Quantum Leap: When the atom Outsmarts AI in Deciphering the Cosmos' Secrets

Susanne Yelin

Harvard University

Post-quantum AI workshop, April 1, 2024

Analog quantum machine learning

Susanne Yelin

Harvard University

Post-quantum AI workshop, April 1, 2024

Analog quantum machine learning

using present-day hardware

Susan

Harvard University

Post-quantum AI workshop, April 1, 2024

Machine learning meets quantum computing

What if ML ran on quantum hardware?

With the advent of quantum computing, one may naturally ask

Can quantum computing **enhance/ speed-up** machine learning algorithms?

Neural Networks

- Neuromorphic computing

  Brain-like architecture for computational network

- Recurrent neural networks (RNNs)

  can have cyclic connections between nodes (as opposed to feed-forward NNs)

Neural Networks

- Neuromorphic computing

  Brain-like architecture for computational network

- Recurrent neural networks (RNNs)

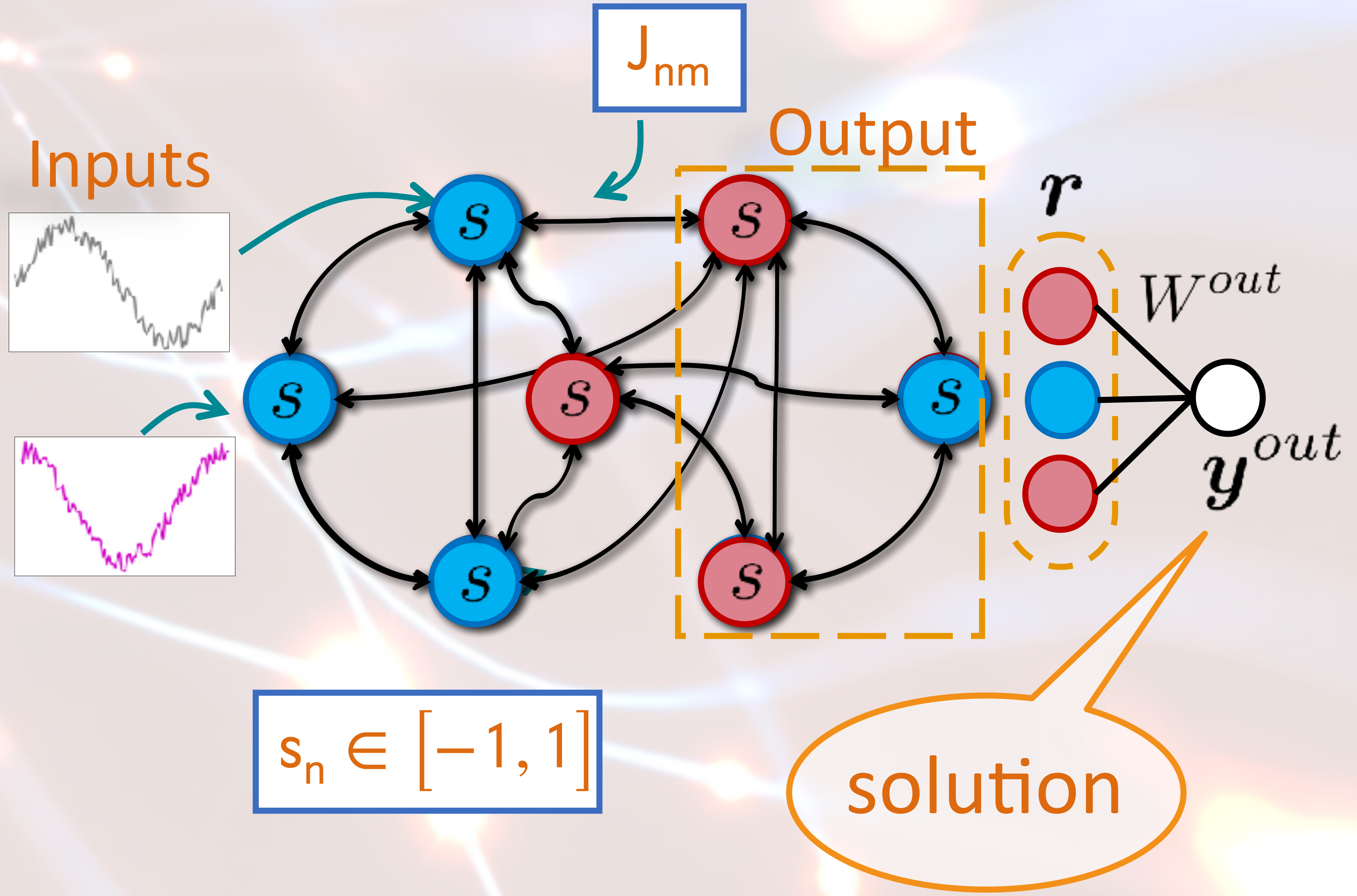
  can have cyclic connections between nodes (as opposed to feed-forward NNs)

- Reservoir computing

  special case of RNN with fixed connections

Example: Reservoir computing (RC)

- Neural networks have trainable connections J_{nm} and output filters W^{out}.



- However, training J_{nm} is resource intensive.

- Reservoir computers assume fixed connections $J_{nm} = J$ or $J_{nm} = 0$, randomly

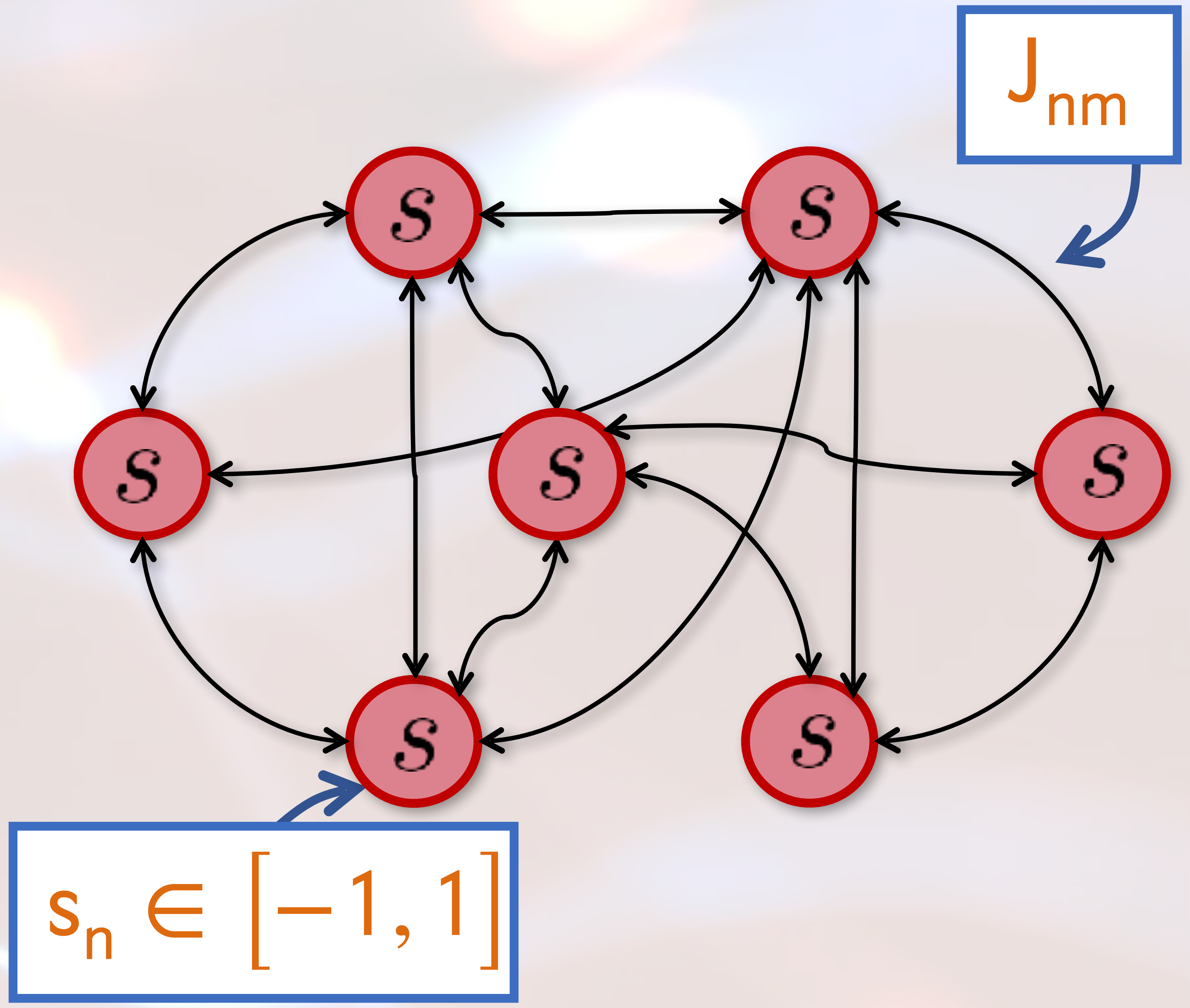
Training an RC: Update rule

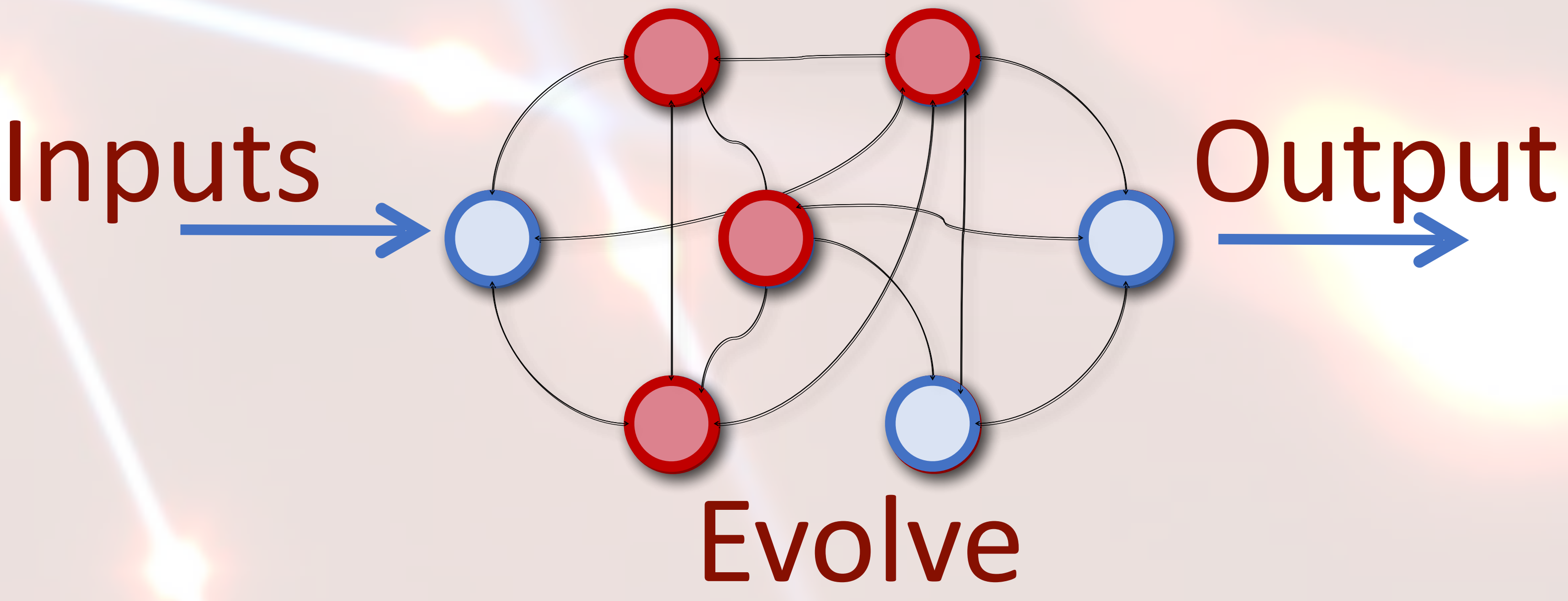
We think of neurons as discrete and interacting classical spins, subject to effective field

$$h_n(t) = \sum_m J_{nm} s_m(t) - \Delta_n(t)$$

Synaptic weights: Brain's structure

Biases: External stimuli





1. If $h_n(t) > 0$ keep $s_n(t)$
2. If $h_n(t) < 0$ flip $s_n(t)$

Training cycle

Goal: Optimize set of parameters (e.g., all J_{nm} and W^{out}) to give desired output

