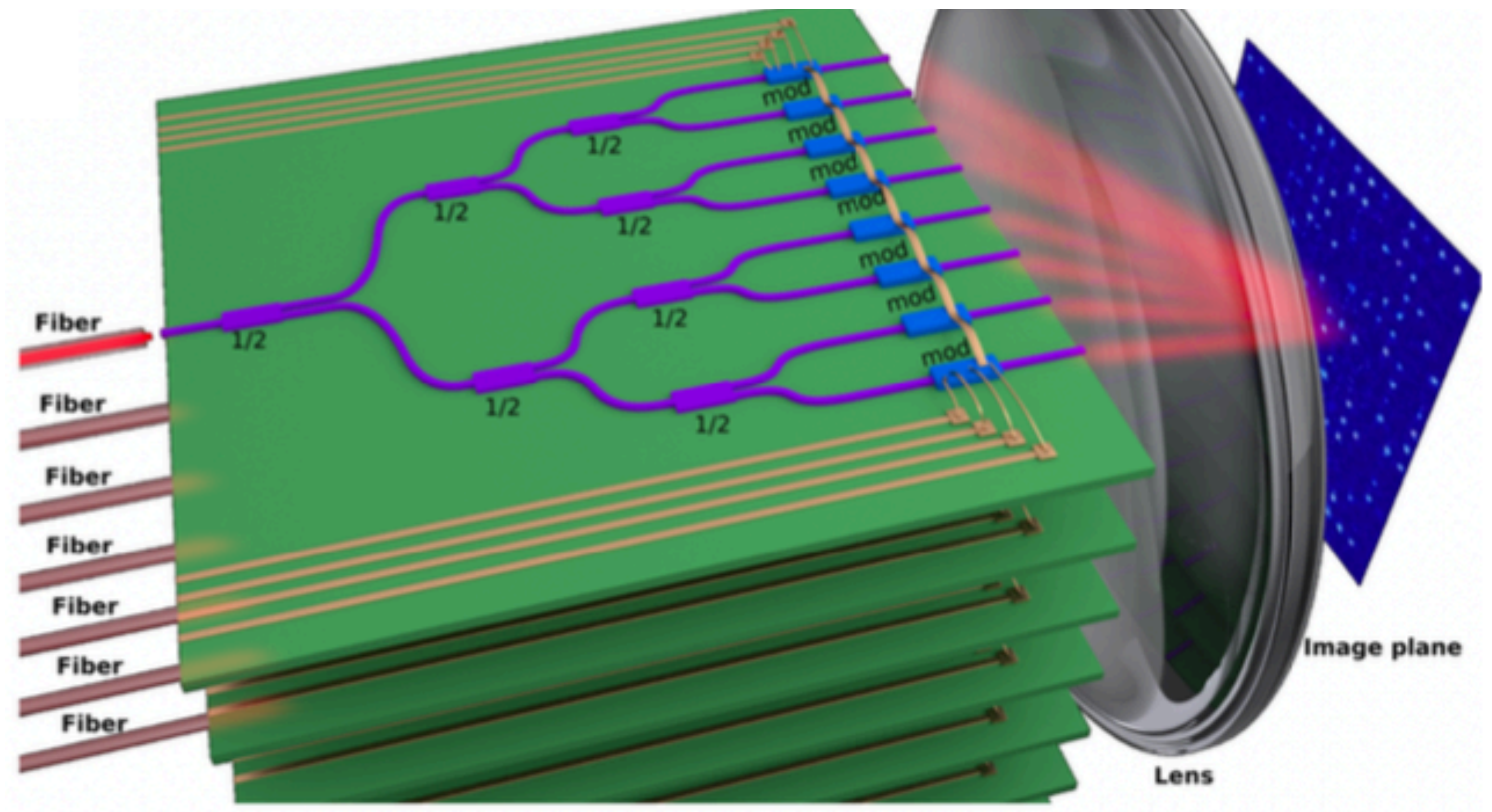


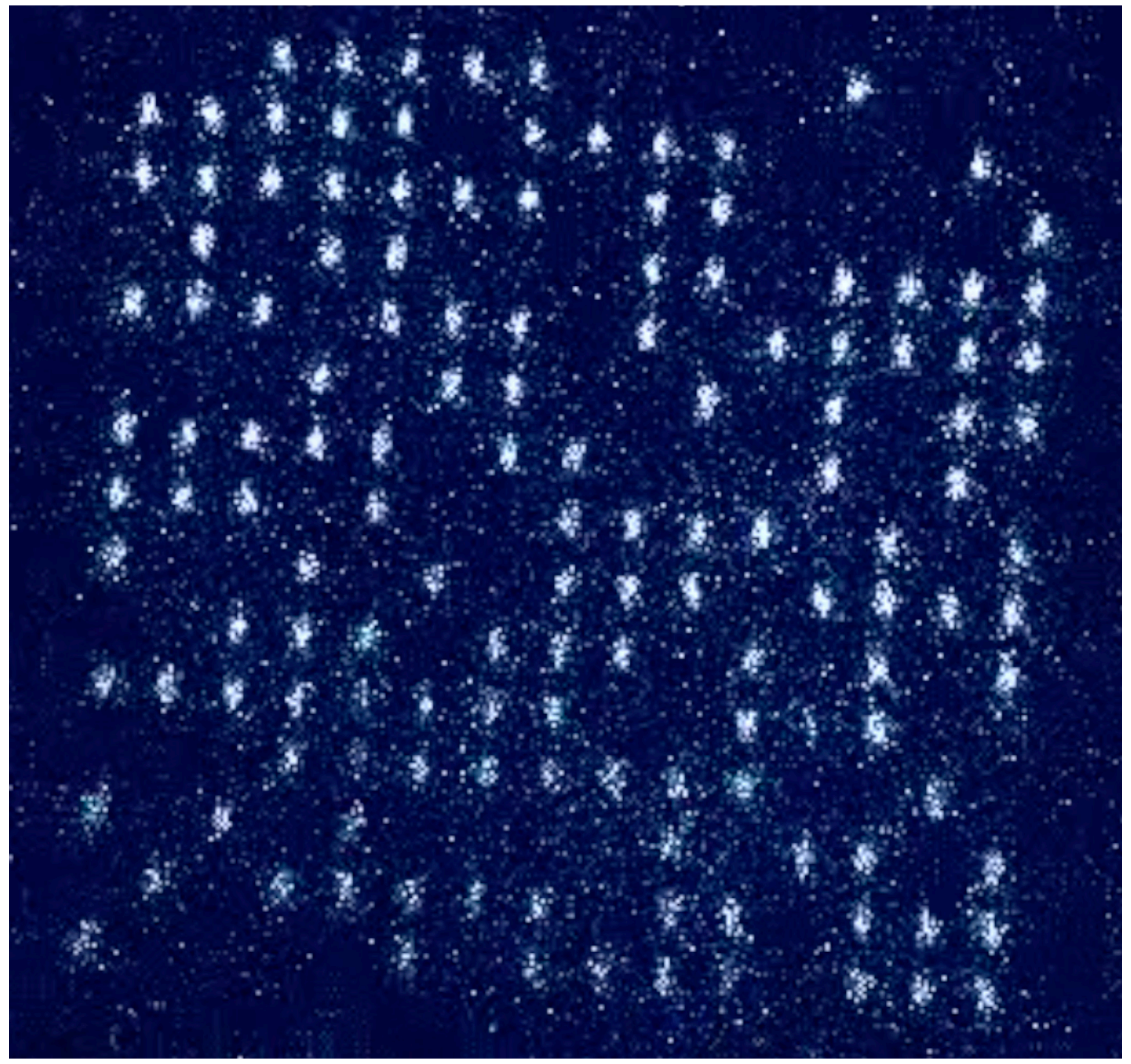
After Atom Loss

Outline

- Neutral Atom Architecture Background
- New Atom Loss Mitigation Strategies
    - Interaction Graph Based Remapping
    - Use-Based Relocation
    - Parallelism
- Conclusion