Update:

Let system evolve under given set of parameters, then measure

Optimize:

Calculate "loss function" and find new set of parameters

Training cycle

Goal: Optimize set of parameters (e.g., all J_{nm} and W^{out}) to give desired output

e.g., overlap with desired outcome with **all** training data sets

Update:

Let system evolve under given set of parameters, then measure

Optimize:

Calculate "loss function" and find new set of parameters

e.g., via gradient descent

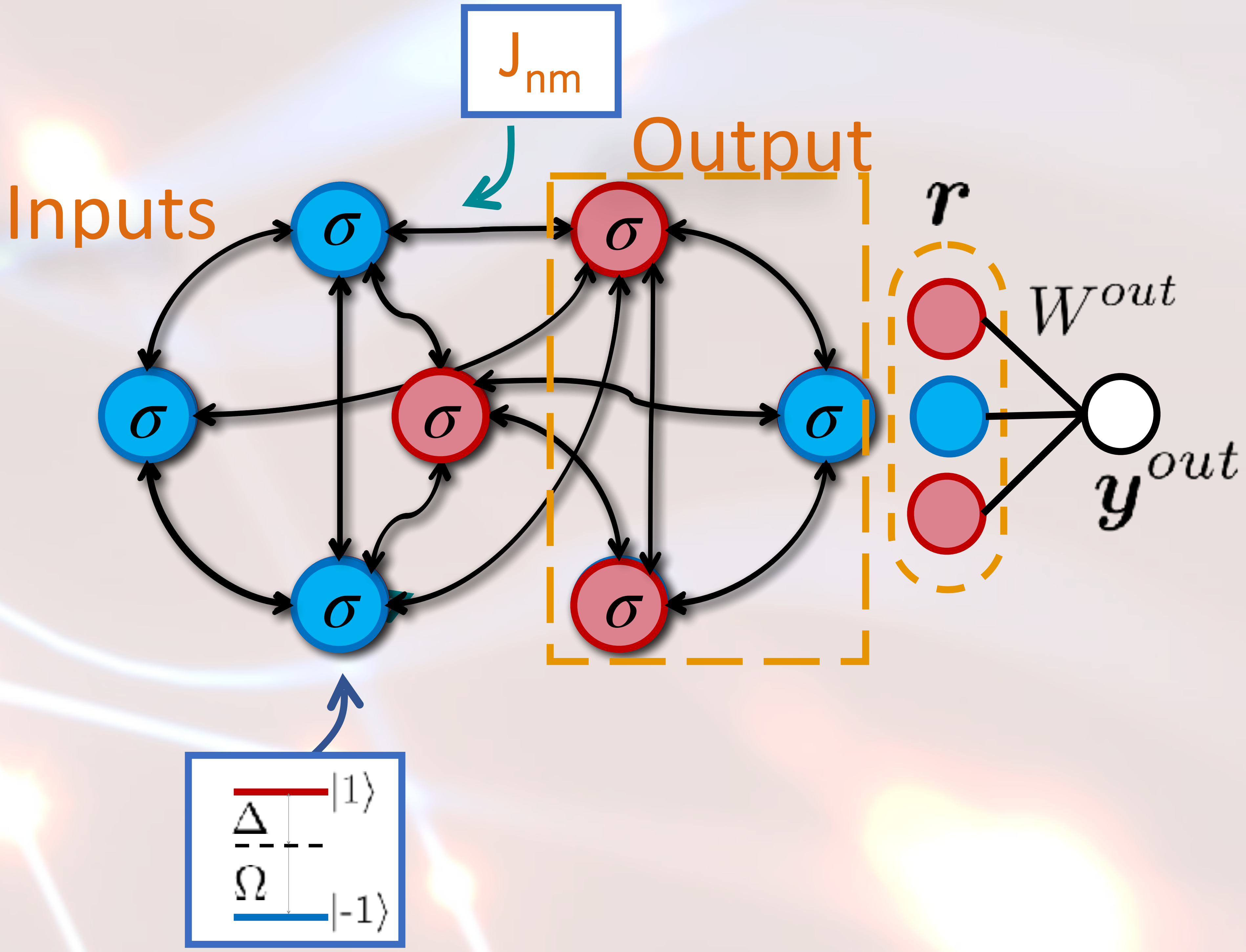
Quantizing a reservoir computer (RNN)

Each qubit evolves under the Hamiltonian

$$H = \sum_n \left(\hat{h}_n(t)\, \sigma_n^z + \frac{\Omega(t)}{2}\, \sigma_n^x \right)$$

$$\boxed{\hat{h}_n(t) = \sum_m J_{nm} s_m(t) - \Delta_n(t)}$$

facilitates flips



Update rule:

evolve for $\Omega\tau = \pi$

If $\hat{h}_n(t) \gg \Omega$: σ_n^z doesn't change

If $\hat{h}_n(t) \ll \Omega$: σ_n^z flips

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

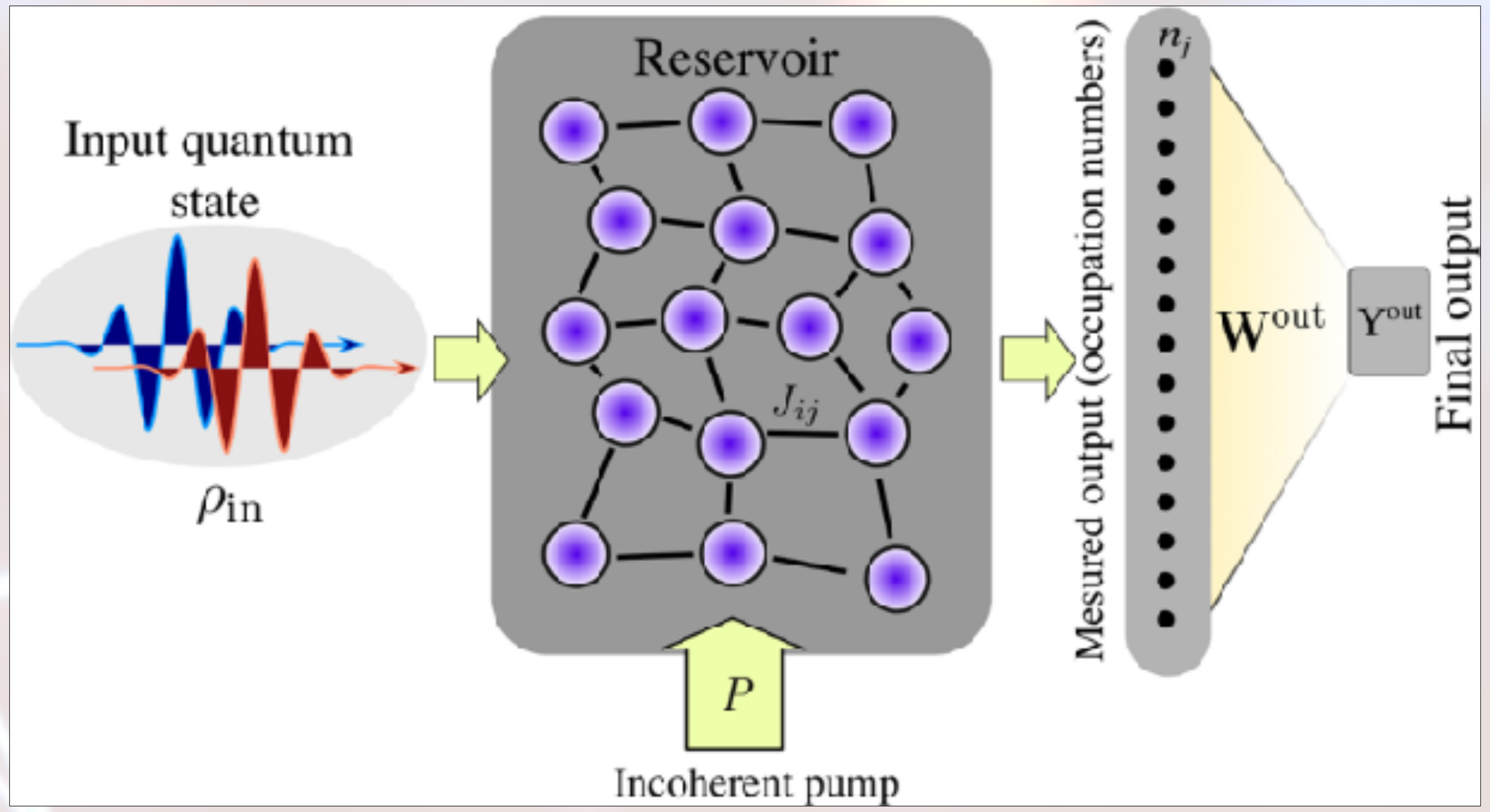
Quantum reservoir computing

Earlier work

Applications:

- Entanglement detection[1]
- Time-series prediction[2]
- Long-term memory[3]

and many more…



Requires large-scale universal quantum computation

Our goal: analog or analog/ digital hybrid near-term devices

[1]Ghosh, S., et. al., Scientific Reports (2019),
[2]Suzuki, Y., et. al., Scientific Reports (2022),
[3]Martínez-Peña, et. al., PRL (2021)

Training cycle of **quantum** RC

Goal: Optimize set of parameters (e.g., all J_{nm} and W^{out}) to give desired output

Update:
Let system evolve under given set of parameters, then measure

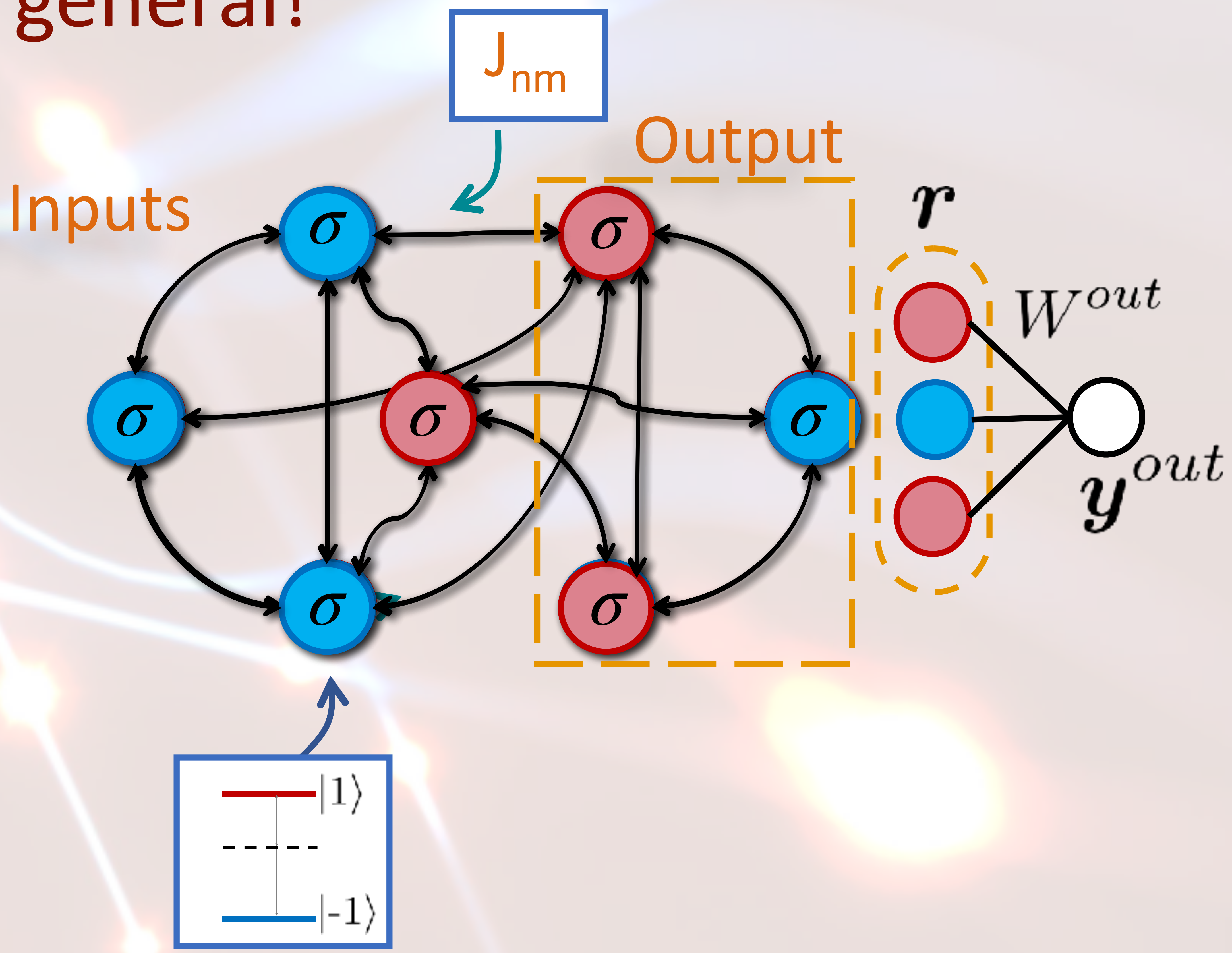
Optimize:
Calculate "loss function" and find new set of parameters