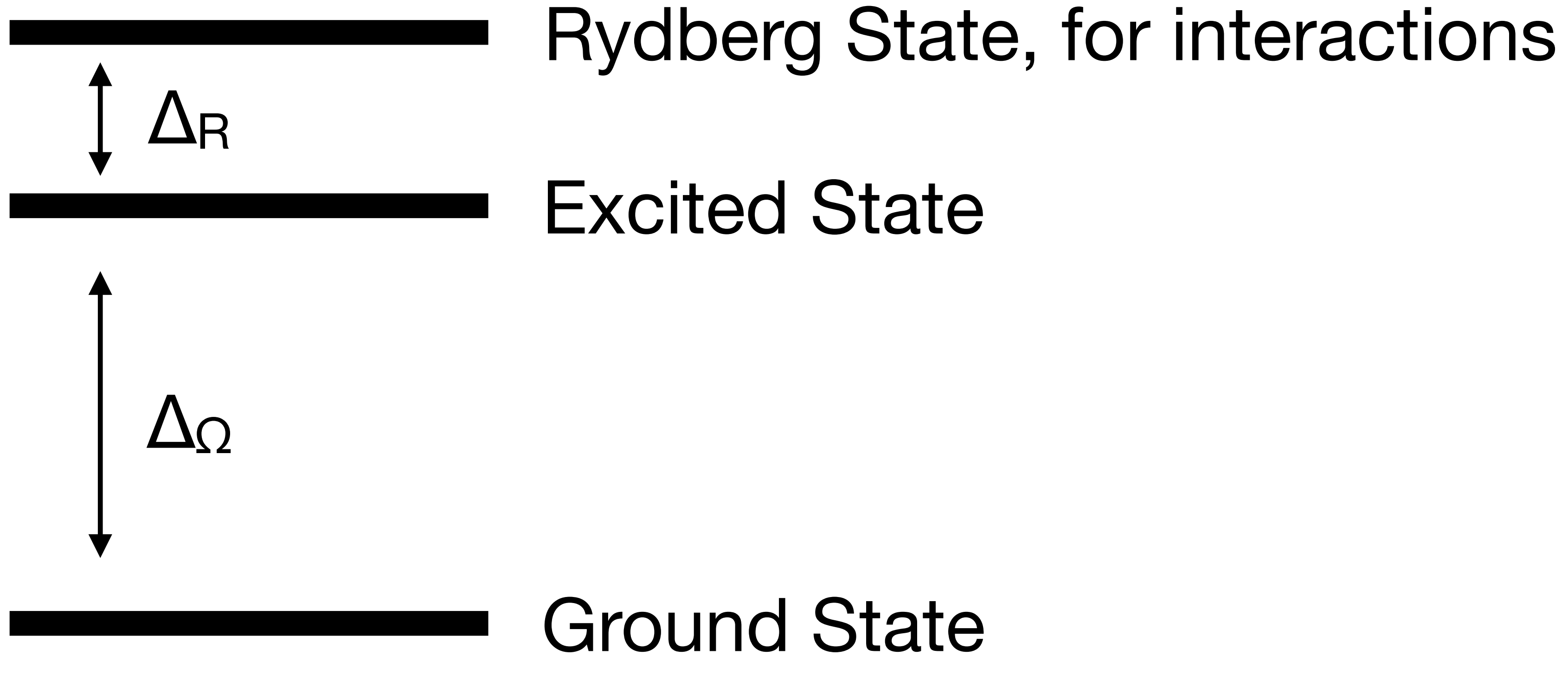
Neutral Atom Architectures





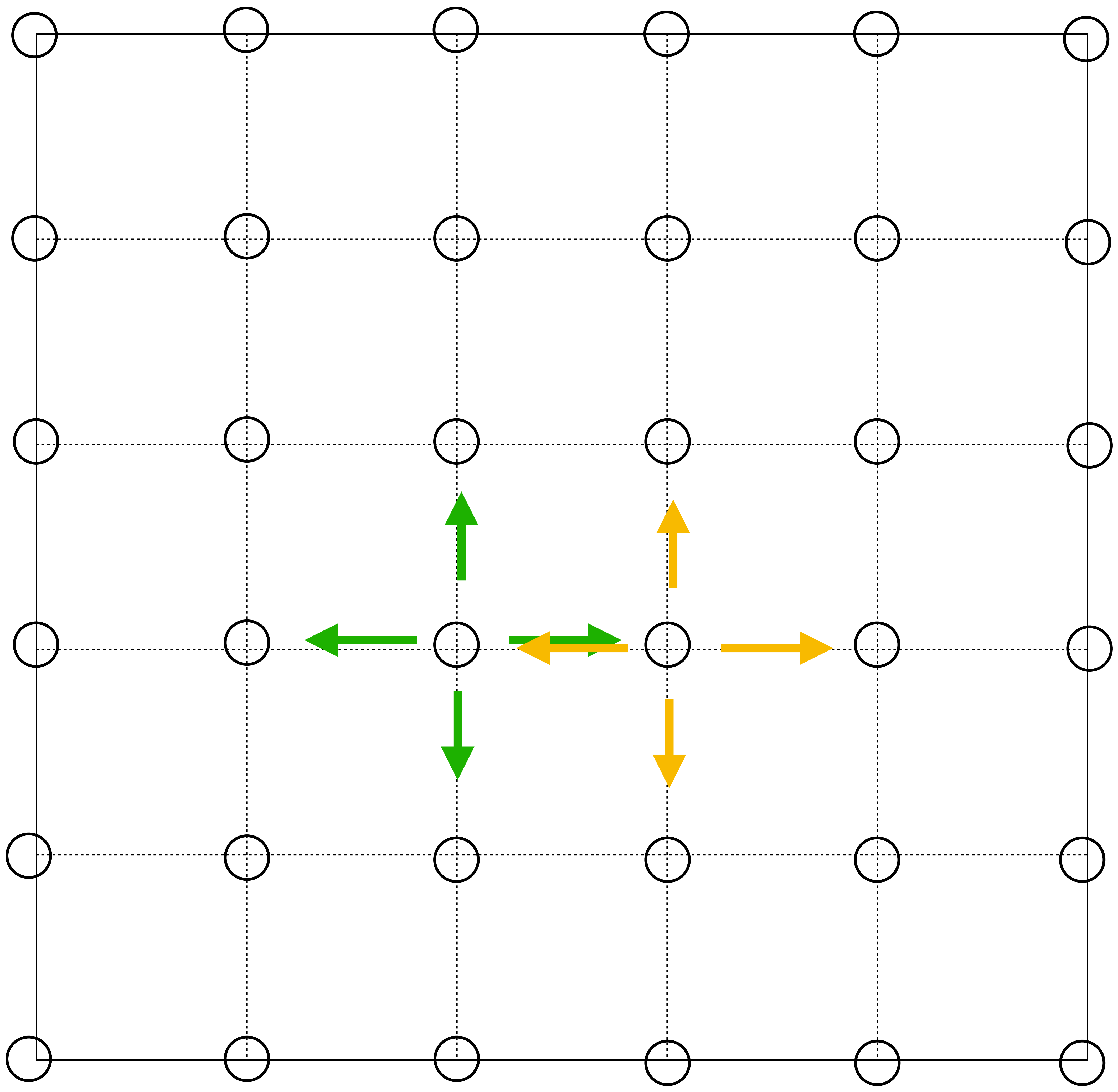
Trapped Neutral Atoms

State of an Atom

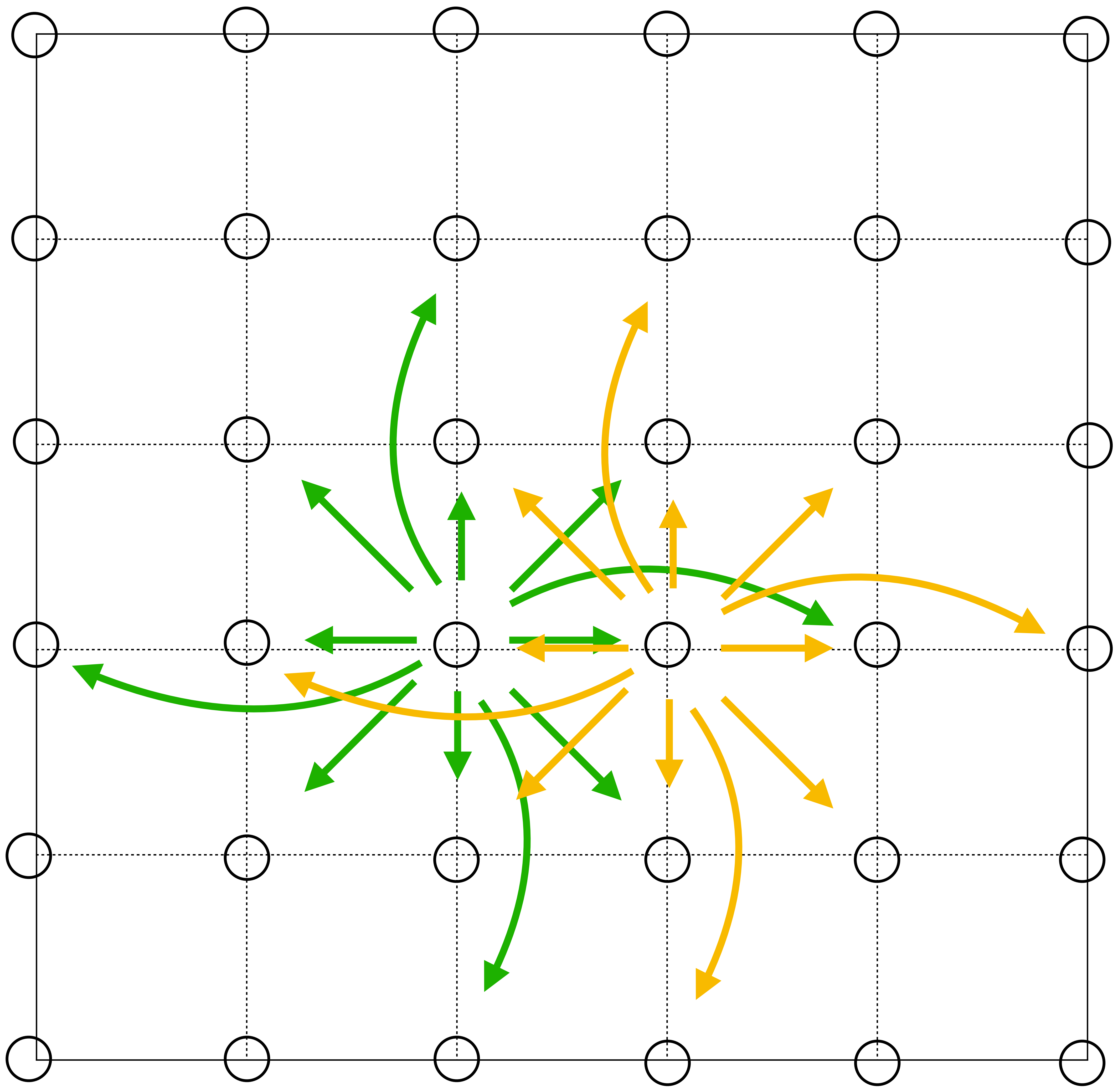


Neutral Atom Architectures

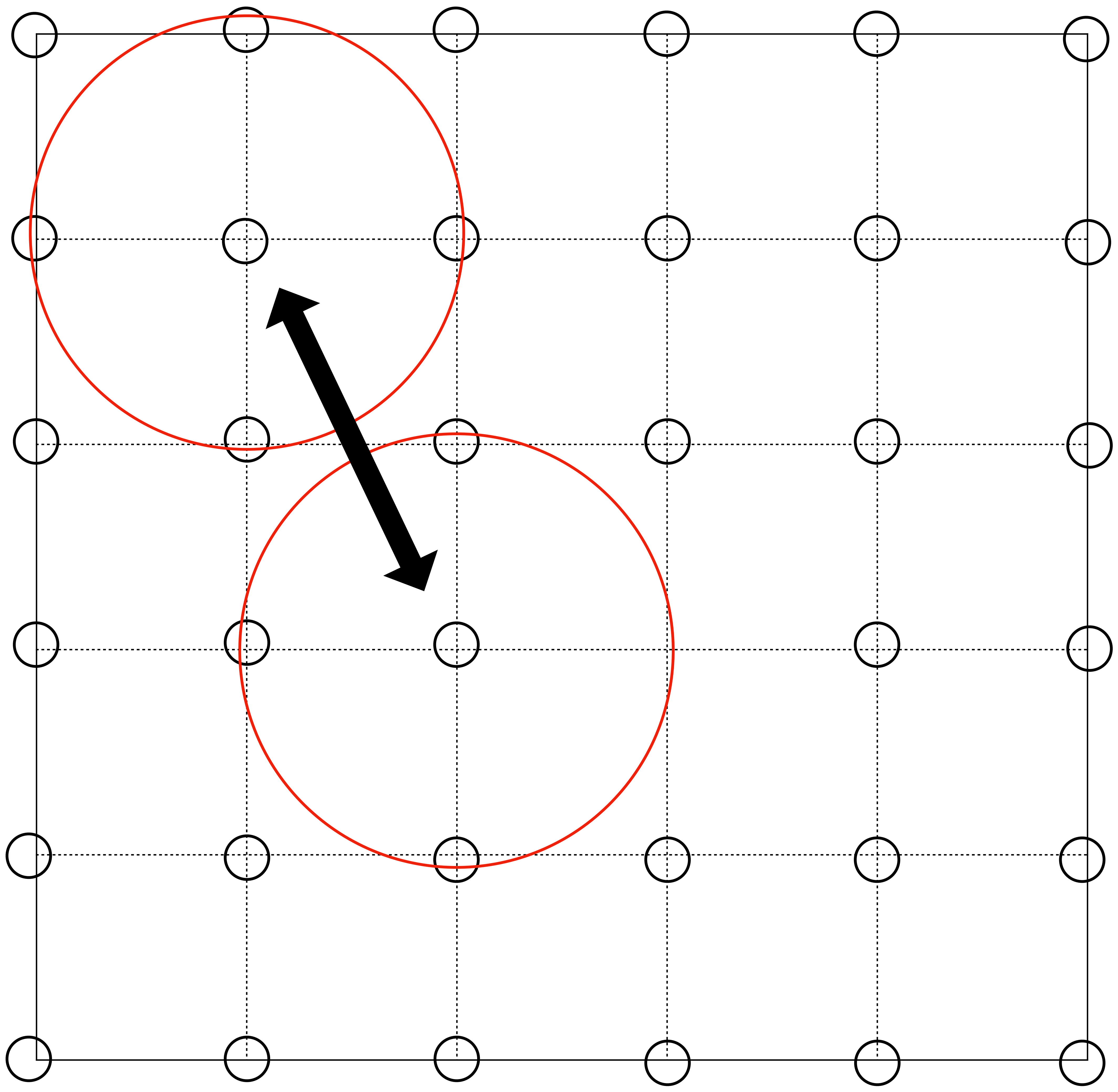
Neutral Atom Architectures



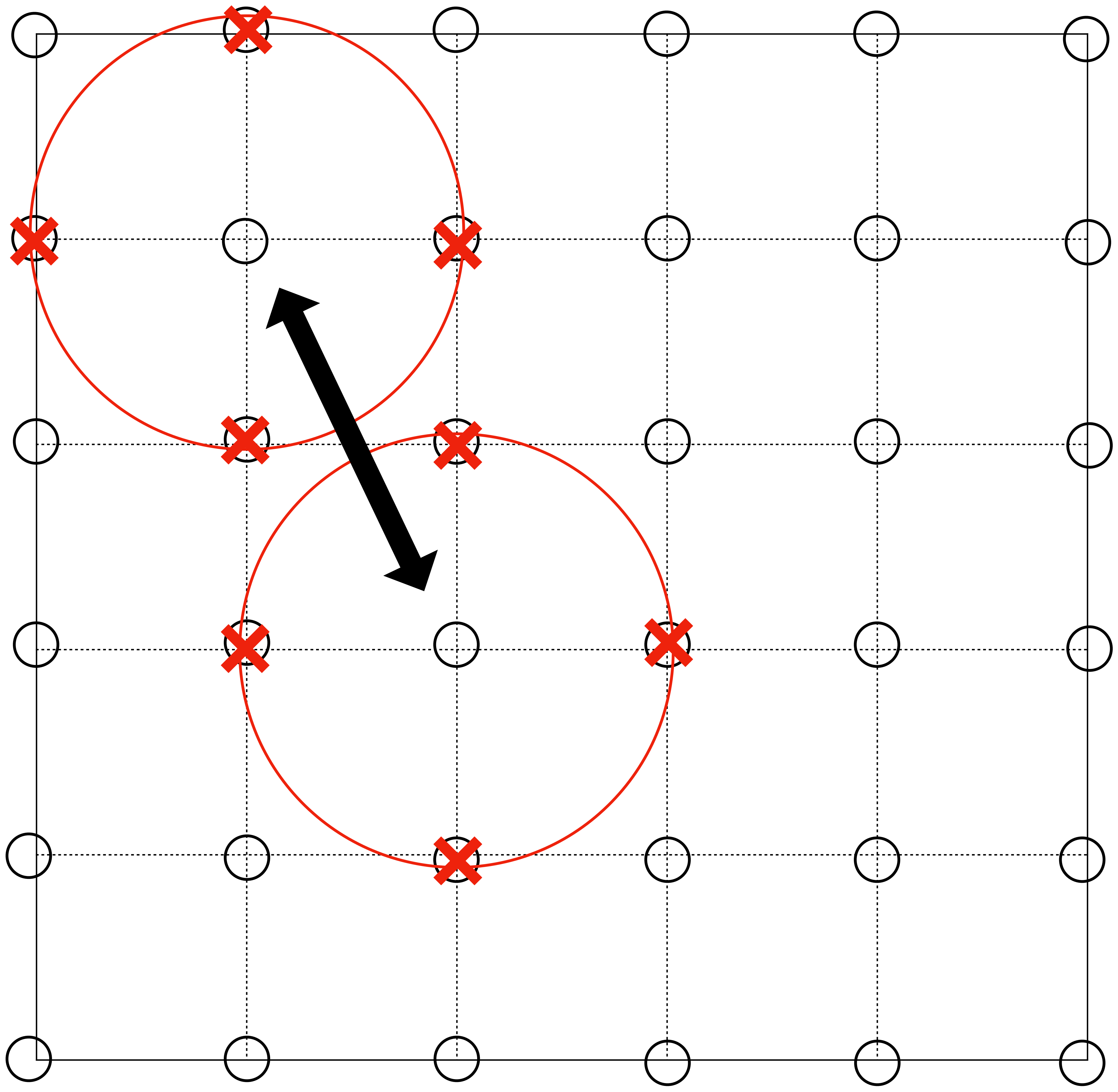
Neutral Atom Architectures



Neutral Atom Background



Neutral Atom Background

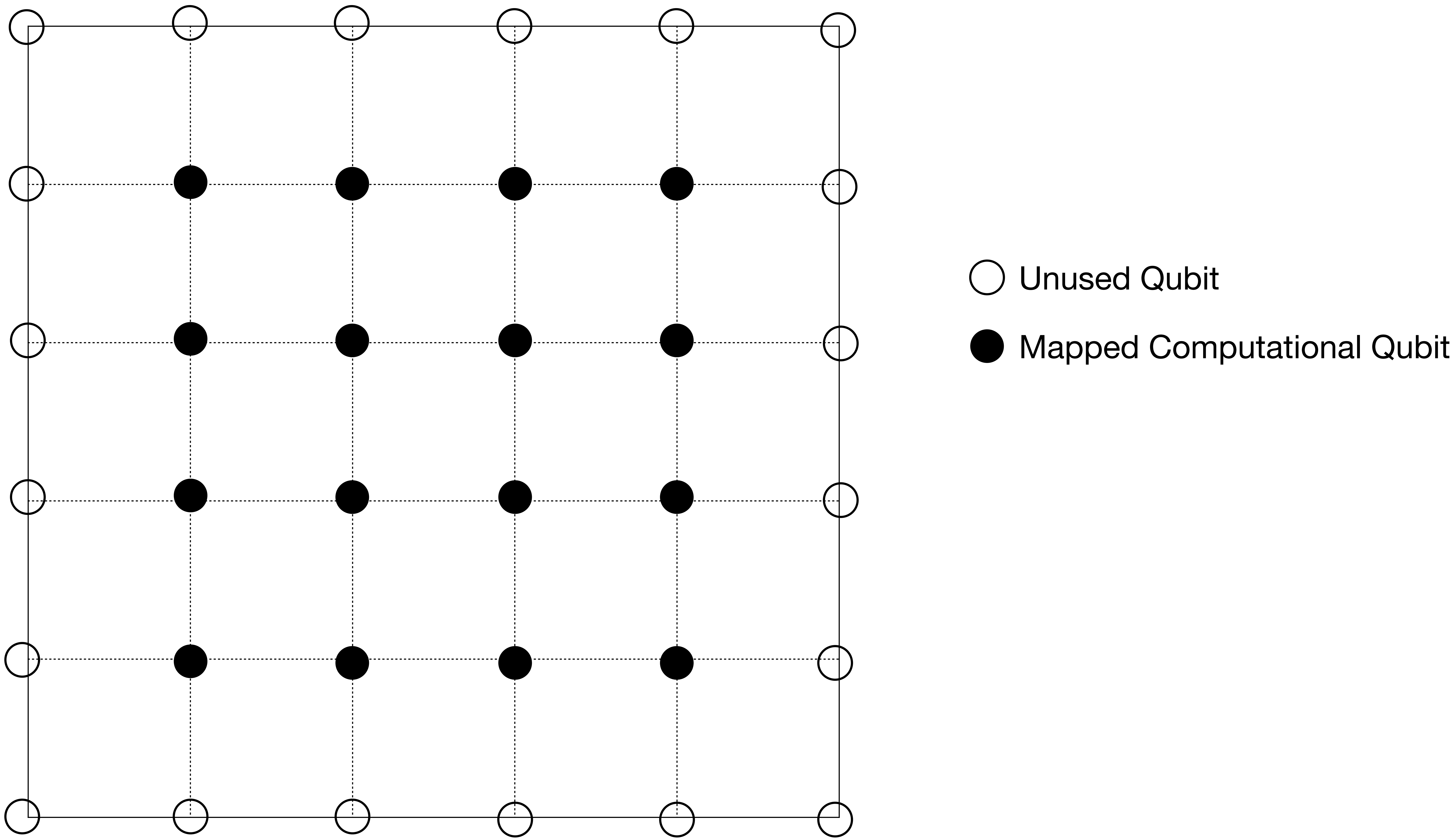


Neutral Atom Compilation

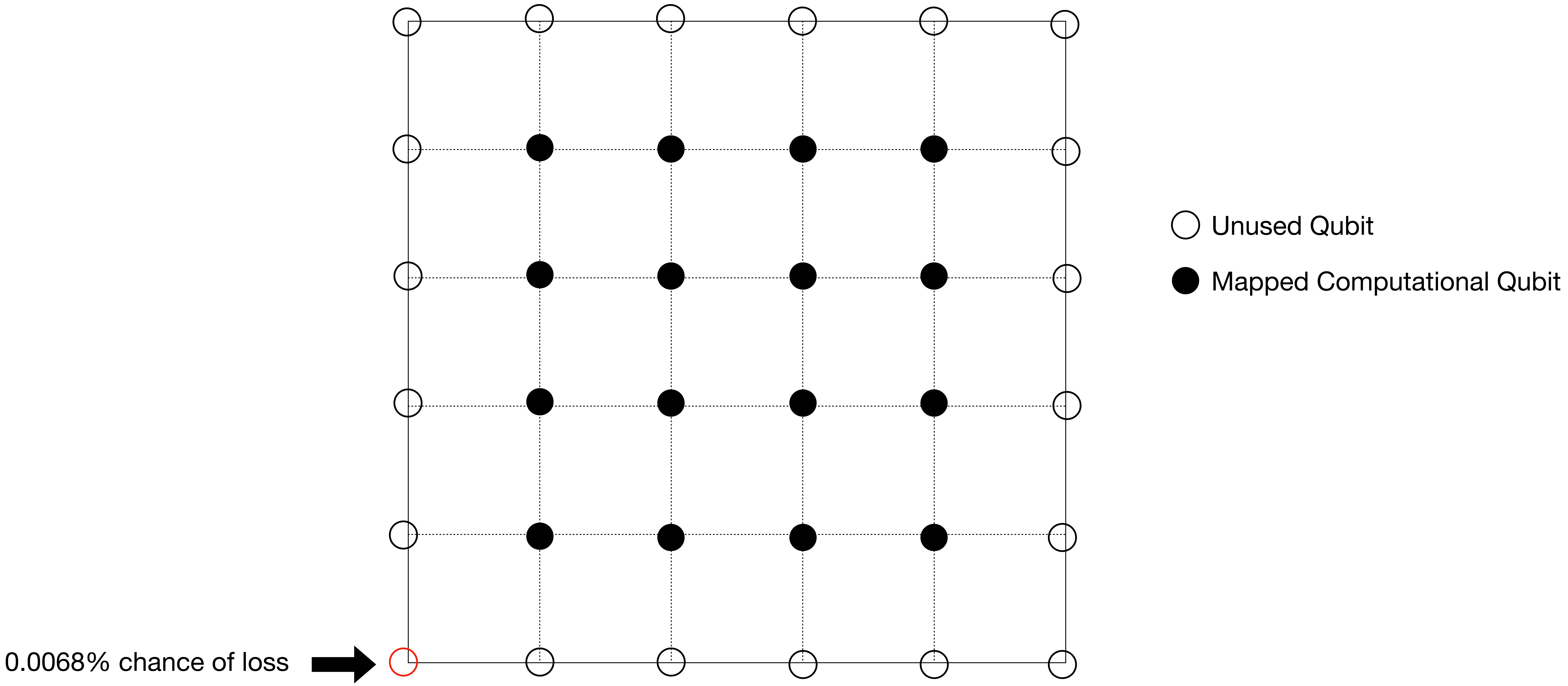
- J. M. Baker, et. al. , "Exploiting Long-Distance Interactions and Tolerating Atom Loss in Neutral Atom Quantum Architectures," 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA), 2021, pp. 818-831, doi: 10.1109/ISCA52012.2021.00069.

- Tirthak Patel, et. al. 2022. Geyser: a compilation framework for quantum computing with neutral atoms. In Proceedings of the 49th Annual International Symposium on Computer Architecture (ISCA '22). Association for Computing Machinery, New York, NY, USA, 383–395. https://doi.org/10.1145/3470496.3527428

Atom Loss

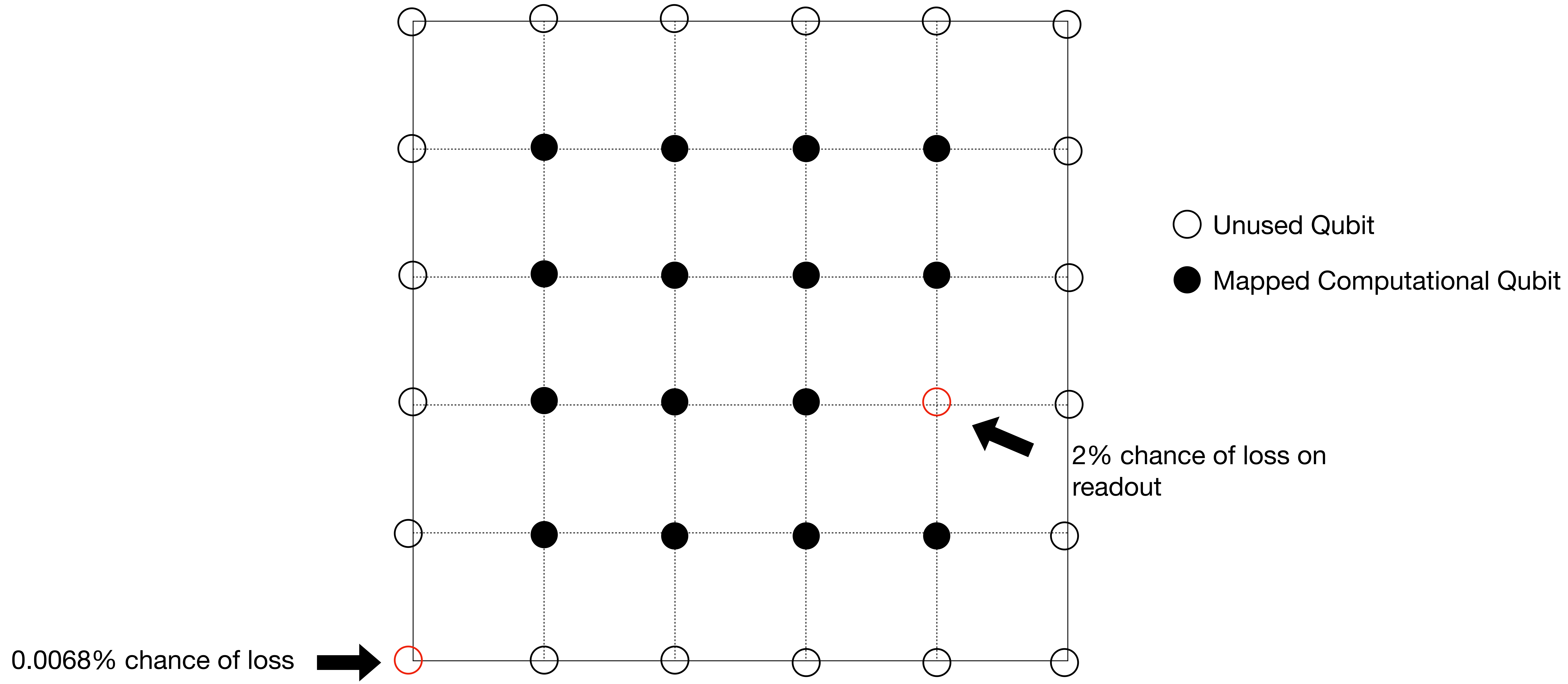


Atom Loss



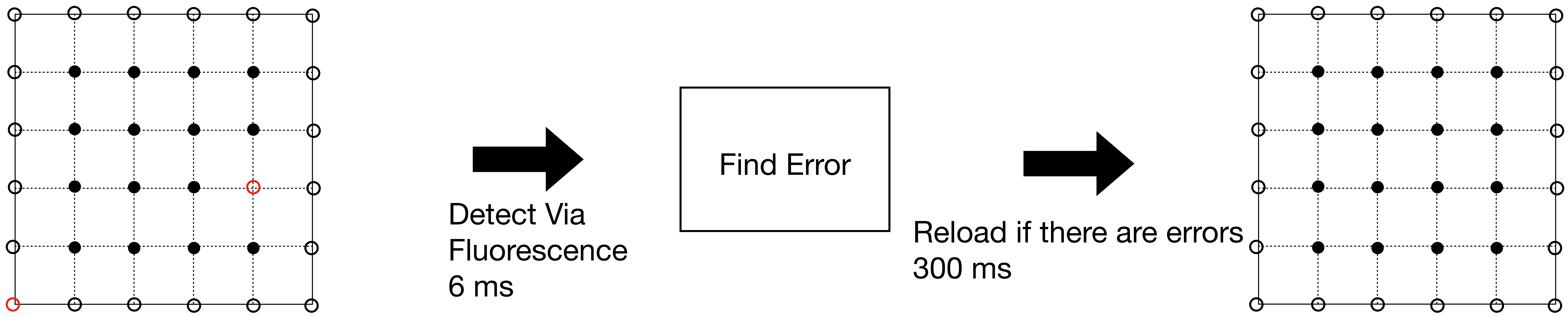
0.0068% chance of loss
Unused Qubit
Mapped Computational Qubit

Atom Loss



Unused Qubit
Mapped Computational Qubit
2% chance of loss on readout
0.0068% chance of loss

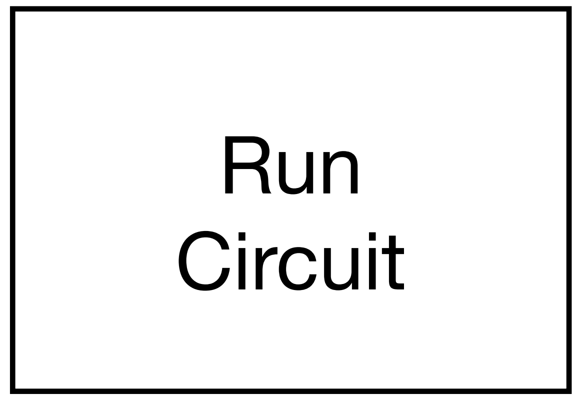
Atom Loss



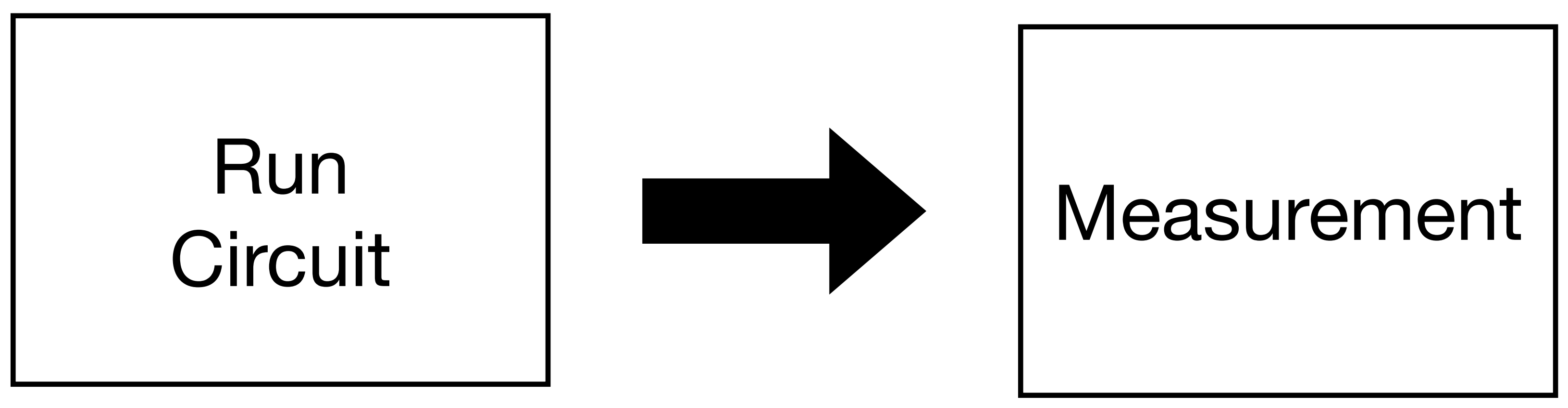
Detect Via
Fluorescence
6 ms
Find Error
Reload if there are errors
300 ms

Neutral Atom Execution Flow

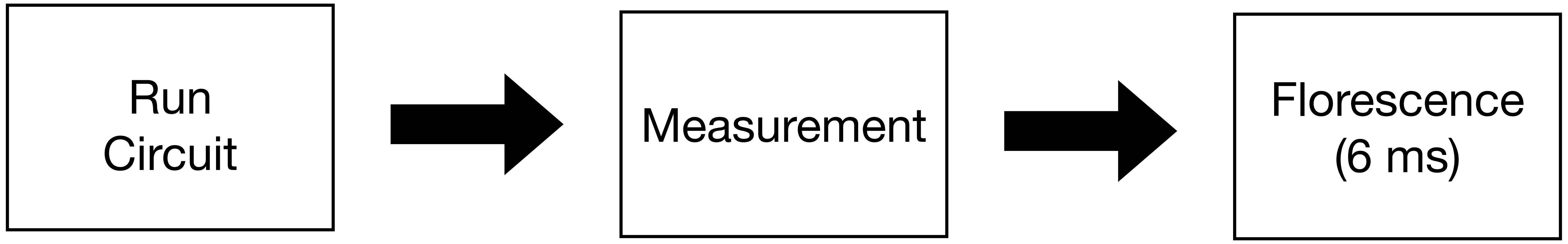
Neutral Atom Execution Flow



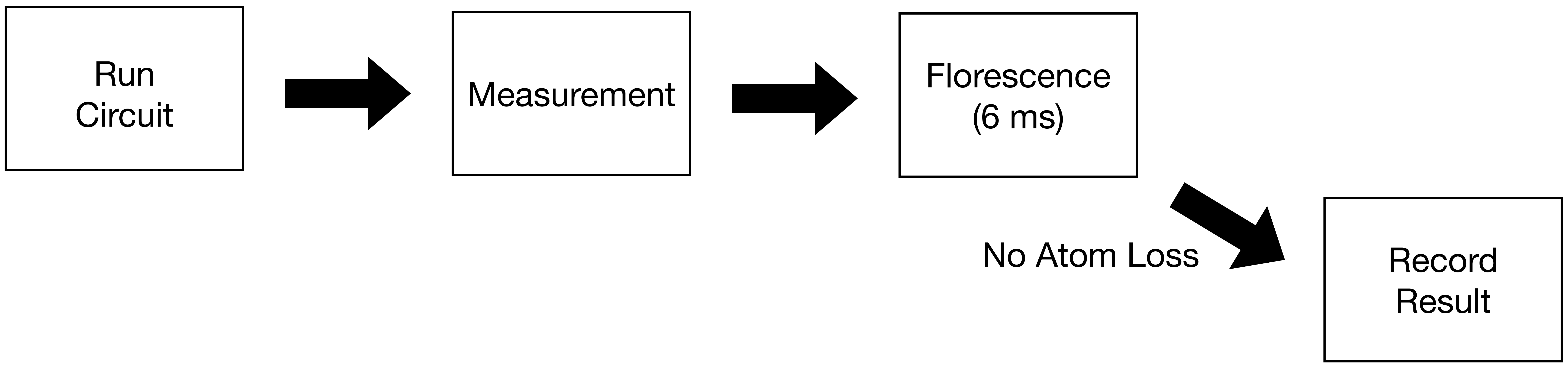
Neutral Atom Execution Flow



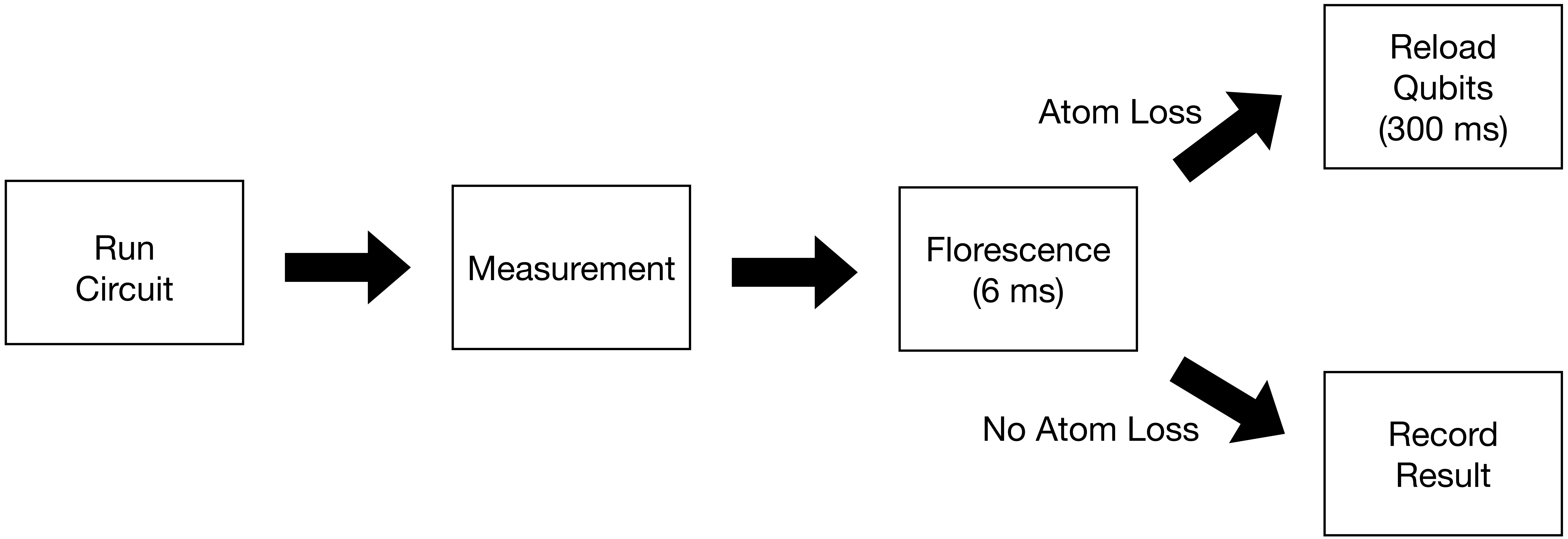
Neutral Atom Execution Flow



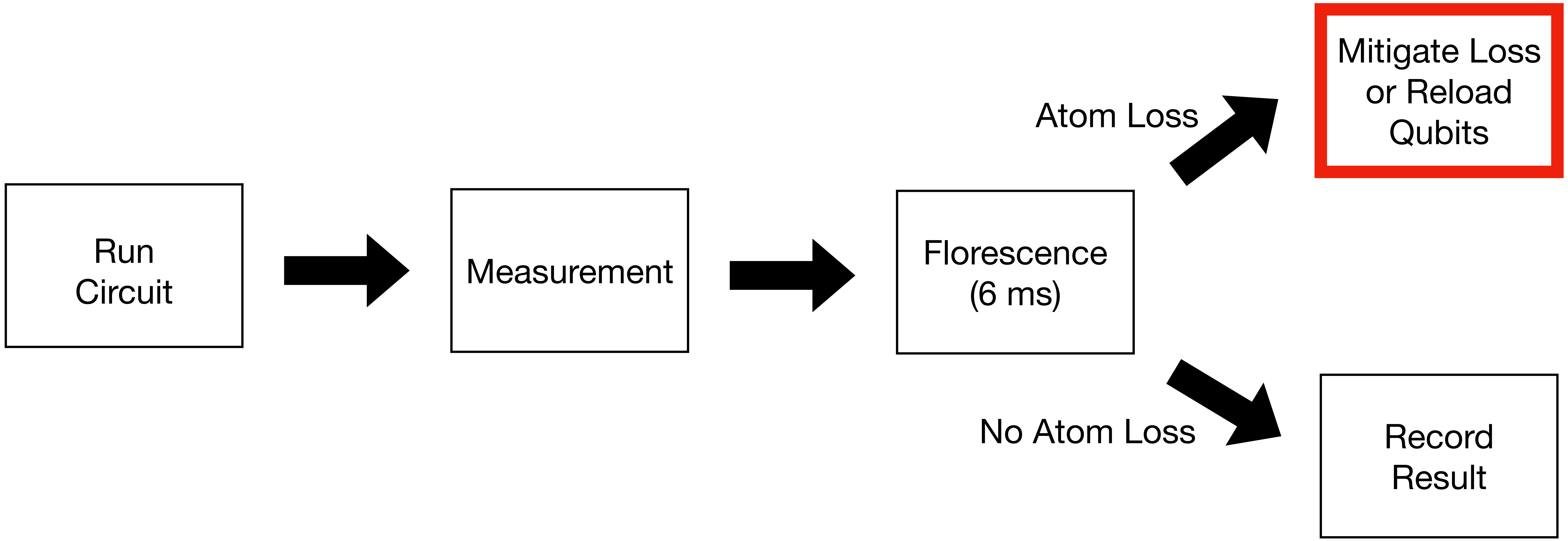
Neutral Atom Execution Flow



Neutral Atom Execution Flow



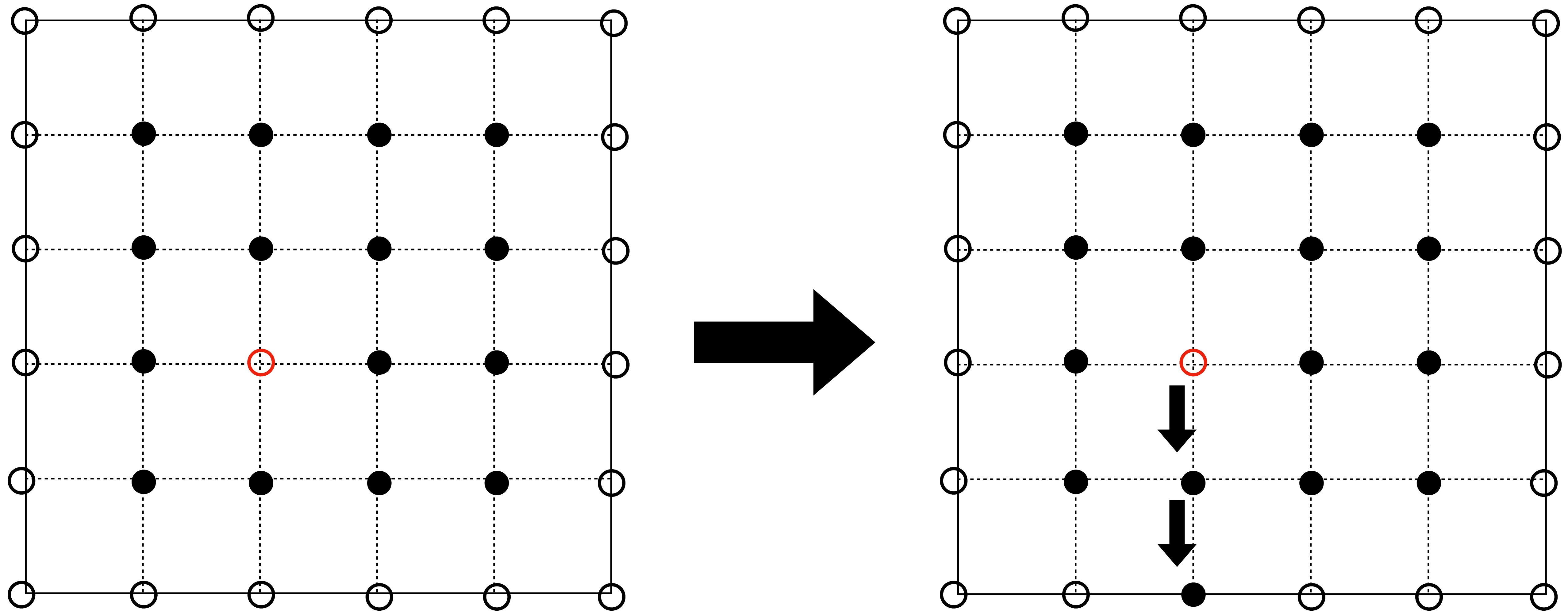
Neutral Atom Execution Flow



Atom Loss Techniques

- Always Reload
- Recompile for New Architecture
- Compile Small + Virtual Remapping + Rerouting