typically classical

Quantizing a reservoir computer (RNN)

We regain the update rule
… the Hamiltonian evolution is far more general!

$$\hat{h}_n(t)\,\sigma_n^z + \frac{\Omega(t)}{2}\,\sigma_n^x$$



New "quantum features" can be used for novel computation:

1. Interference/freedom of measurement basis can be used for error detection / correction
2. Arbitrary (measurement) basis can produce training speedups relative to classical RNNs
3. Efficient stochastic processes

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Quantizing an RNN

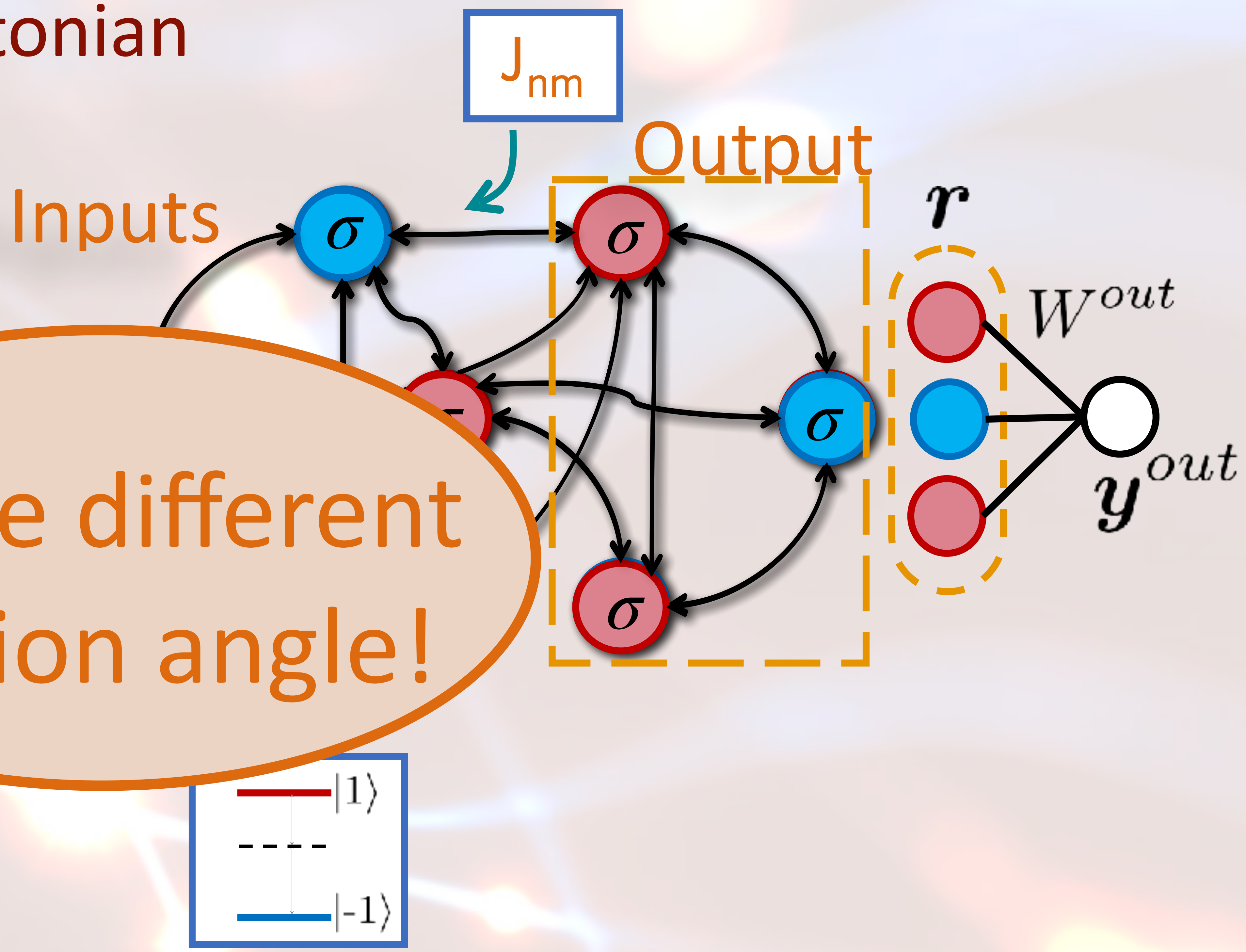
Each qubit evolves under the Hamiltonian

$$\hat{h}_n(t)\,\sigma_n^z + \frac{\Omega(t)}{2}\,\sigma_n^x$$

$\boxed{\hat{h}_n(t) = \sum_m J_{nm} s_m(t) - \Delta_n(t)}$

Fa

Can be different rotation angle!



Update rule:

evolve for $\Omega\tau = \pi$

If $\hat{h}_n(t) \gg \Omega$ keep σ_n^z

If $\hat{h}_n(t) \ll \Omega$ flip σ_n^z

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Feature #1: interference

σ_1 Parity Computer σ_2

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Feature #1: interference

Parity Computer

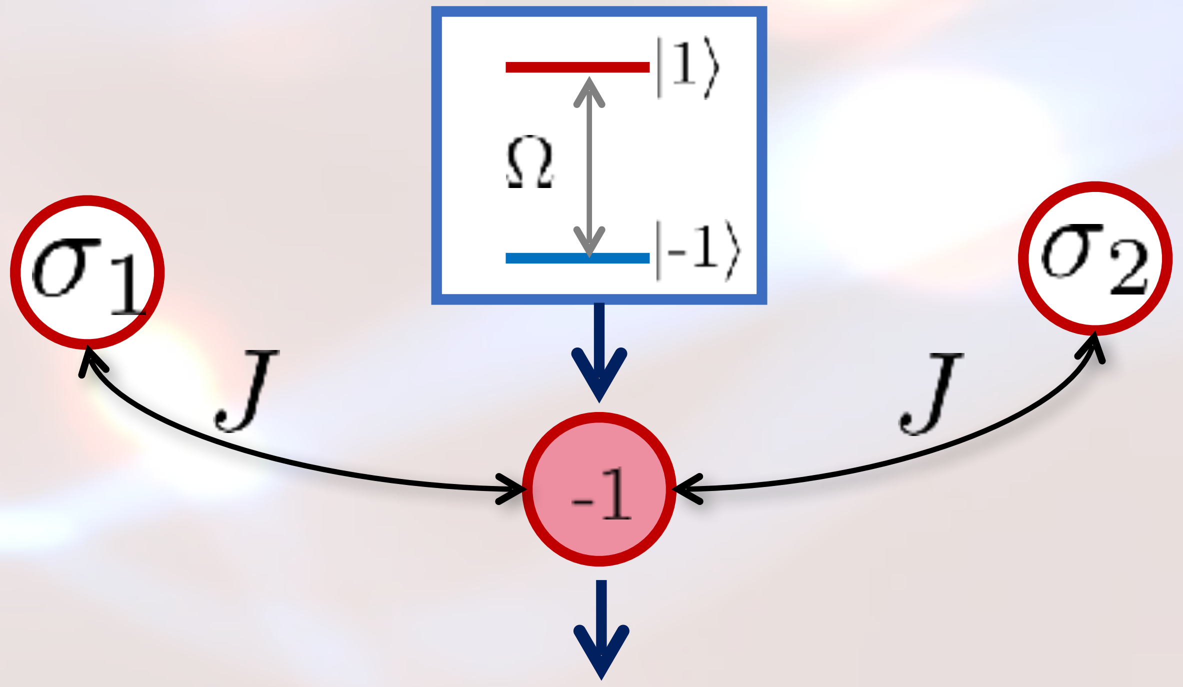
$\sigma_1 \sigma_2$

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Feature #1: interference

Parity Computer

$\sigma_1\sigma_2$

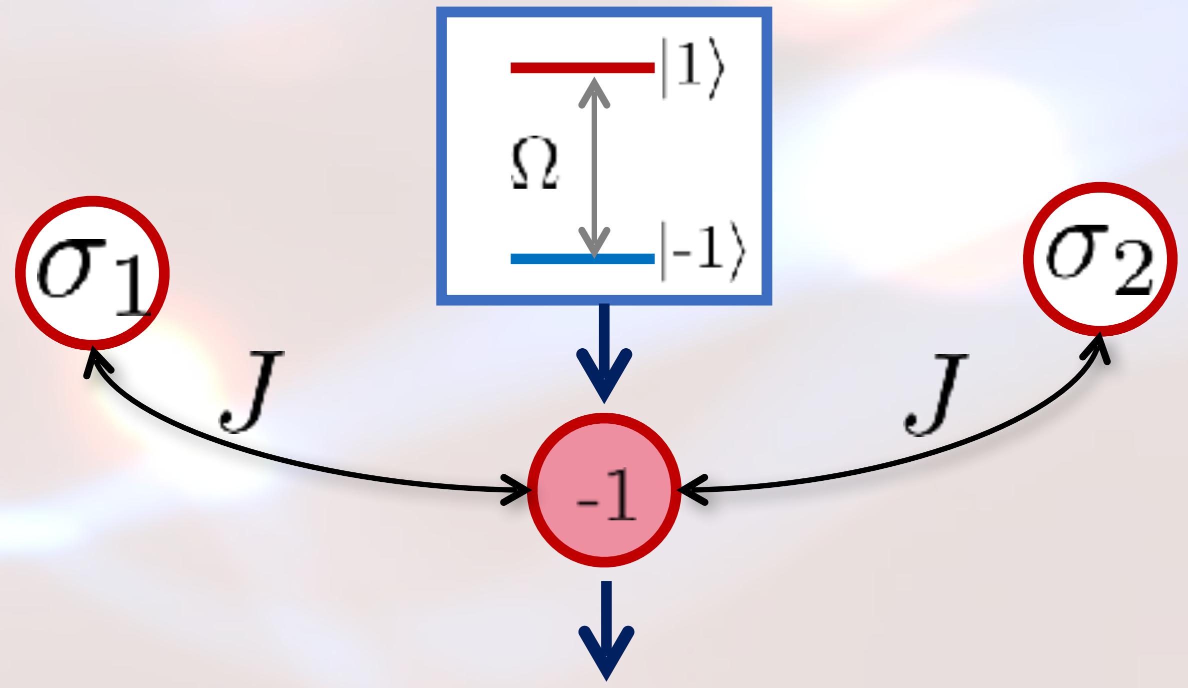


Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Feature #1: interference

Parity Computer

$\sigma_1 \sigma_2$



$$h_{\text{output}} = J(\sigma_1 + \sigma_2)$$

$$\sigma_1 = \sigma_2 \longrightarrow h_{\text{output}} \gg \Omega$$

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

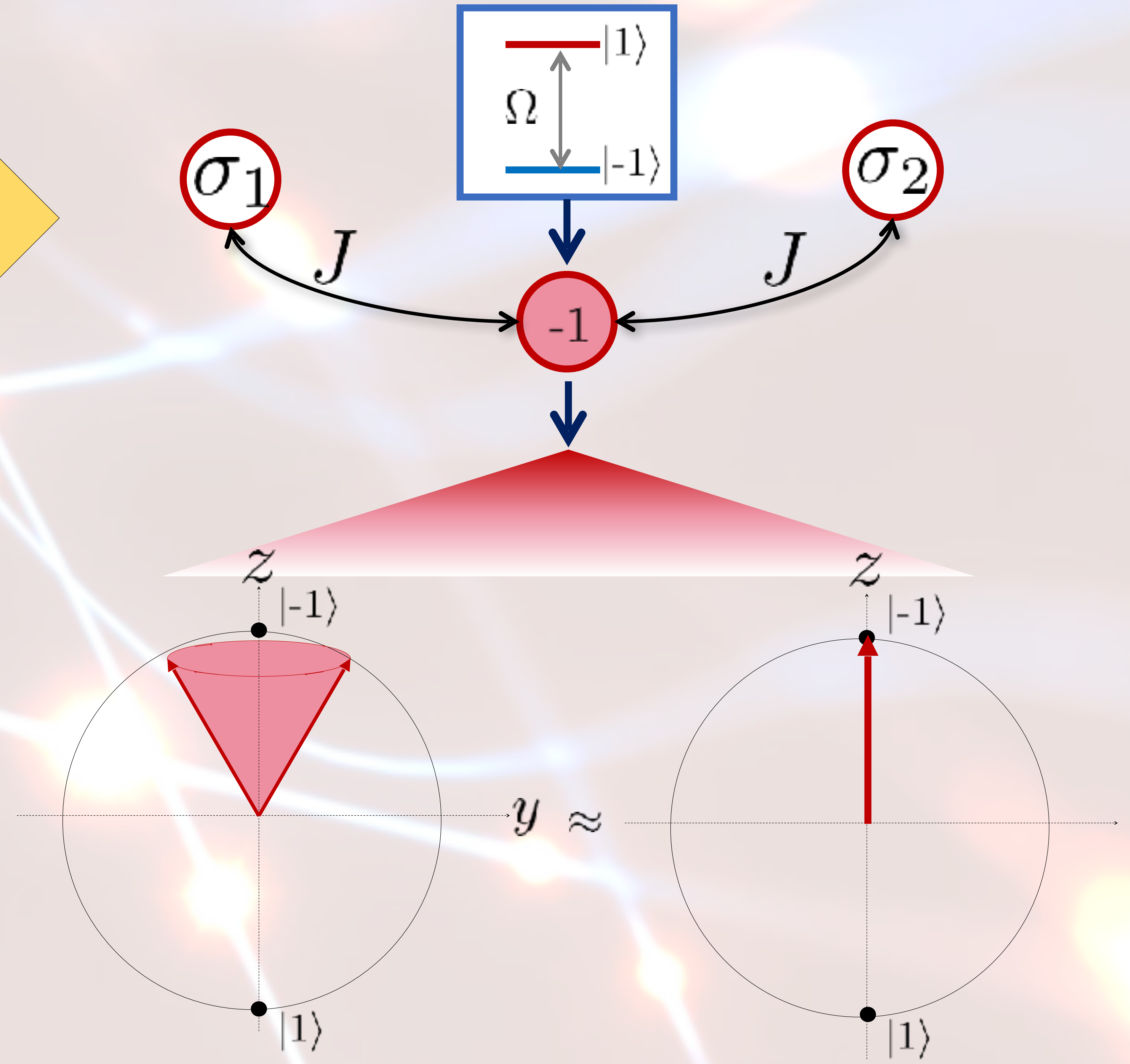
Feature #1: interference

Parity Computer

$\sigma_1 \sigma_2$

$$h_{\text{output}} = J(\sigma_1 + \sigma_2)$$

$$\sigma_1 = \sigma_2 \longrightarrow h_{\text{output}} \gg \Omega$$

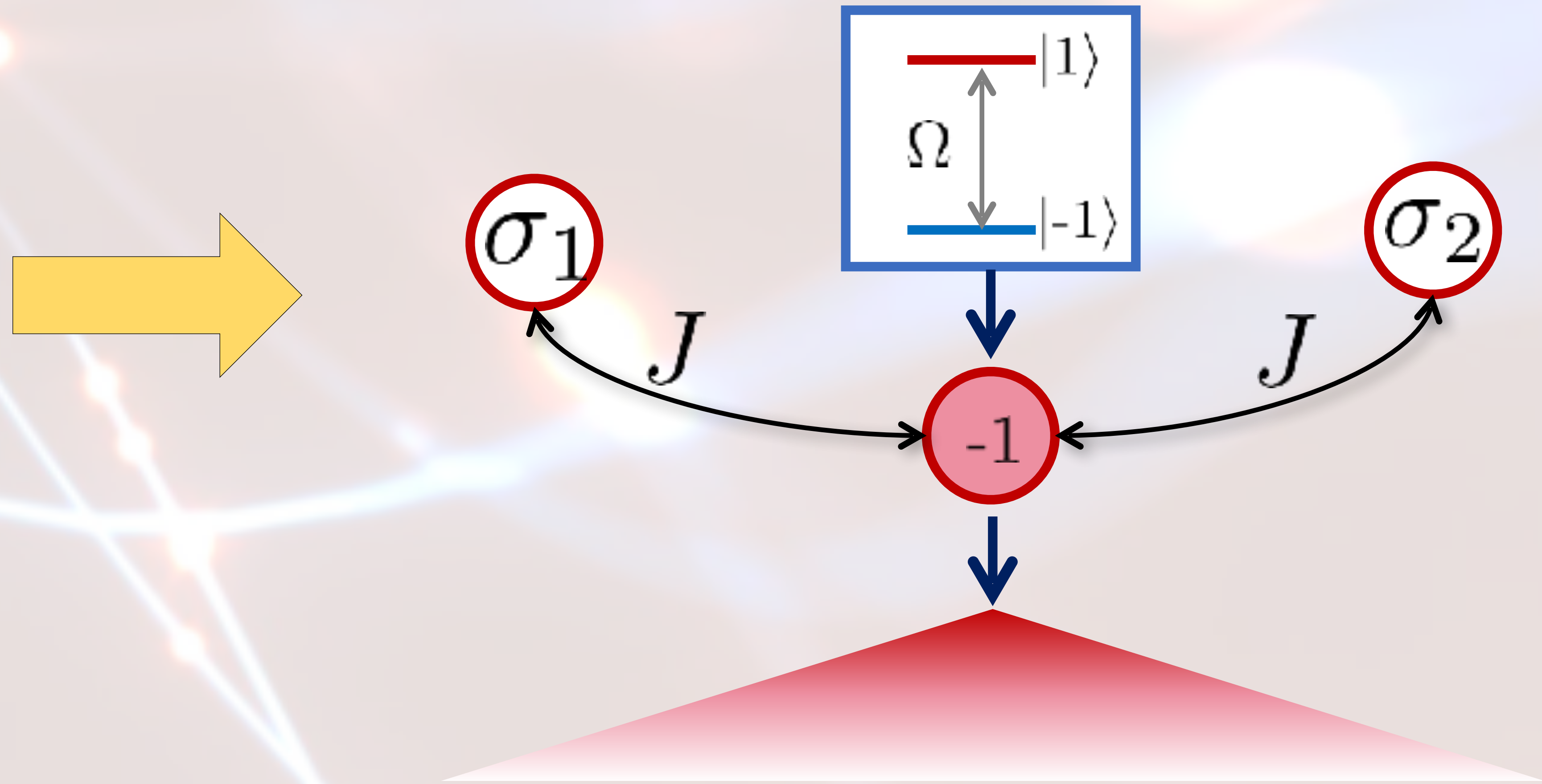


Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Feature #1: interference

Parity Computer

$\sigma_1 \sigma_2$



$$h_{\text{output}} = J(\sigma_1 + \sigma_2)$$

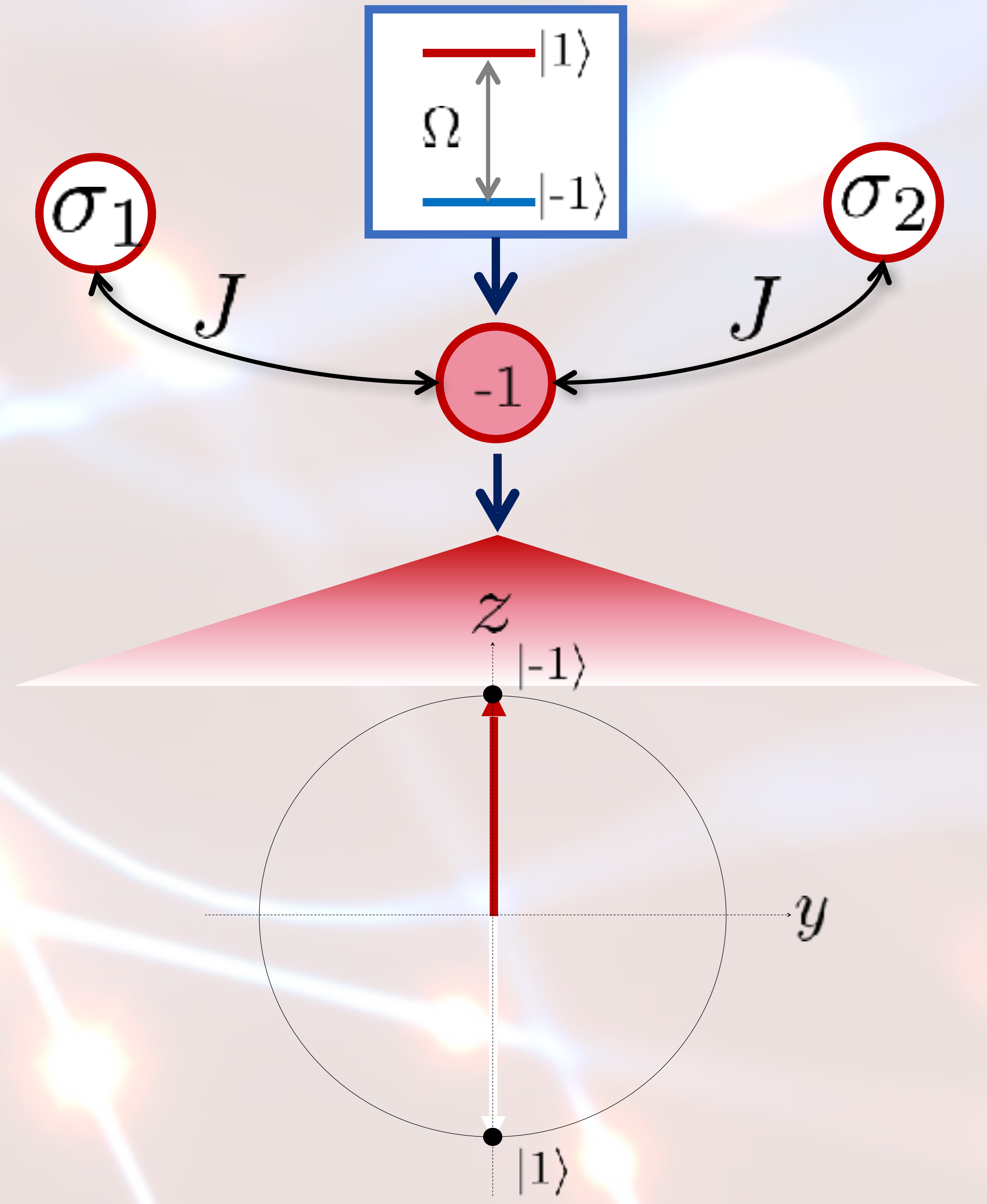
$$\sigma_1 \neq \sigma_2 \longrightarrow h_{\text{output}} \ll \Omega$$

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Feature #1: interference

Parity Computer

$\sigma_1\sigma_2$



$$h_{\text{output}} = J(\sigma_1 + \sigma_2)$$

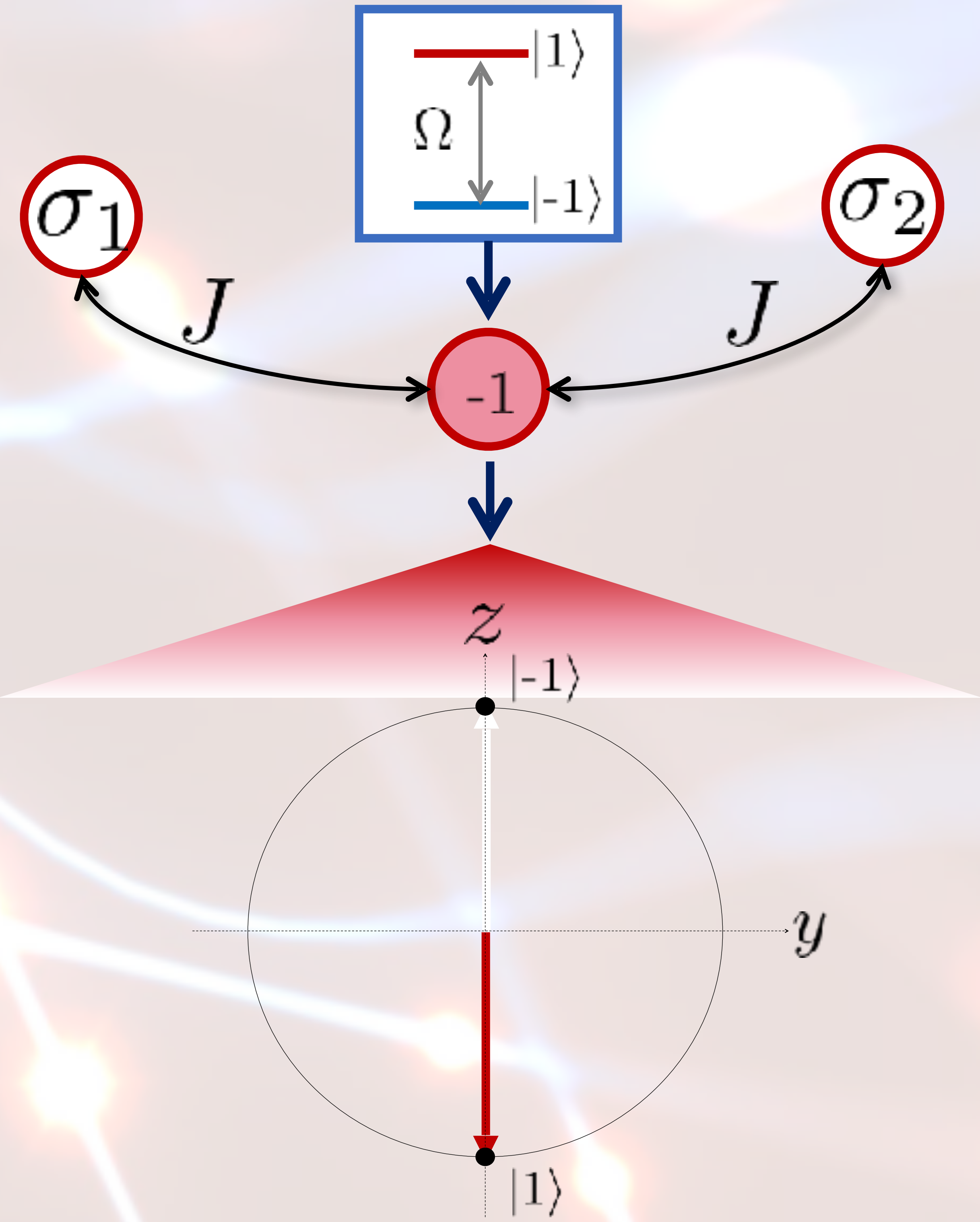
$$\sigma_1 \neq \sigma_2 \longrightarrow h_{\text{output}} \ll \Omega$$

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Feature #1: interference

Parity Computer

$\sigma_1\sigma_2$

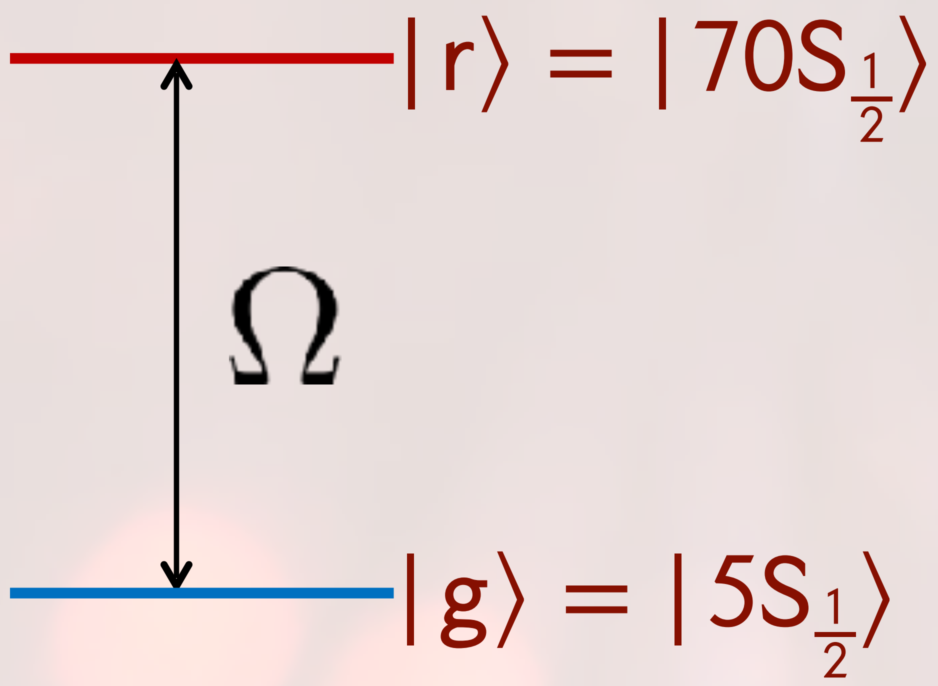


$$h_{\text{output}} = J(\sigma_1 + \sigma_2)$$

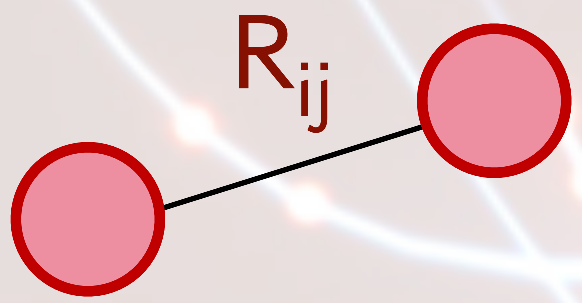
$$\sigma_1 \neq \sigma_2 \longrightarrow h_{\text{output}} \ll \Omega$$

Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

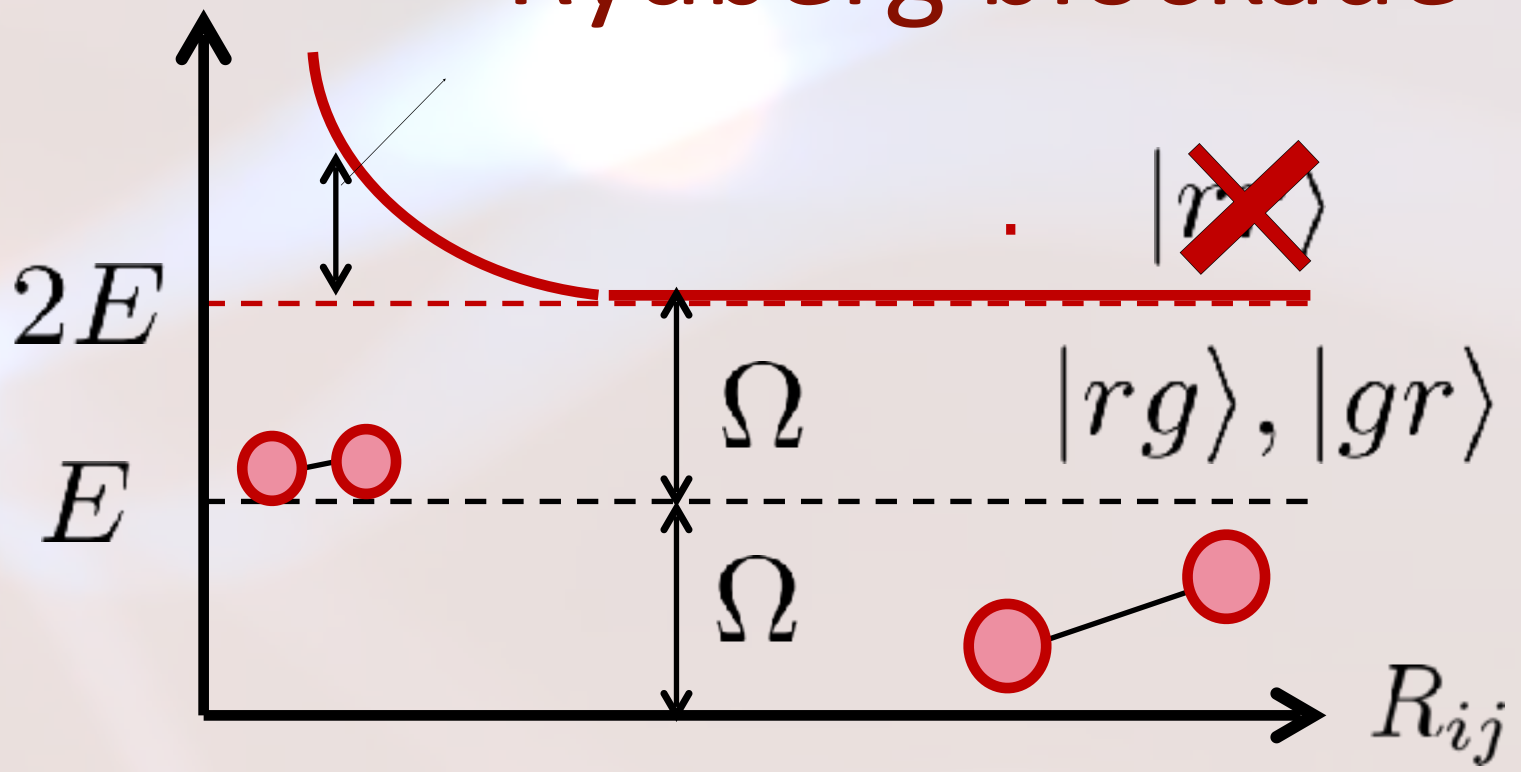
Example Implementation: Rydberg Arrays



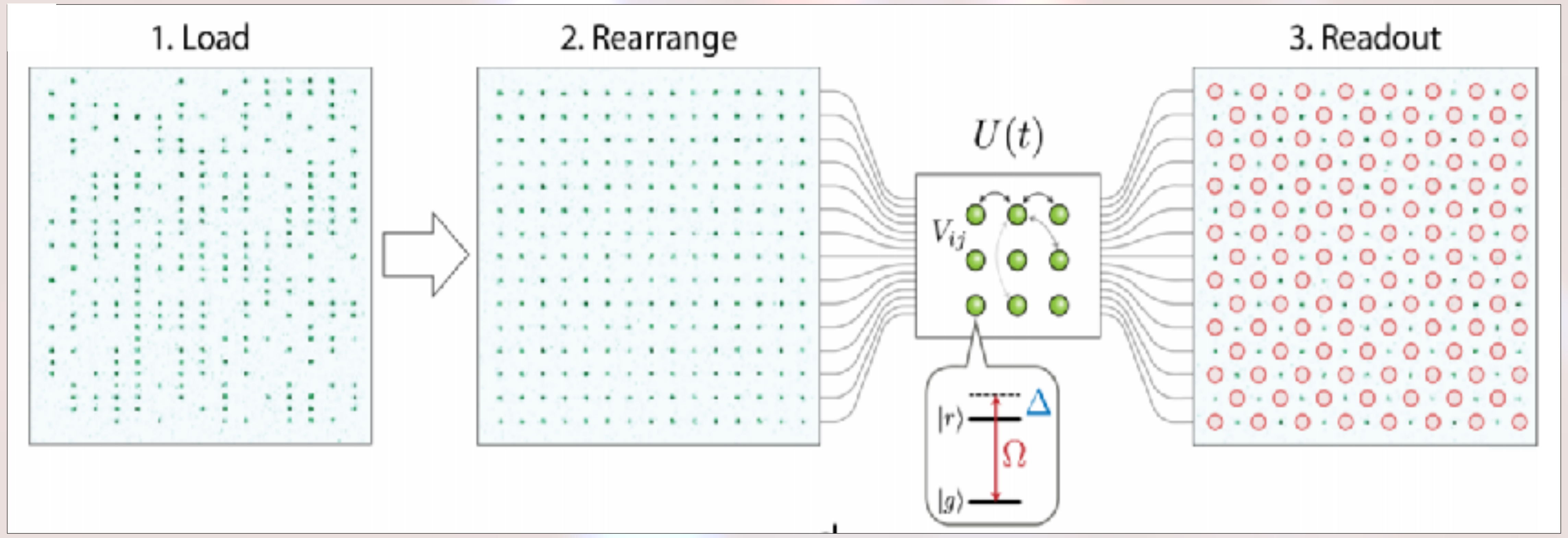
Van der Waals interaction



Rydberg blockade



Programmable arrays of Rydberg atoms using optical tweezers[1] can implement qRCs

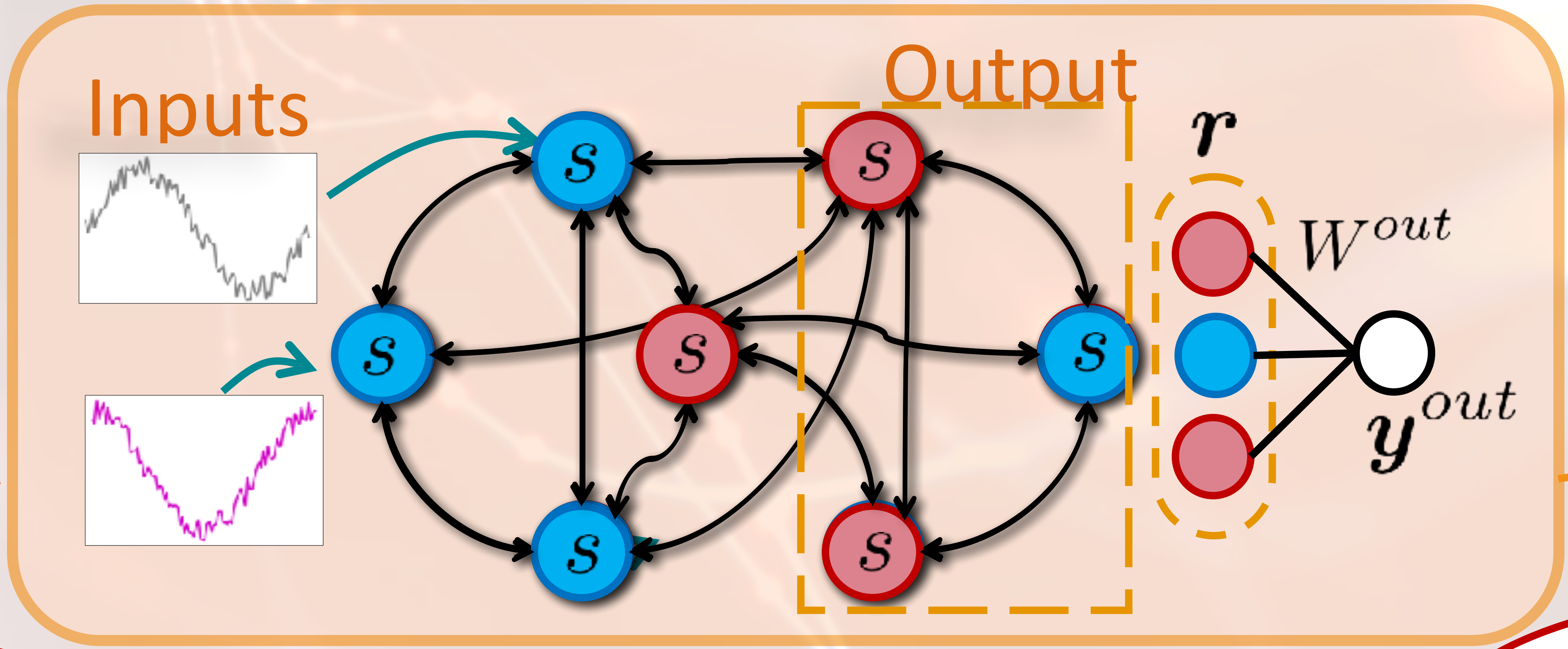


Arrays of Rydberg atoms behave like qRCs:

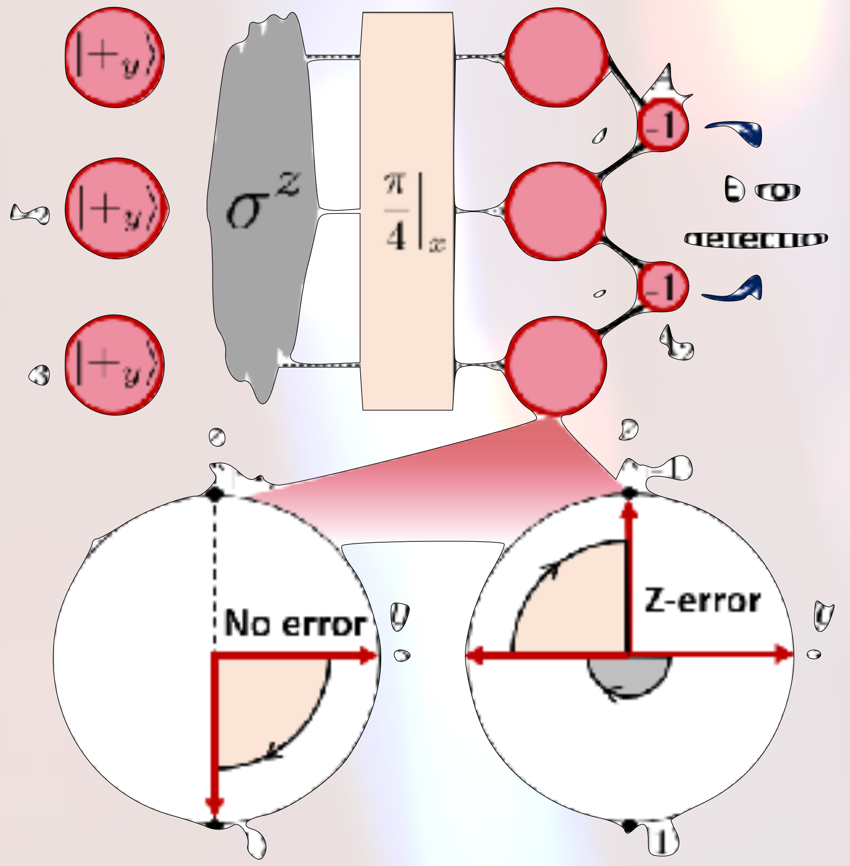
$$H = -\sum_n \Delta_n \sigma_n^z + \sum_{n<m} \frac{C_6}{R_{nm}^6} \sigma_m^z \sigma_n^z + \frac{\Omega}{2} \sum_n \sigma_n^x$$

[1]Ebadi, S., et al., Nature (2020)