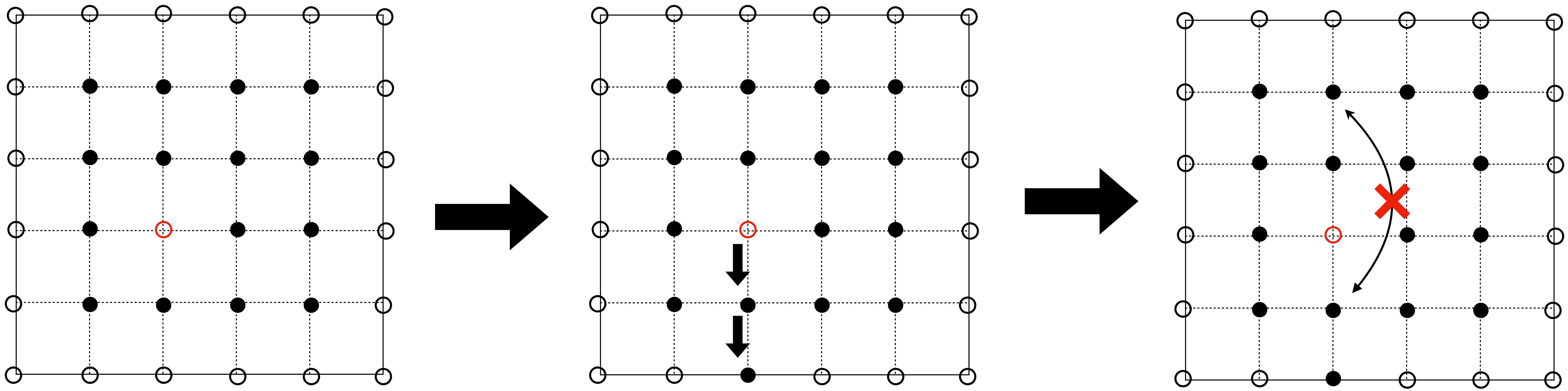
Compile Small + Virtual Remapping + Rerouting (Baker et al, ISCA ’21)



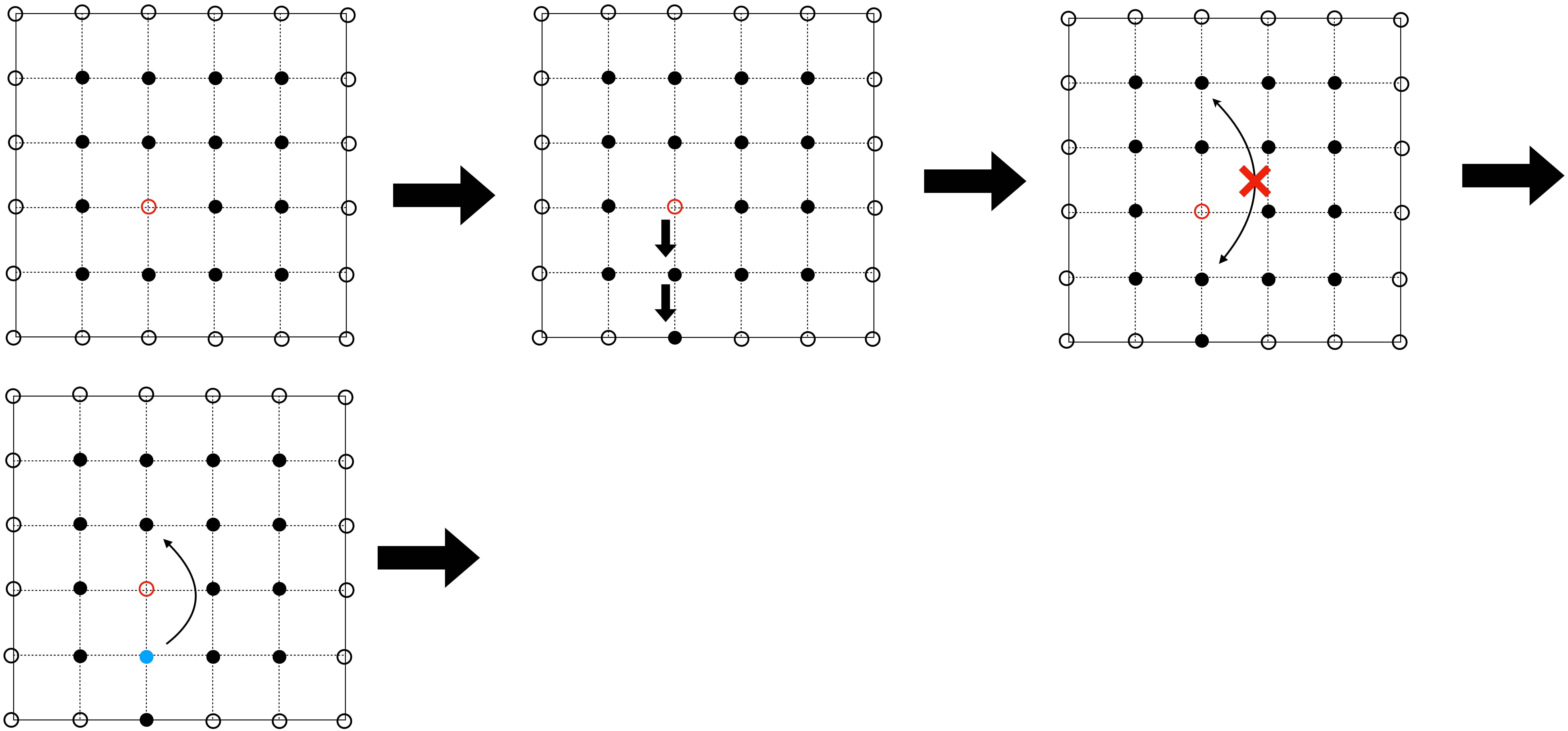
(3, 2) → (4, 2)

(4, 2) → (5, 2)

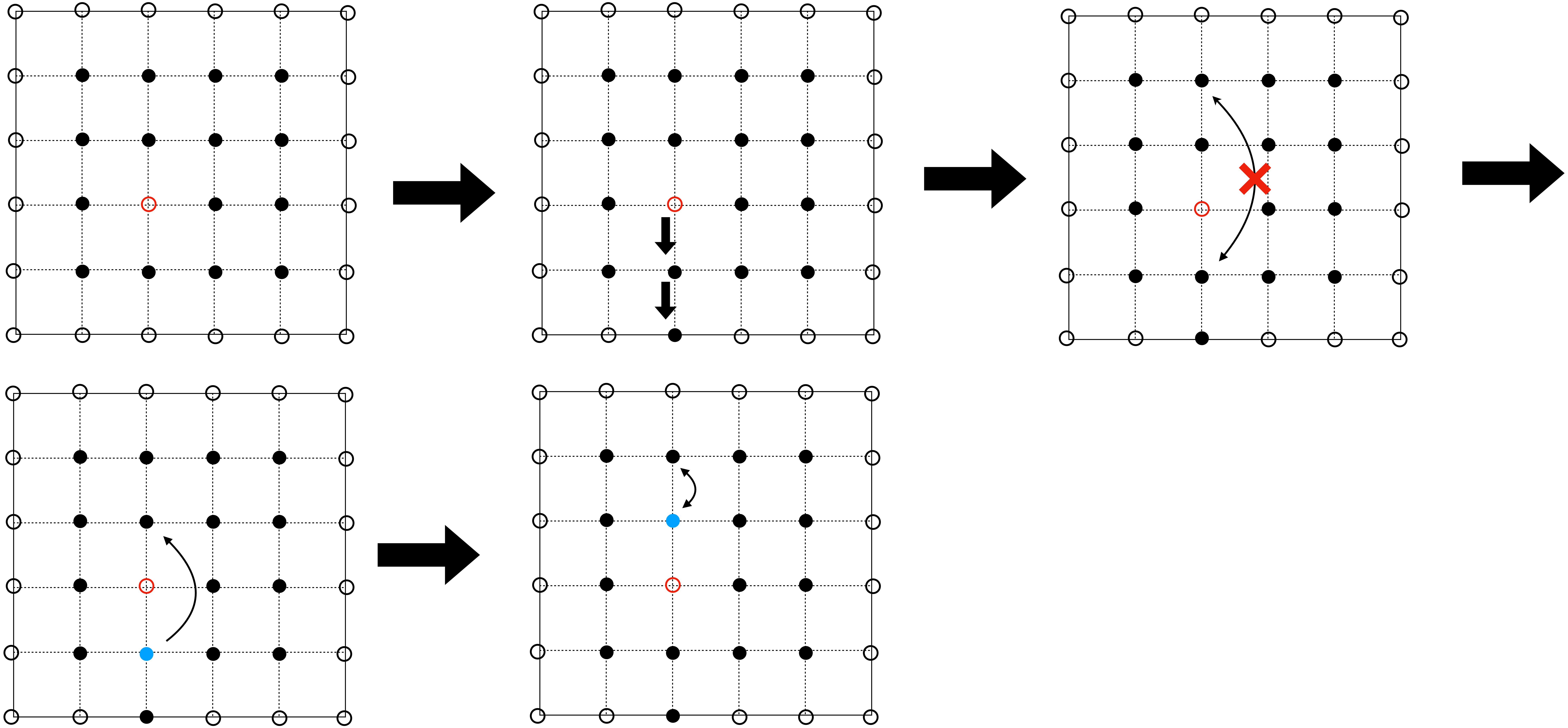
Compile Small + Virtual Remapping + Rerouting (Baker et al, ISCA ’21)



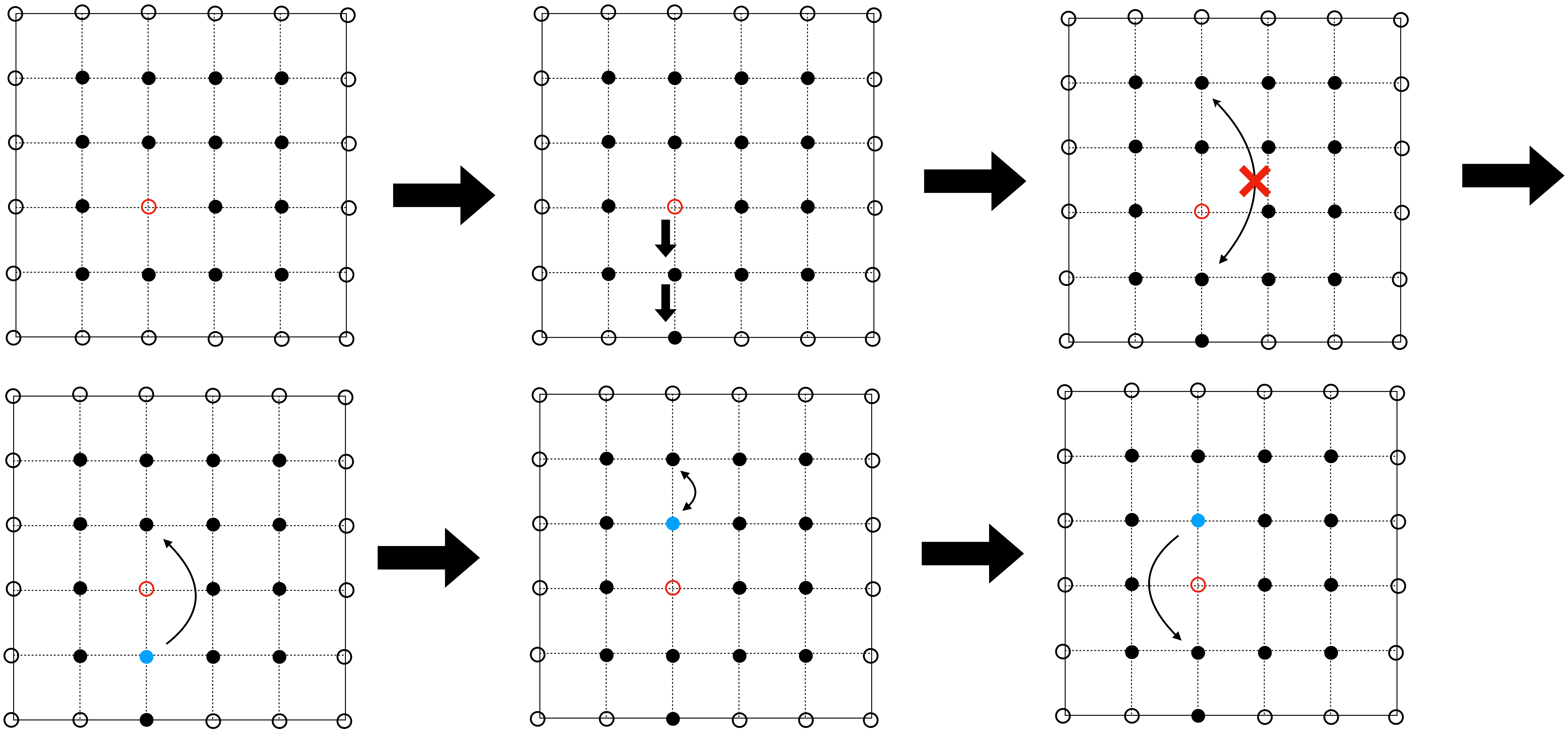
Compile Small + Virtual Remapping + Rerouting (Baker et al, ISCA ’21)



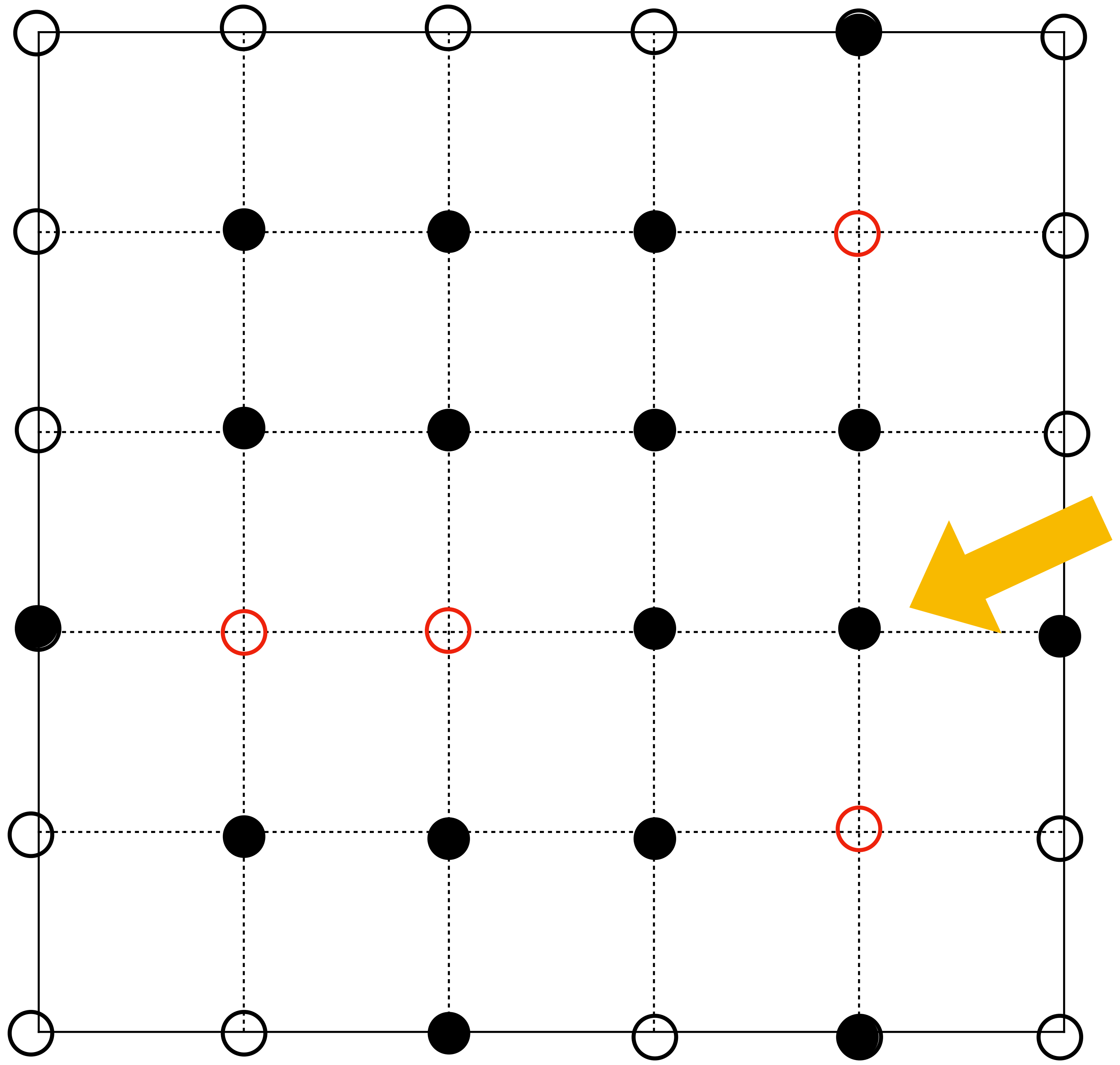
Compile Small + Virtual Remapping + Rerouting (Baker et al, ISCA '21)



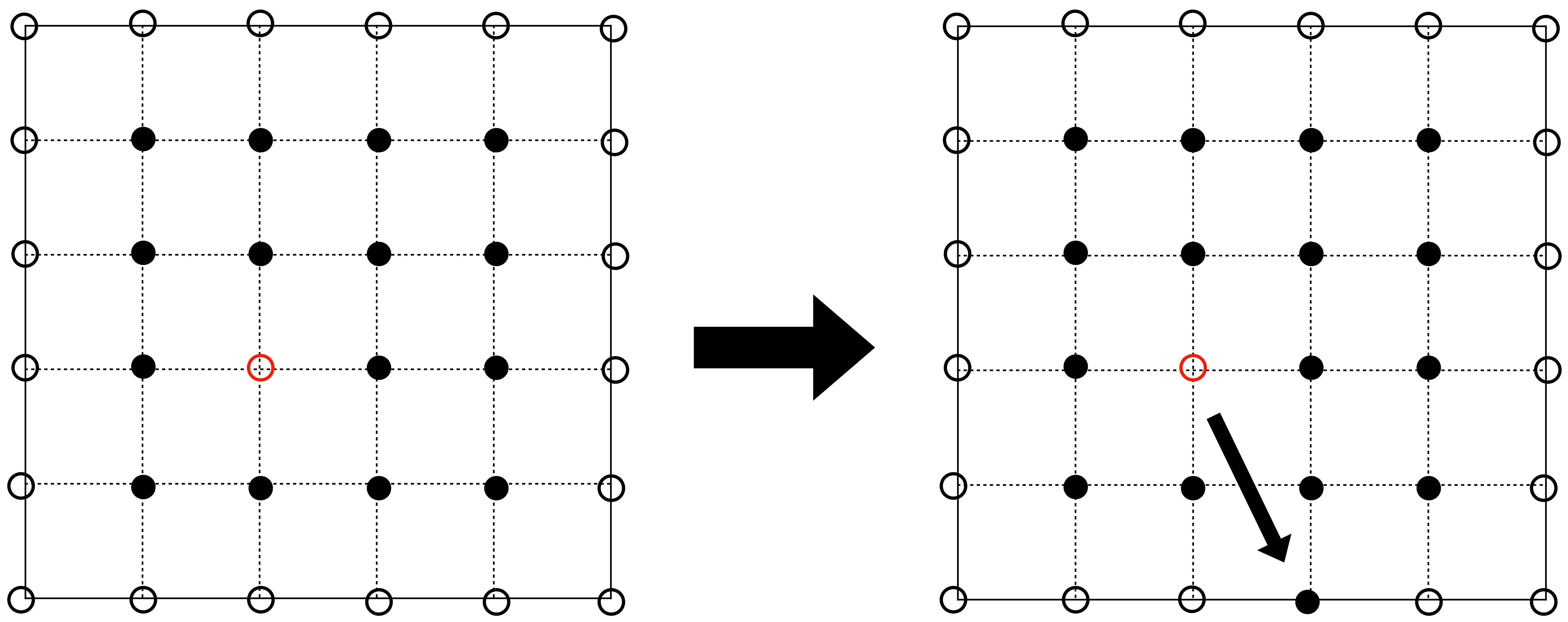
Compile Small + Virtual Remapping + Rerouting (Baker et al, ISCA '21)