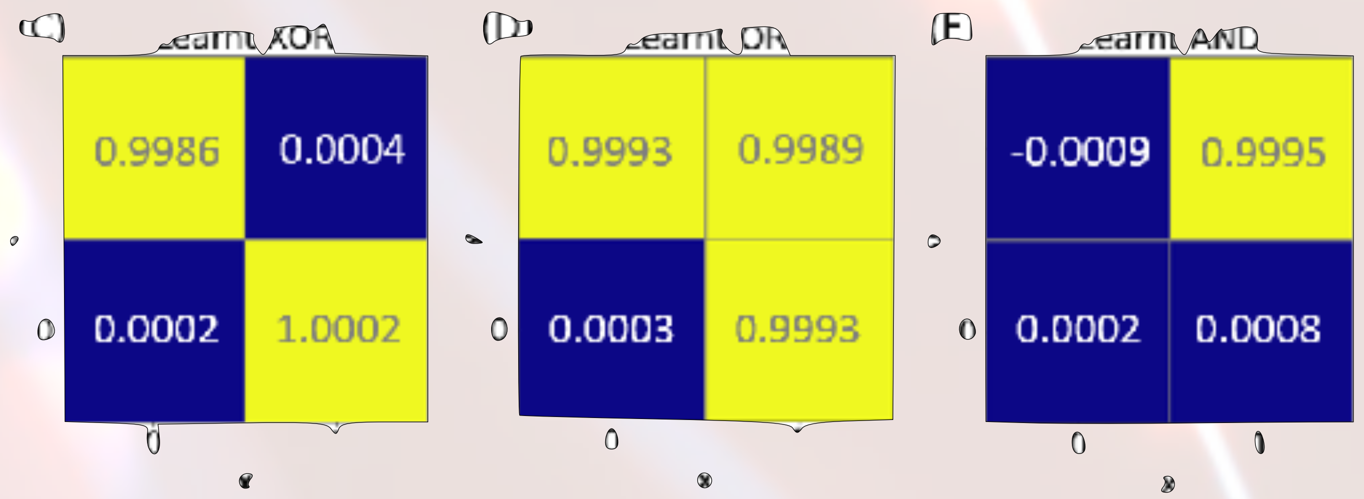
Examples: Applications for a qRC



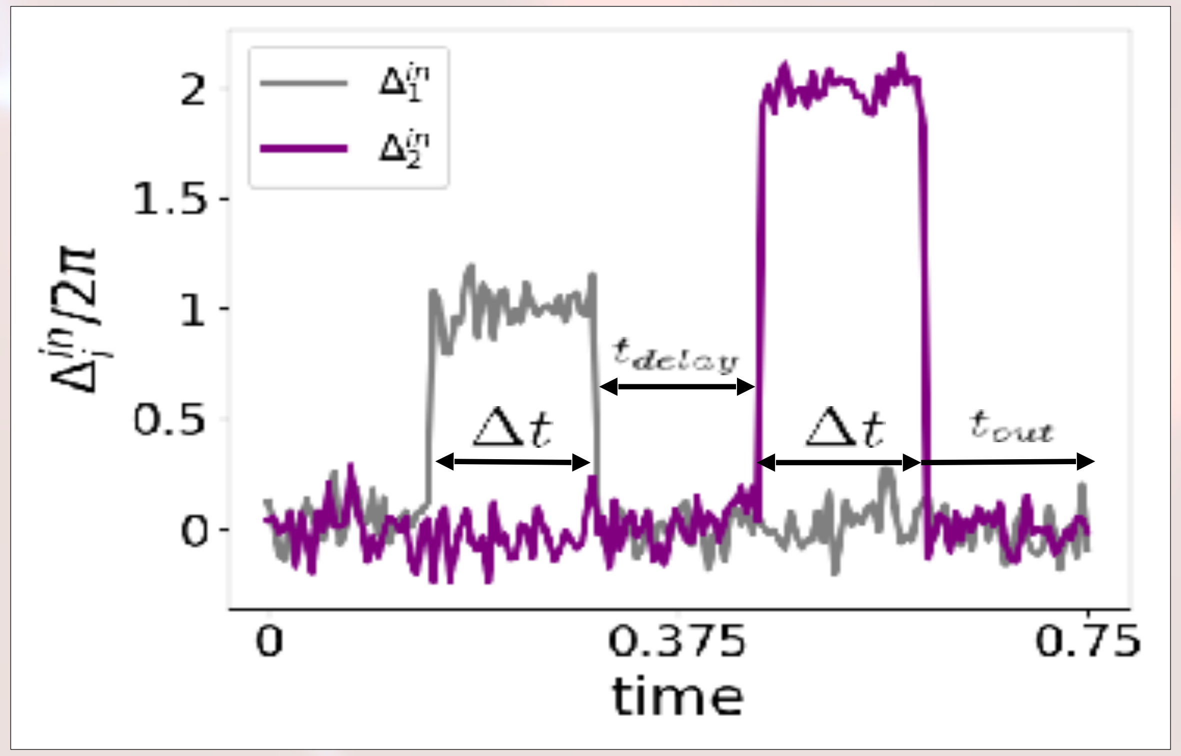
Quantum error detection



Multitasking



Short- and long-term memory



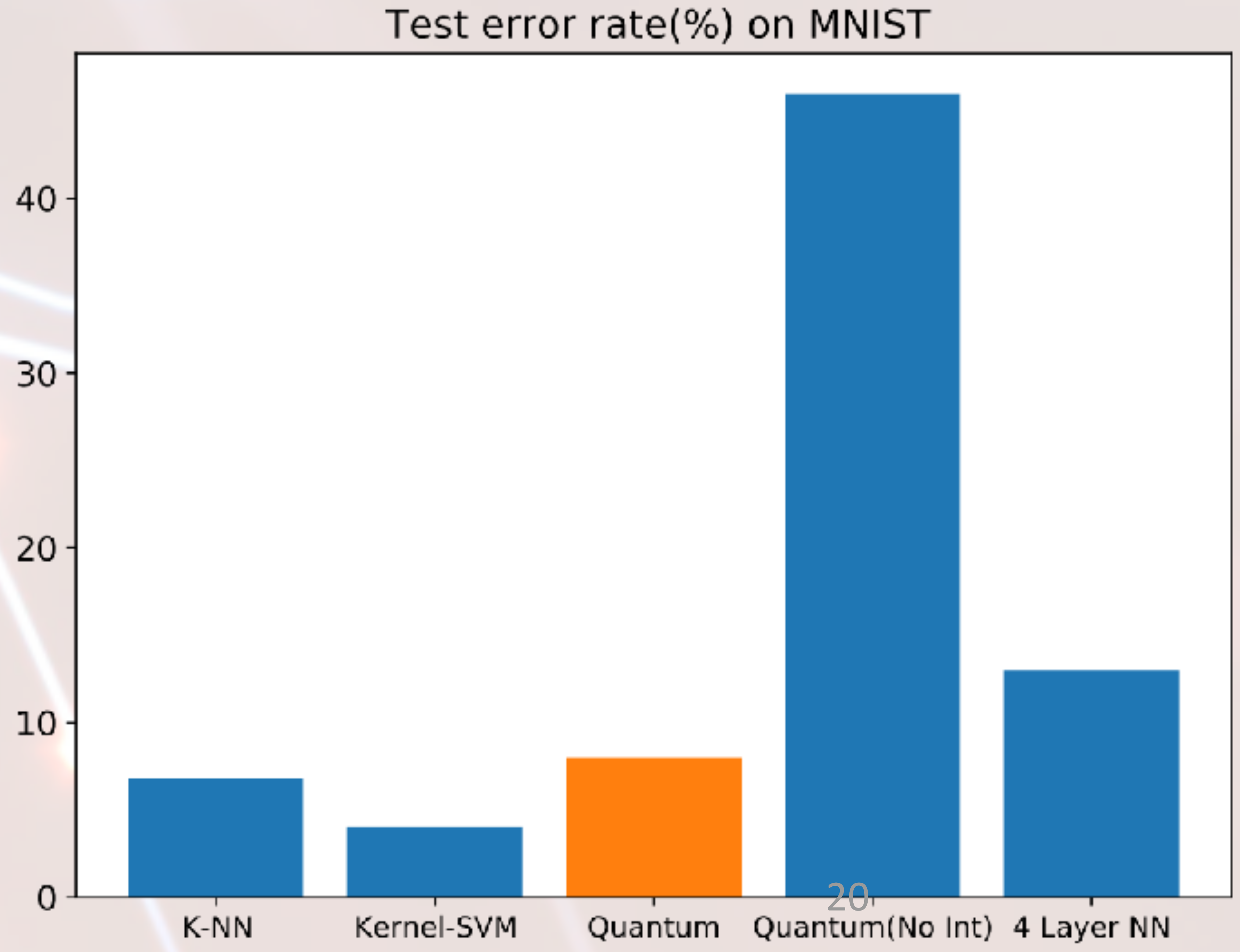
Araiza Bravo, Gao, Najafi, Yelin, PRXQ 3, 030325 (22)

Example: Quantum reservoir computing for pattern recognition

Train: 10K samples
Test: 1K samples
Accuracy: 92%

Classical methods: > 200 neurons, ~10^4 tuning parameters

Our method: Rydberg array (simulated) with 15 atoms



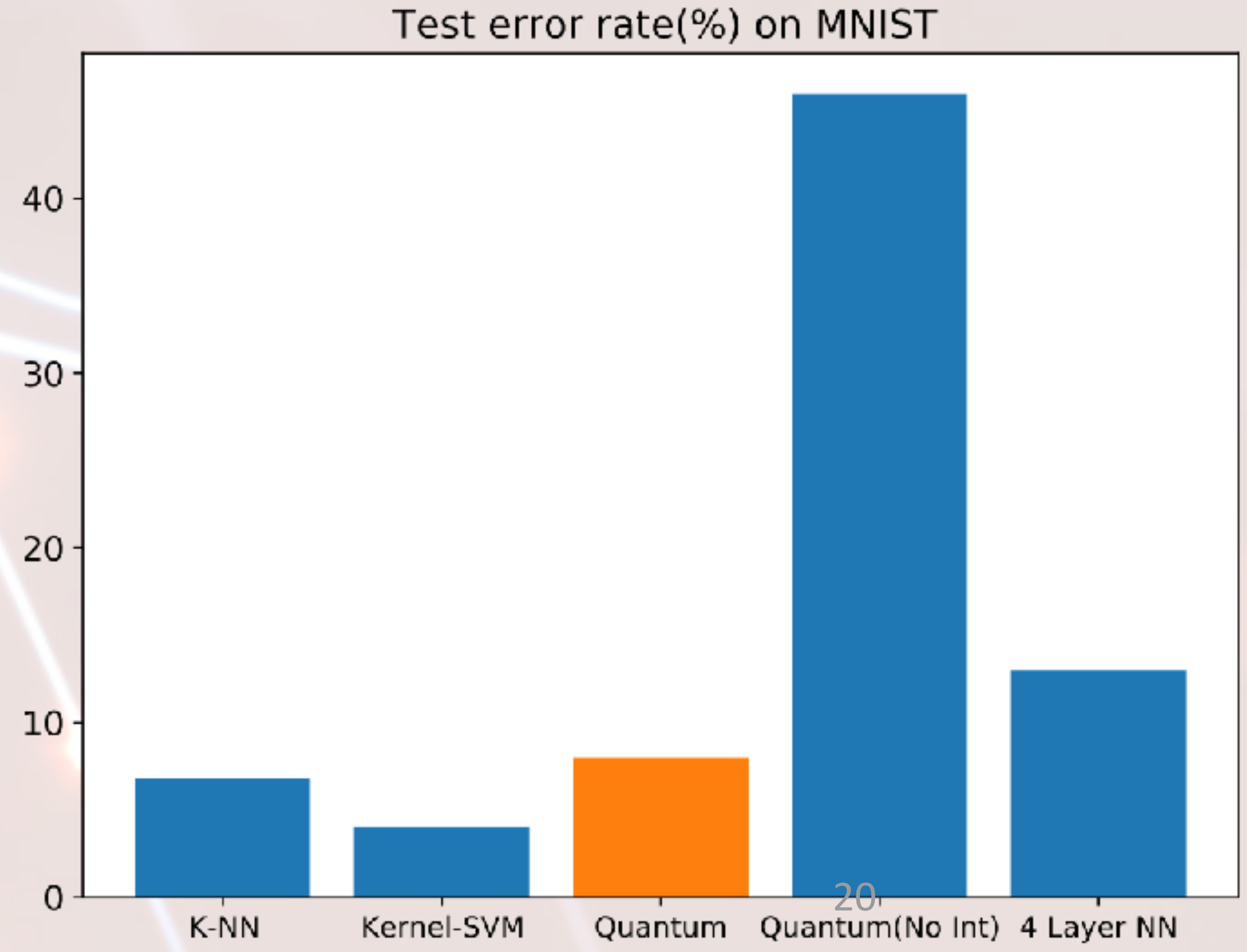
Example: Quantum reservoir computing for pattern recognition

Train: 10K samples
Test: 1K samples
Accuracy: 92%

Classical methods: > 200 neurons, ~10^4 tuning parameters

Our method: Rydberg array
lated) with 15 atoms

Experimental implementation in Rydberg Arrays:
Possible with present setup (small number or input and output qubits, global drive)