New Strategies: Interaction Graph Remapping

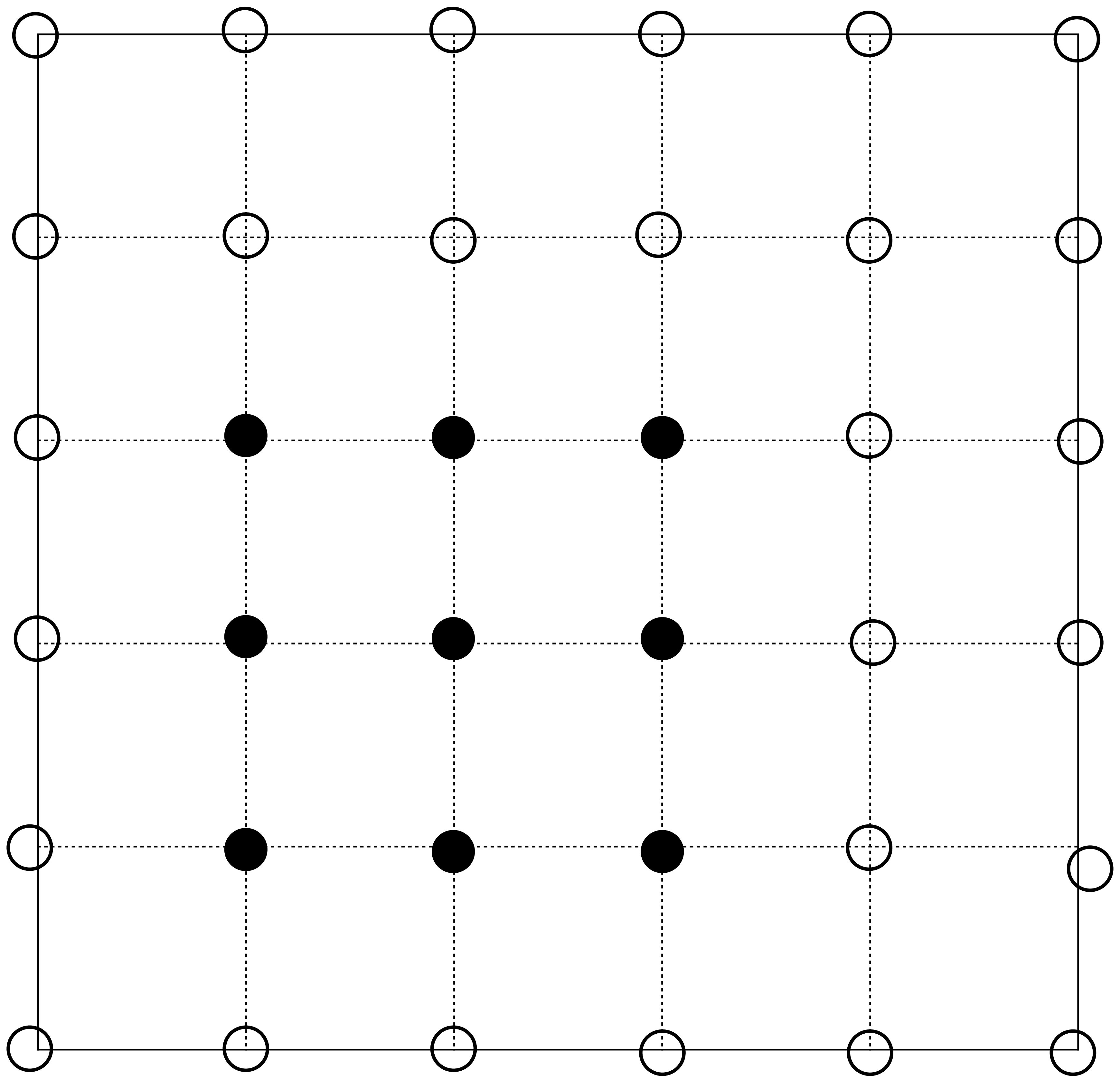


New Strategies: Interaction Graph Remapping

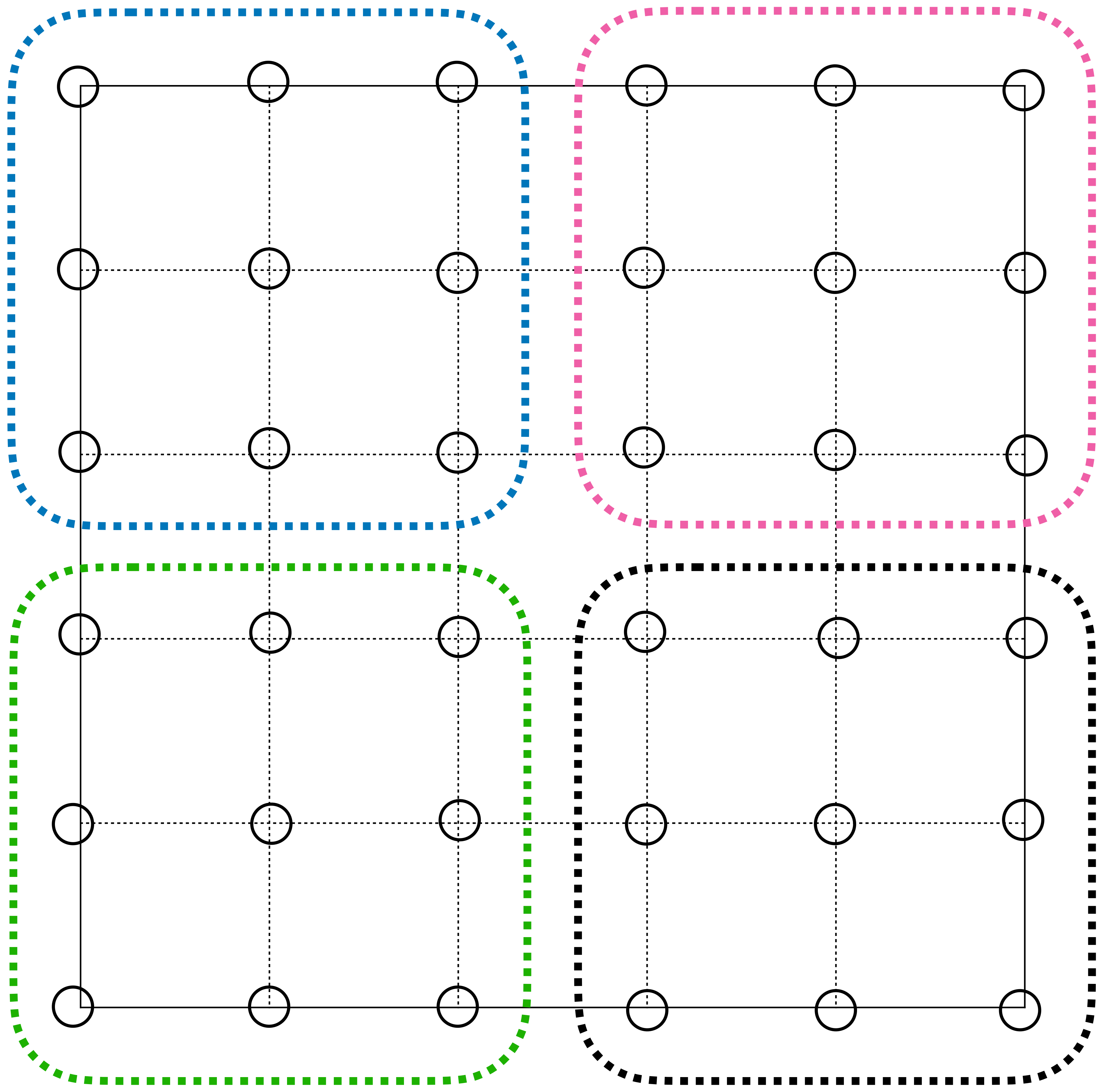


(3, 2) ➡ (5, 3)

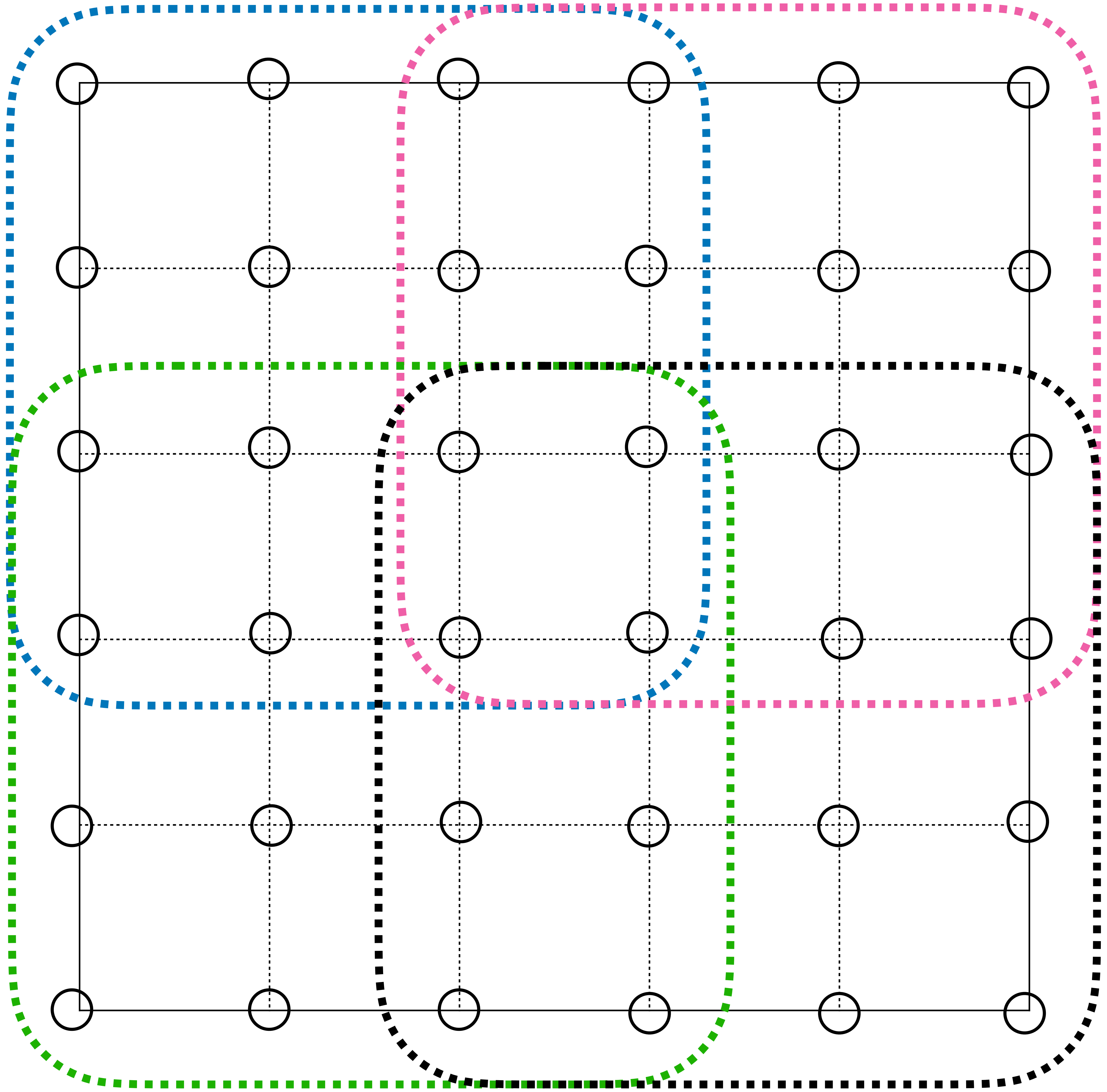
New Strategies: Use-Based Relocation



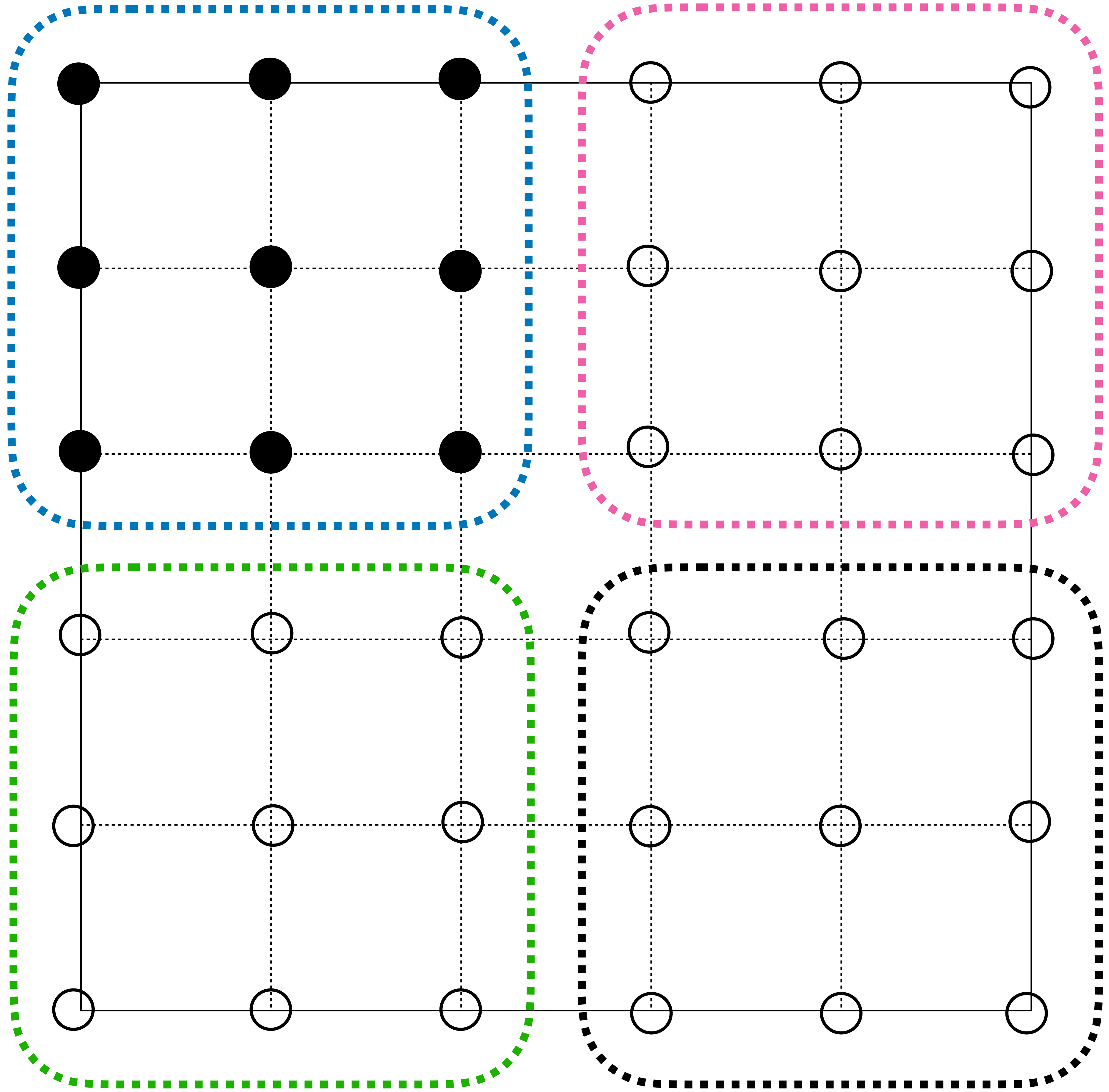
New Strategies: Use-Based Relocation



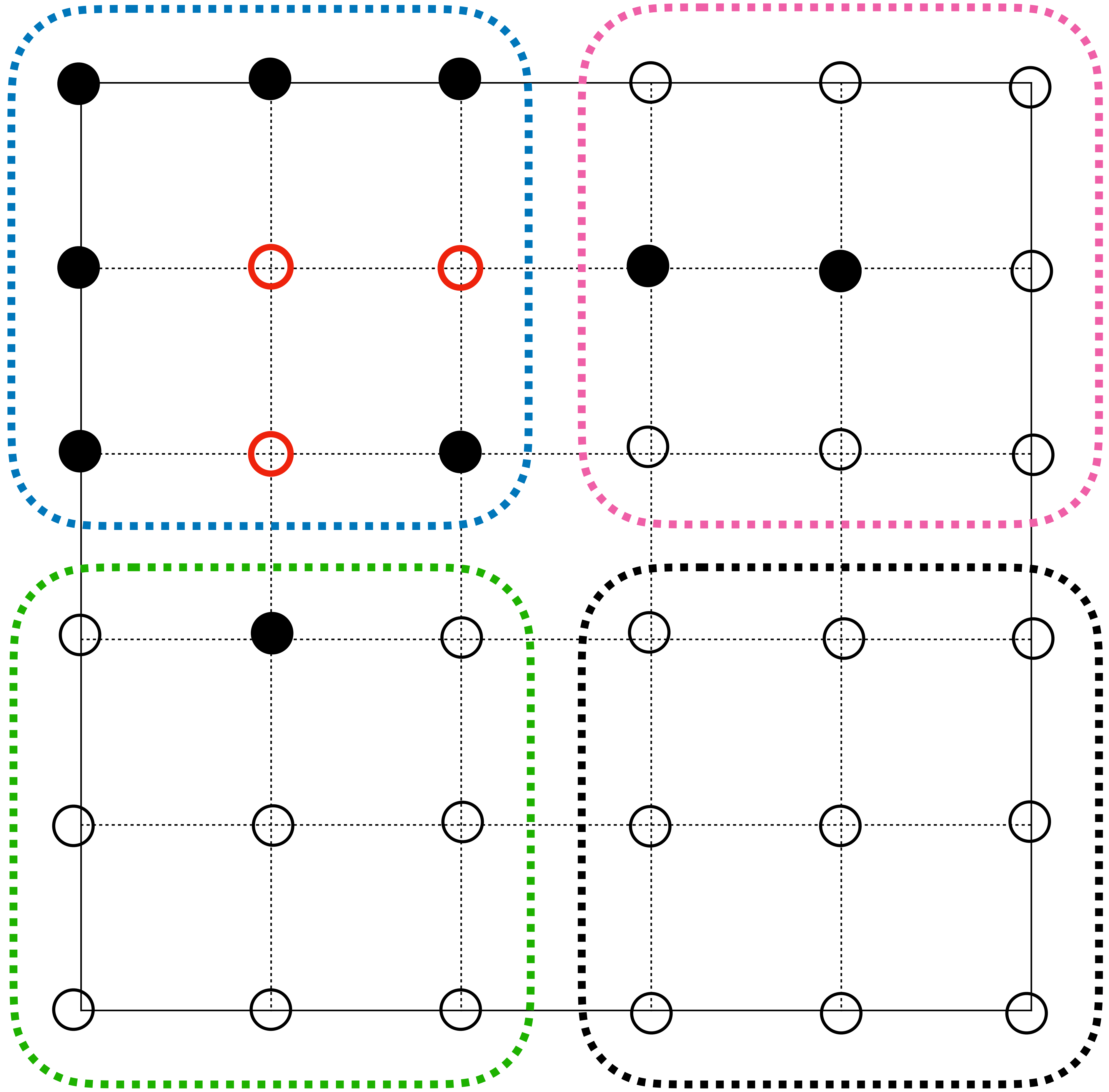
New Strategies: Use-Based Relocation



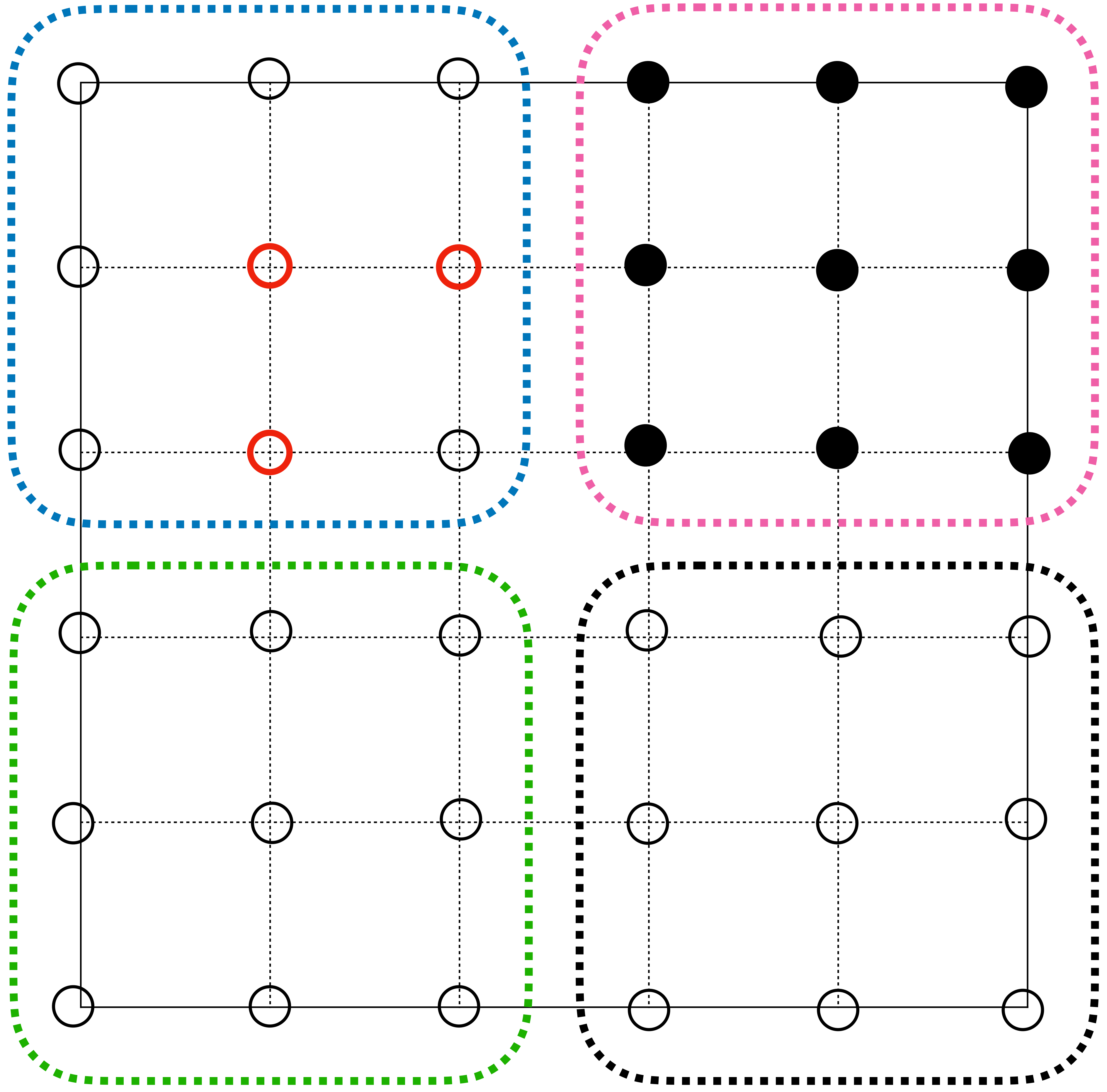
New Strategies: Use-Based Relocation



New Strategies: Use-Based Relocation



New Strategies: Use-Based Relocation



Evaluation

Architecture

- 10 by 10, 100 qubit grid of equally space qubits
- One Qubit Gate Fidelity: 99.6%, Two Qubit Gate Fidelity: 96.5%
- T1 Ground state coherence: 30s, T2 Ground State Coherence: 7s

Benchmarks

- Cuccaro Adder (Serial)
- Generalized Toffoli (Parallel)
- QAOA (See Paper)
- Linear VQE (See Paper)

Recovery Methods: Success Rate

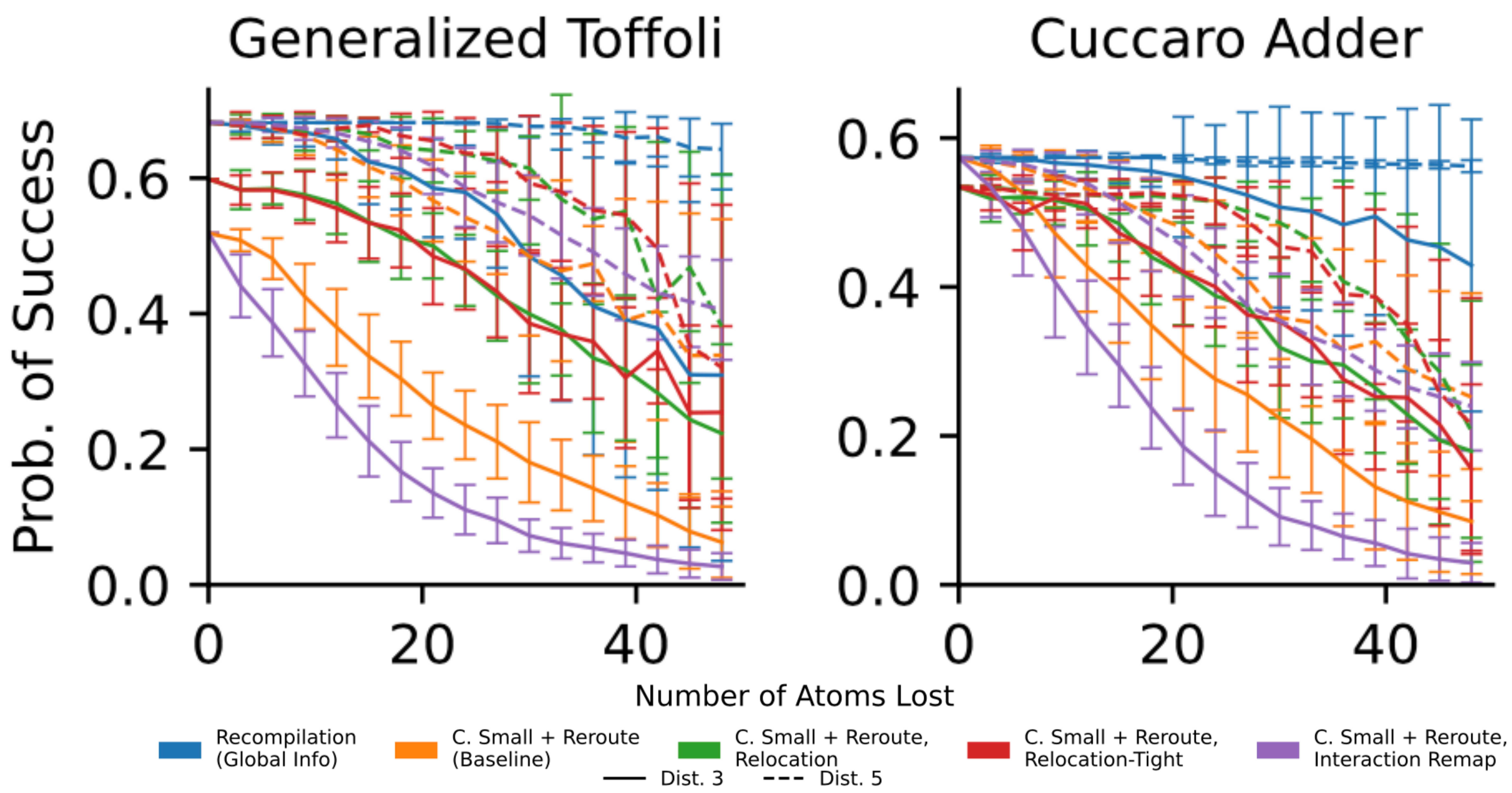
Gate Success Rate

$$S_{Gate} = \prod_{i=1}^{2} p_{gate,i}^{n_i}$$

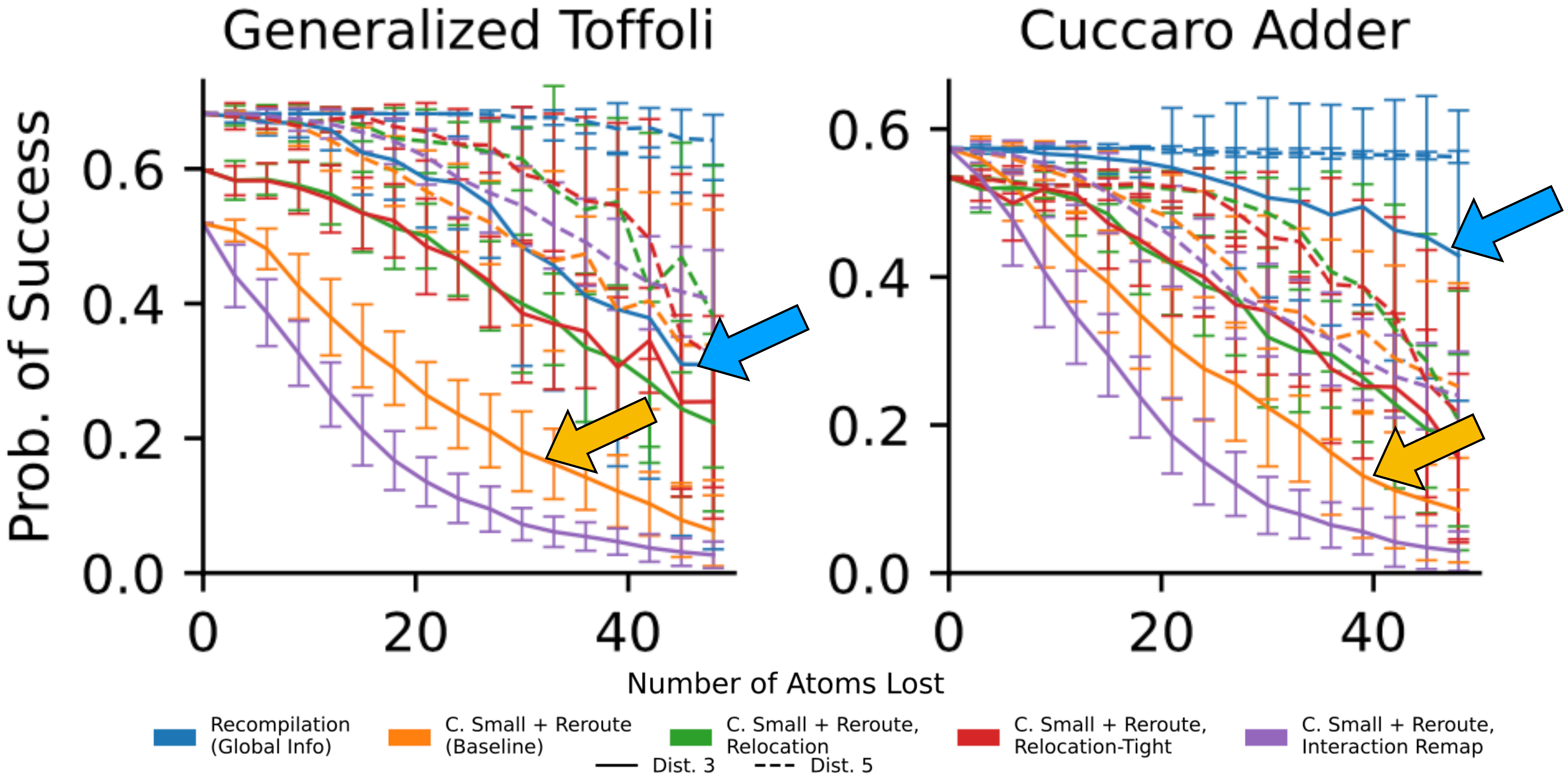
Coherence Time Success Rate

$$S_{time} = \prod_{q \in C.qubits} exp(-t_g(q)/T_{1g} - t_g(q)/T_{2g}))$$

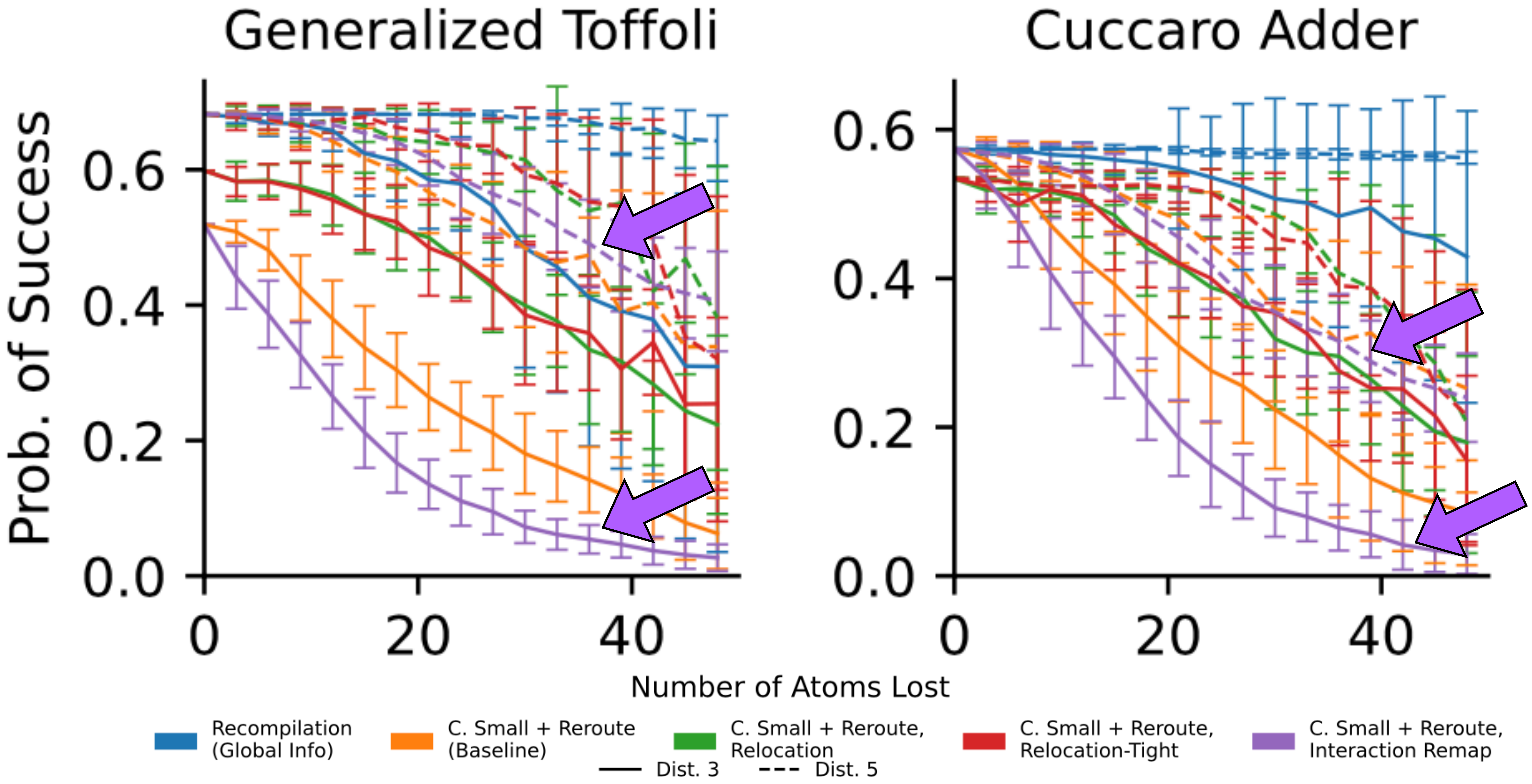
Recovery Methods: Success Rate, 30 Qubits



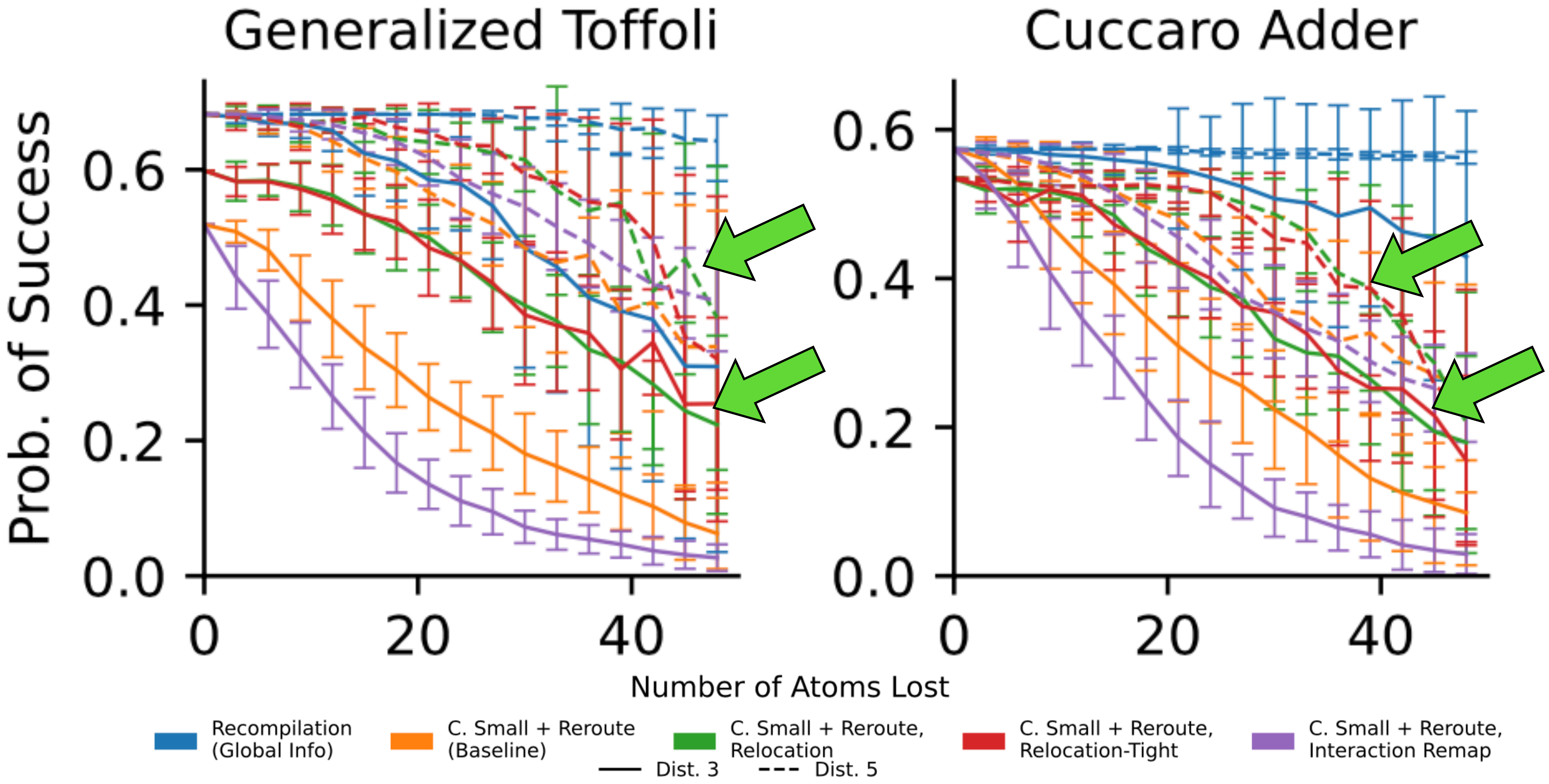
Recovery Methods: Success Rate, 30 Qubits



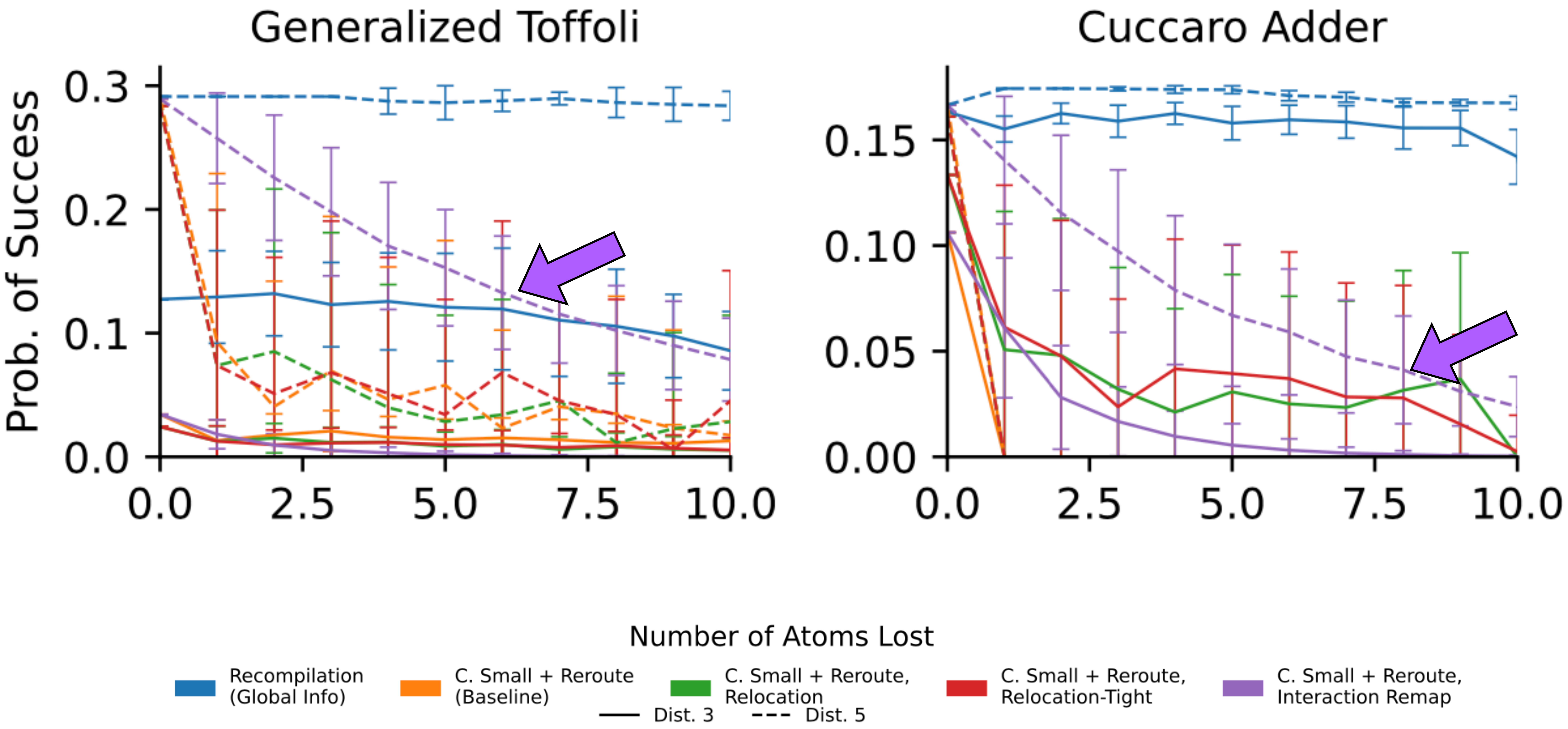
Recovery Methods: Success Rate, 30 Qubits



Recovery Methods: Success Rate, 30 Qubits

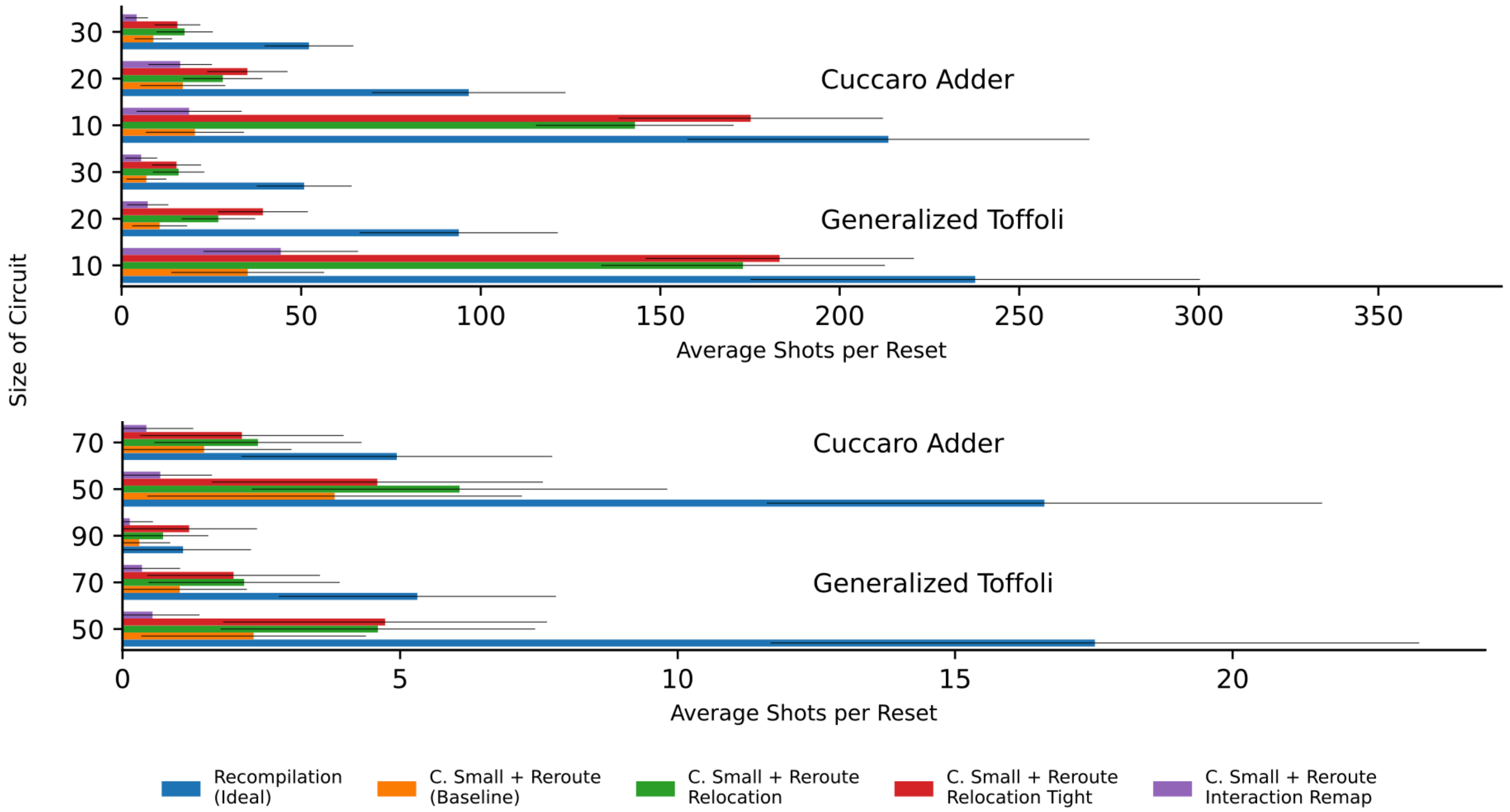


Recovery Methods: Success Rate, 90 Qubits

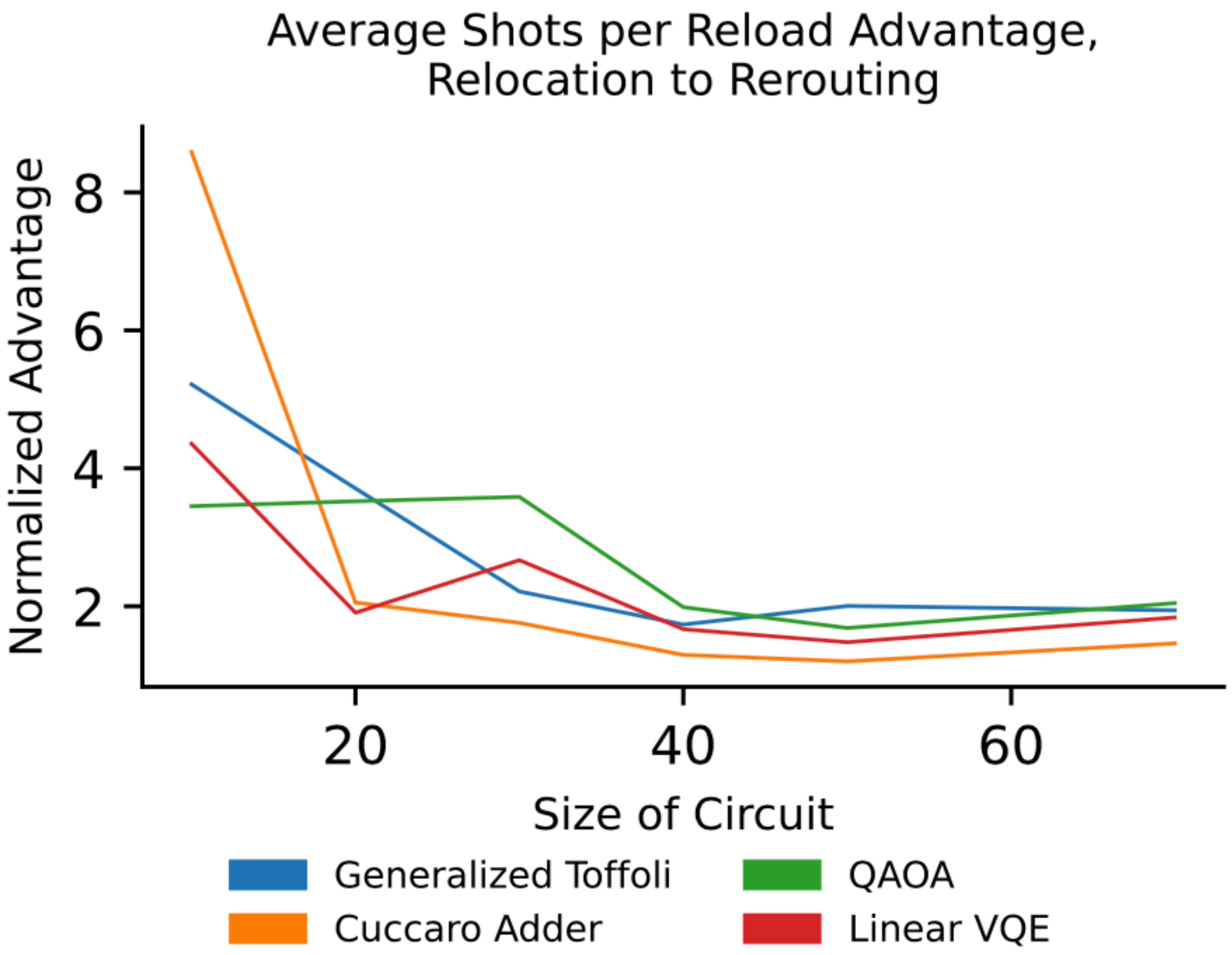


Recovery Methods: Shots per Reload

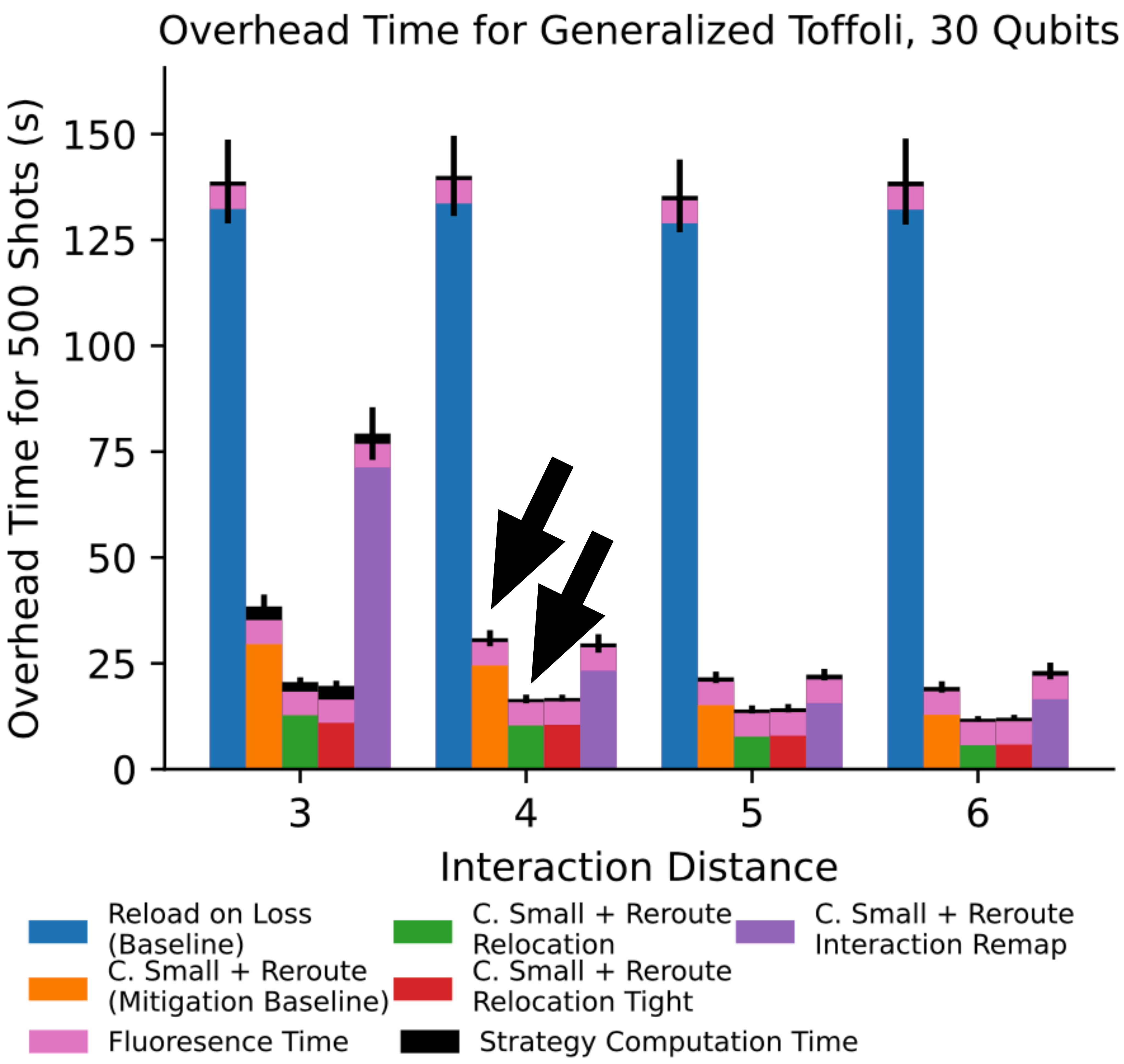
Recovery Methods: Shots per Reload



Recovery Methods: Shots per Reload

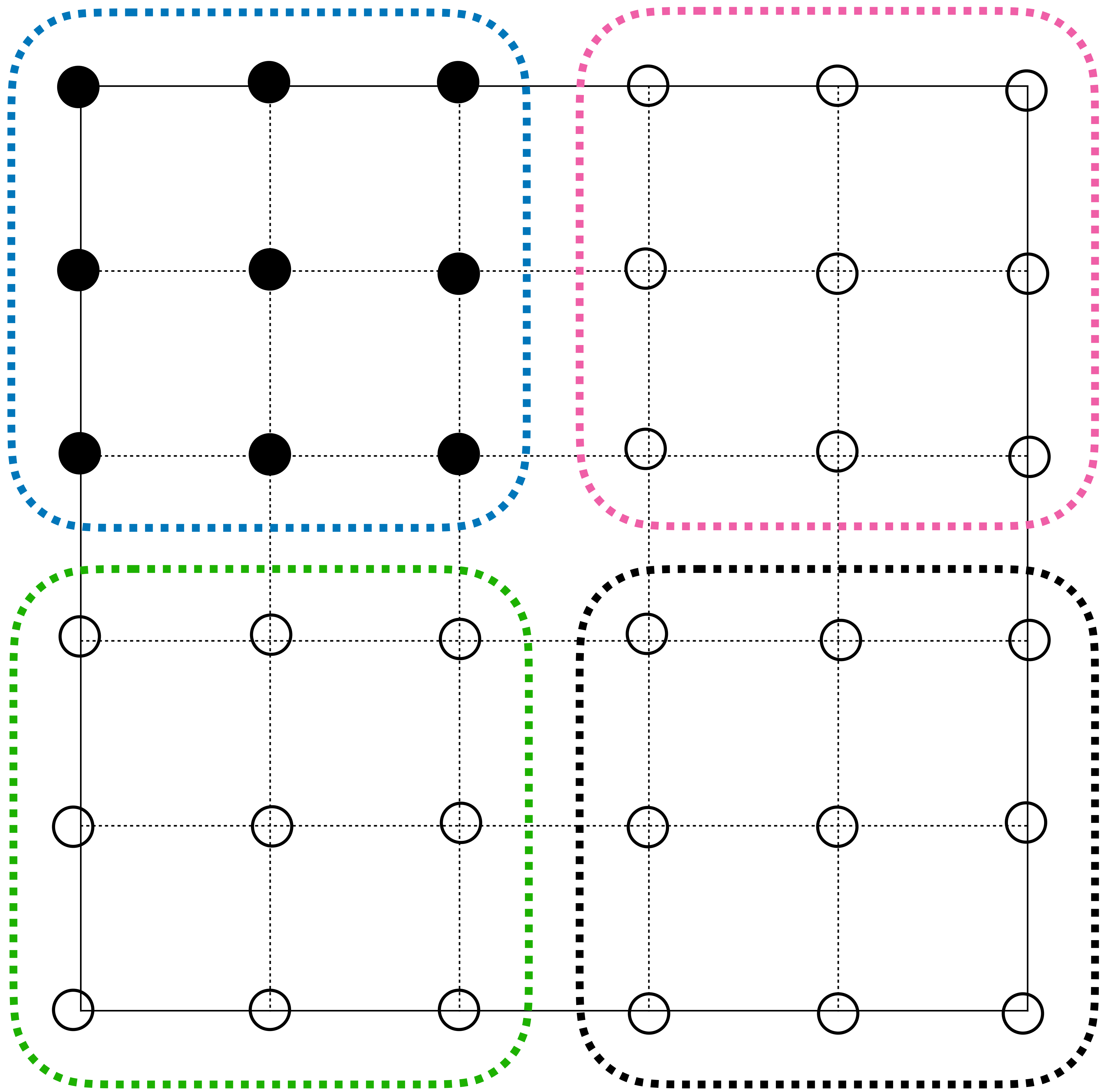


Recovery Methods: Overhead Time

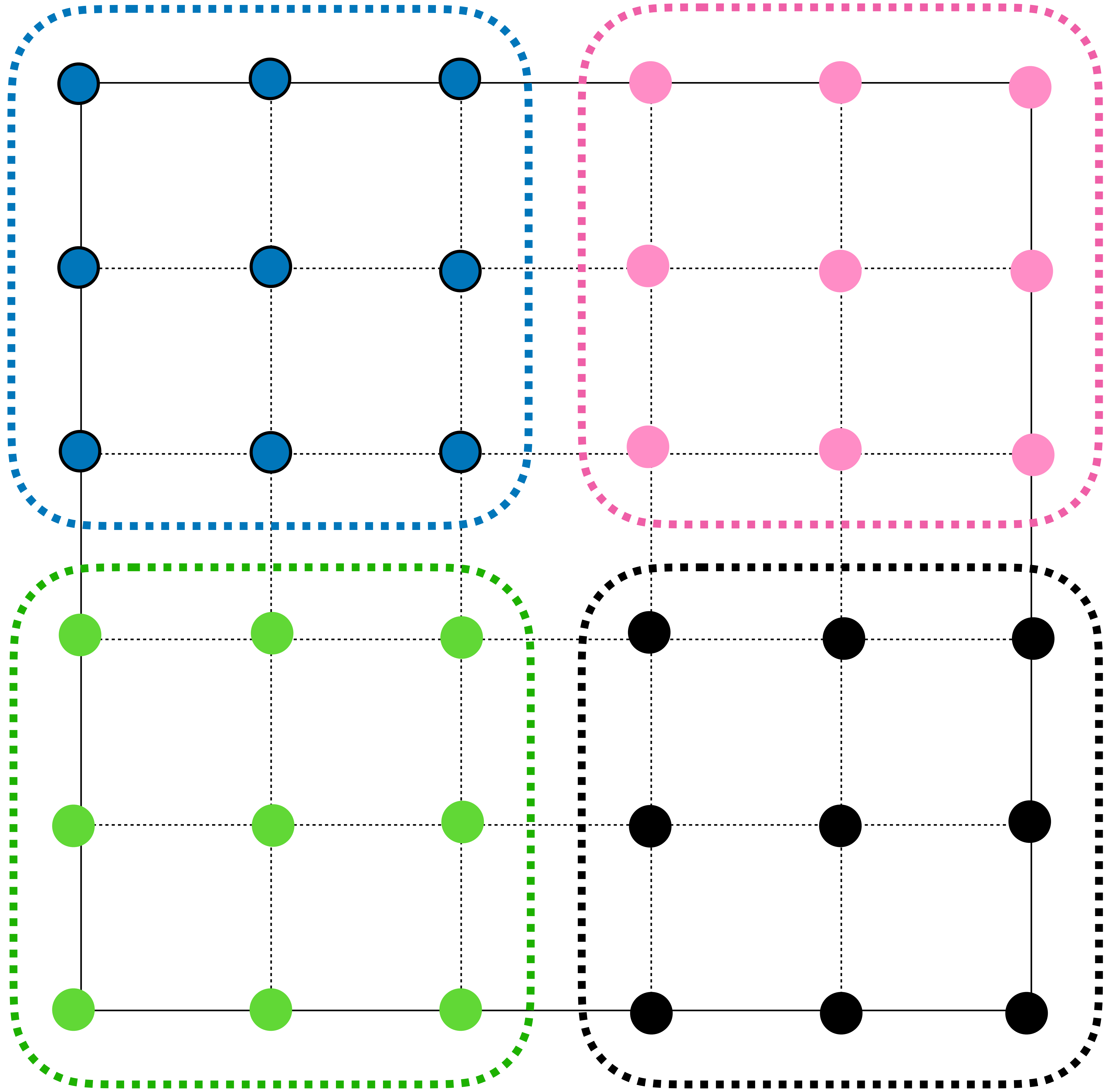


New Problem: Fluorescence

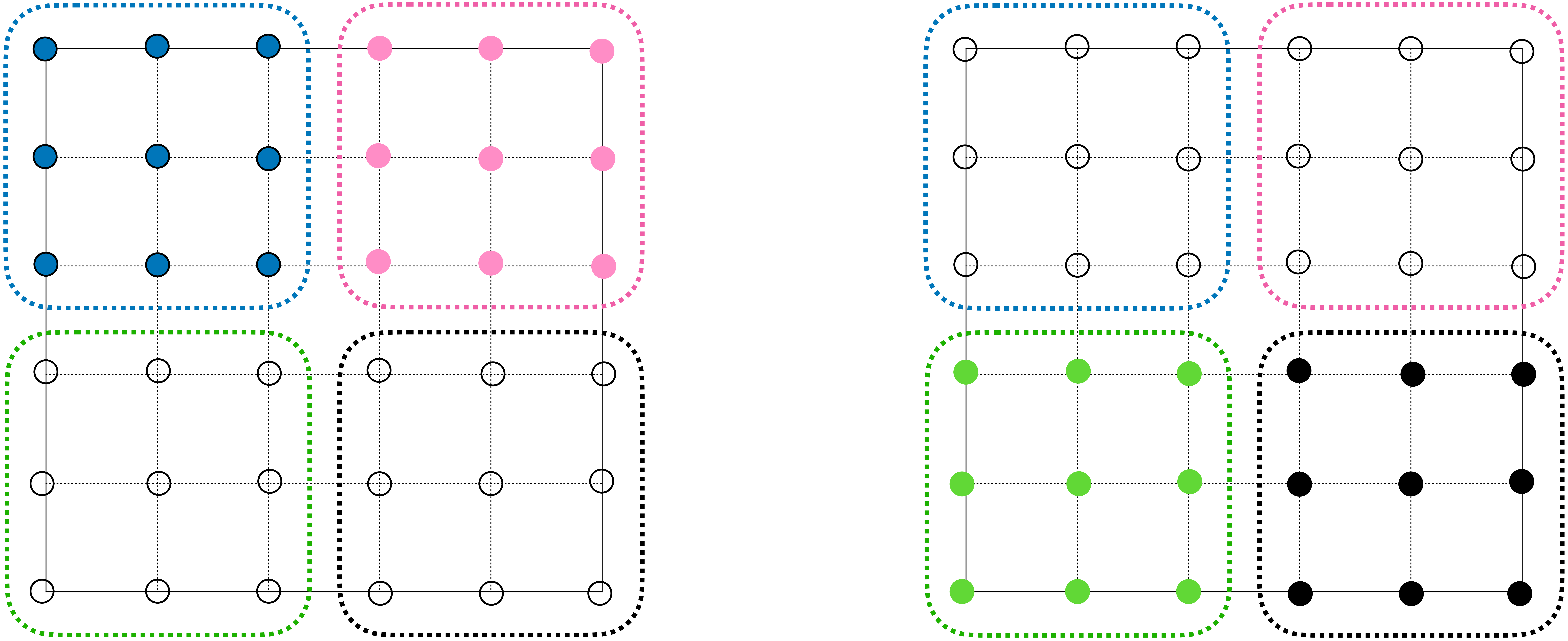
New Strategy



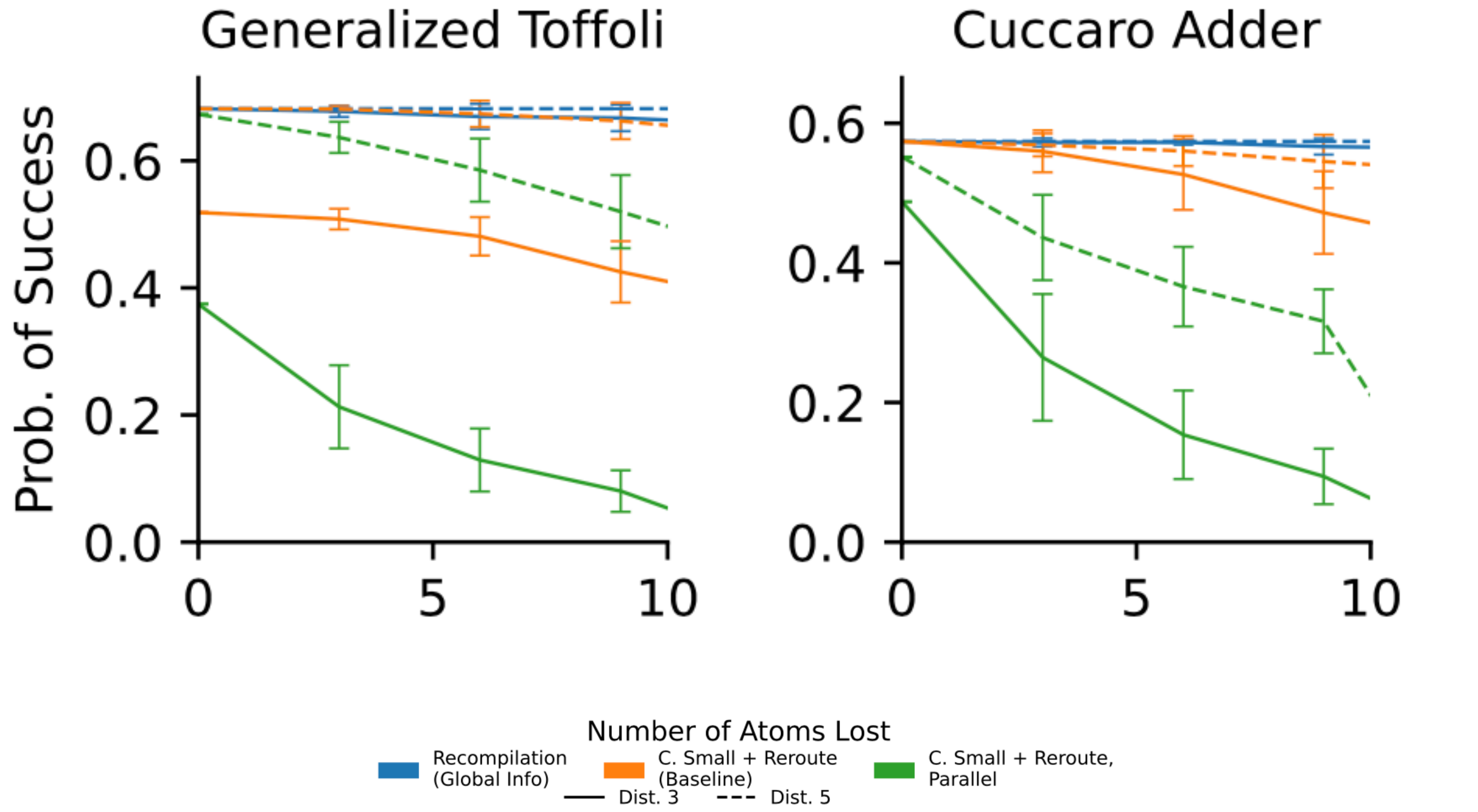
New Strategy: Parallelism



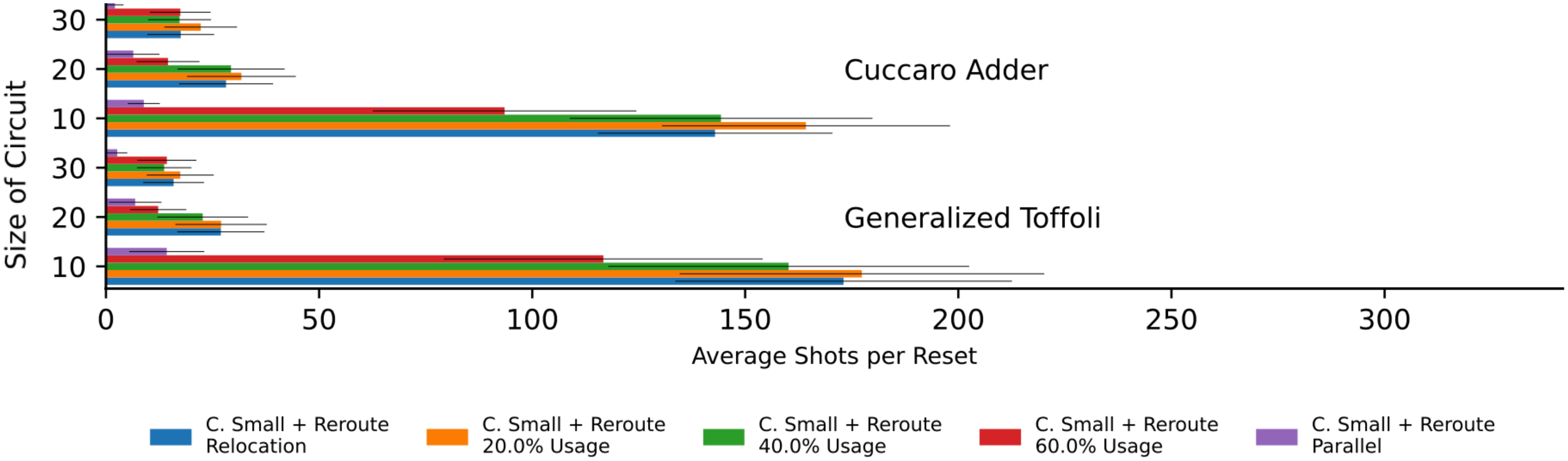
New Solution: Partial Parallelism



Parallelism and Success Rate



Parallel Recovery Methods: Shots per Reload



Partial Parallelism Can Have Best Overhead

Runtime for 500 shots

Benchmark	Baseline Time (s)	Relocation Time (s)	2 Parallel Time (s)	% Fluorescence Decrease
Cuccaro-10	10.91	4.67	3.13	51.22
Cuccaro-20	13.82	9.58	8.24	54.02
Cuccaro-30	26.19	17.05	14.80	40.71
CNU-10	7.98	4.59	2.82	50.60
CNU-20	24.09	8.68	10.50	53.12
CNU-30	30.61	16.58	15.15	35.80

Conclusions

When we need more flexibility, shifting along the interaction graph can provide necessary benefits

Relocation provides good general purpose overhead time improvements, more effectively using the entirety of the architecture.

For smaller circuits we can exploit the section defined for relocation for parallelism to even further reduce the run time through the reduction of fluorescence.