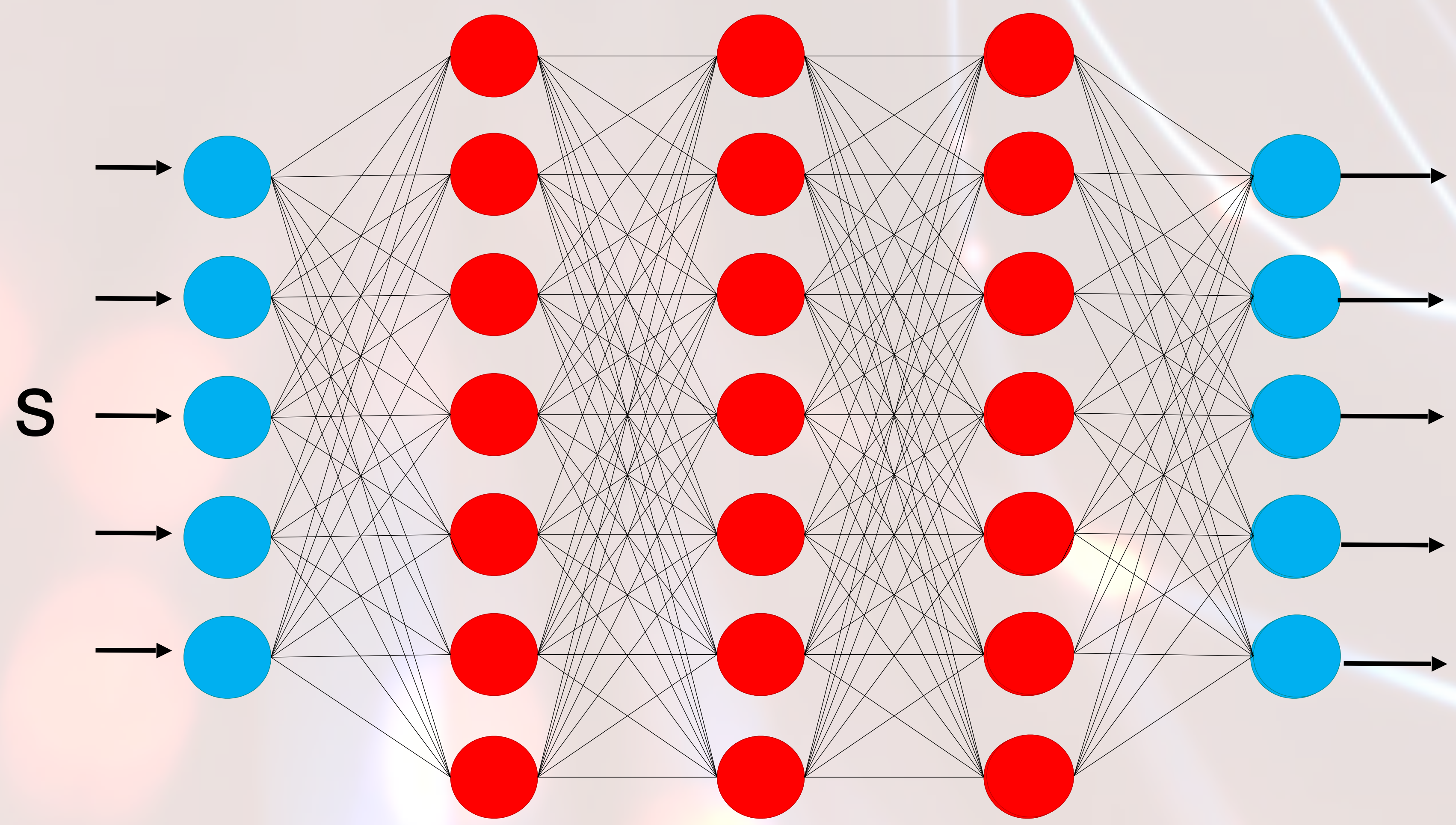


Quantum information with qRNNs

- qRNNs work, but when and why?
- Can one have **provable** quantum advantage?
- Can one build quantum algorithms **systematically**?
- What is the **smallest building block**?

The building blocks of neural networks

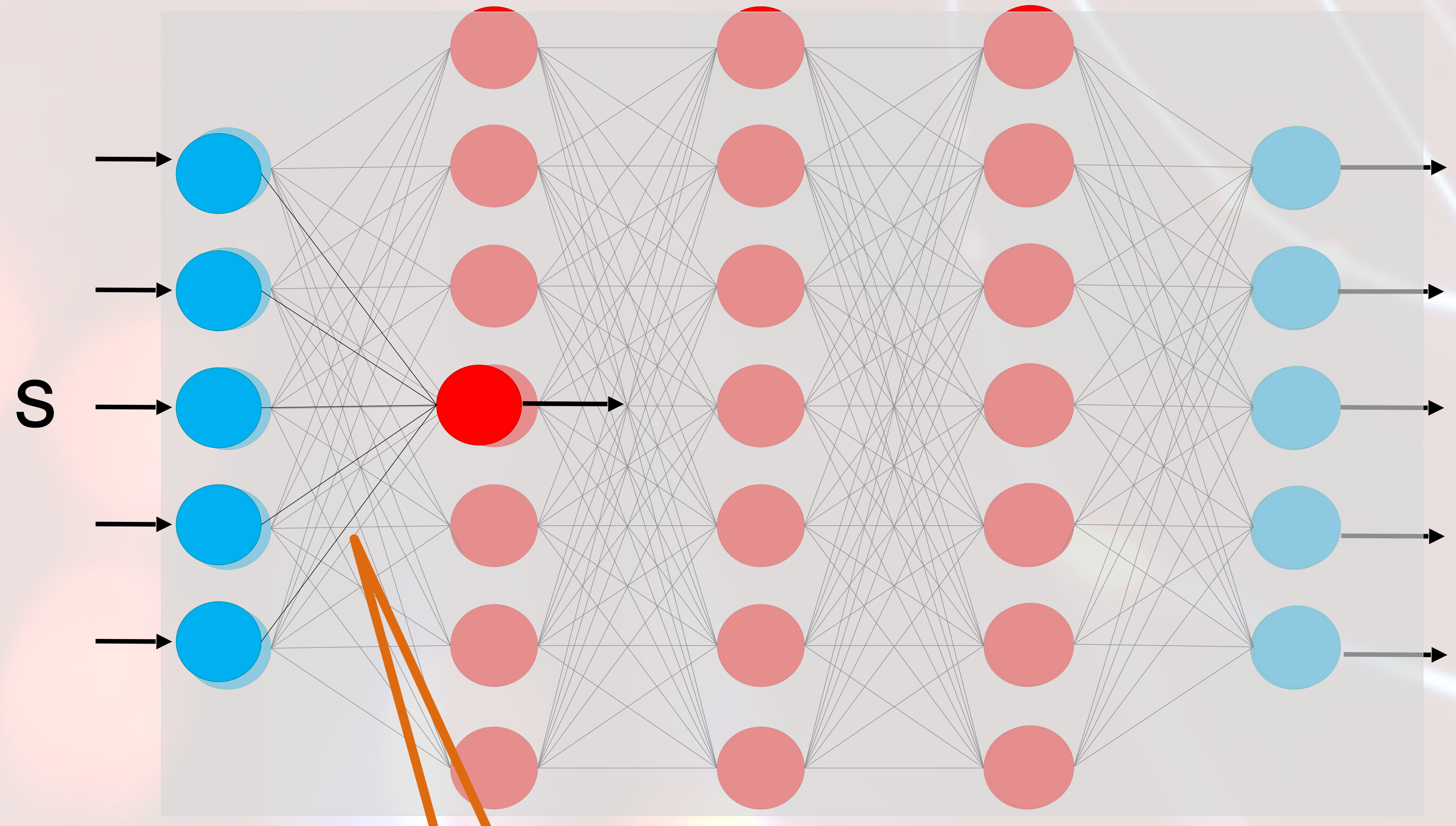
Artificial neural network



[1]Hornik, K., et. al., ScienceDirect (1991)

The building blocks of neural networks

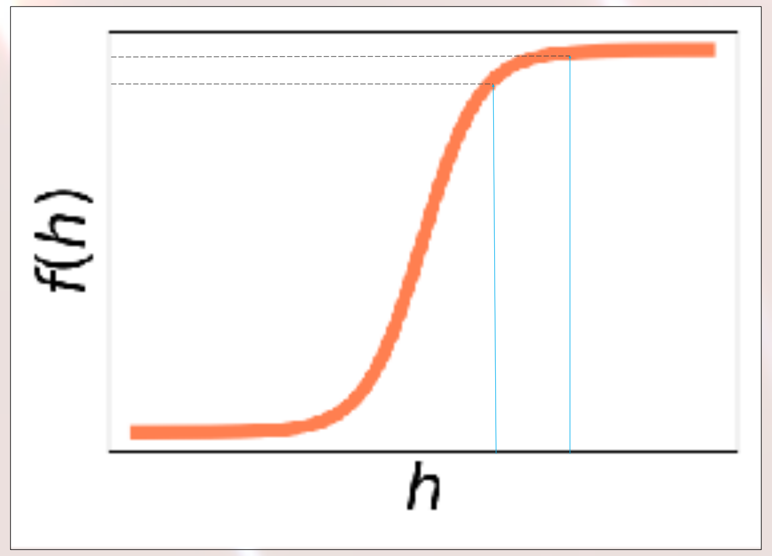
Artificial neural network



Perceptron

$$y^{out} = f\left(-\Delta + \sum_i J_i s_i\right)$$

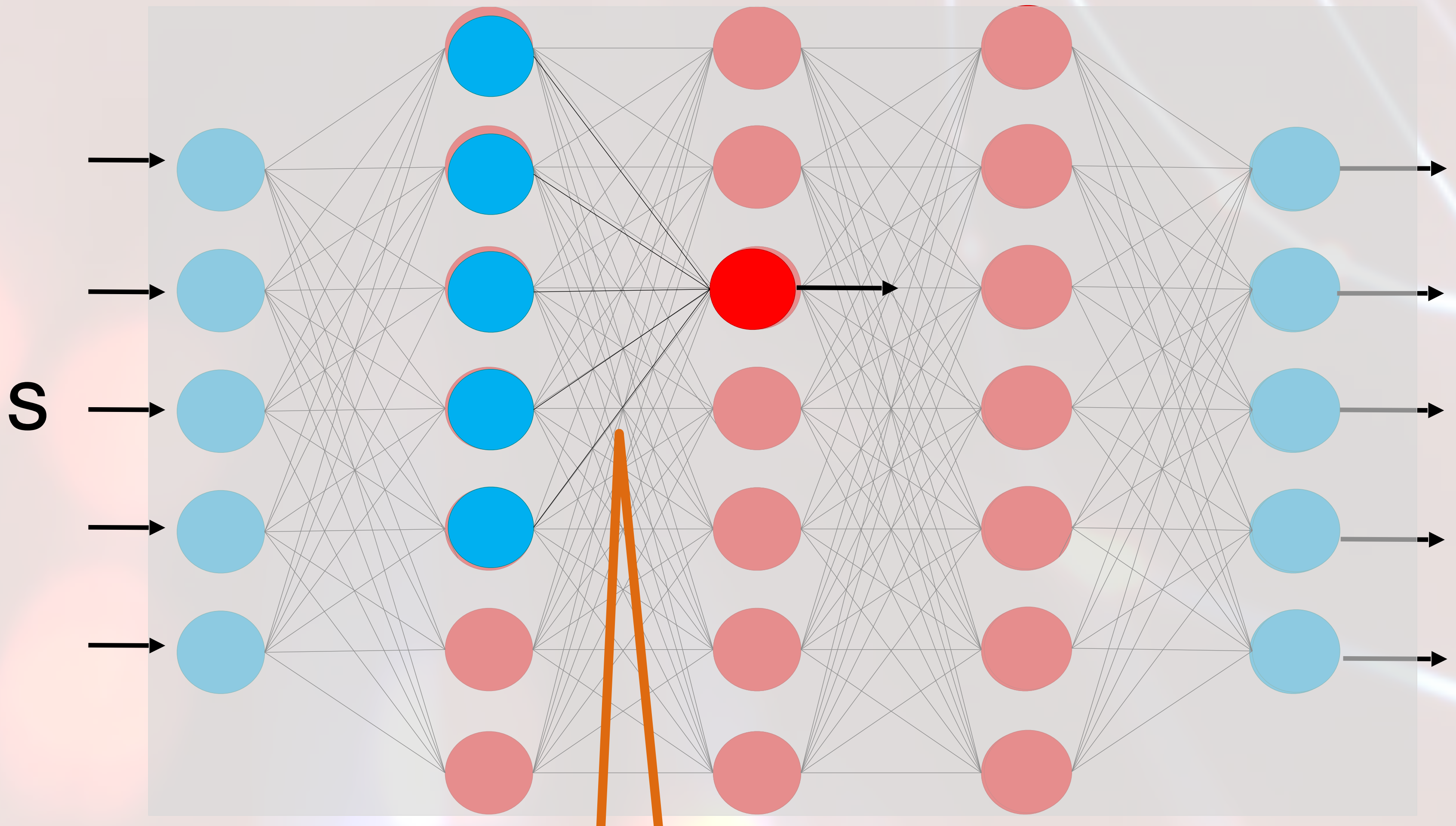
- Perceptrons are an oversimplistic model of neuronal computation.
- A perceptron cannot approximate all functions but many perceptrons together can[1].
- Their architecture makes them resilient to noise in the input.



[1]Hornik, K., et. al., ScienceDirect (1991)

The building blocks of neural networks

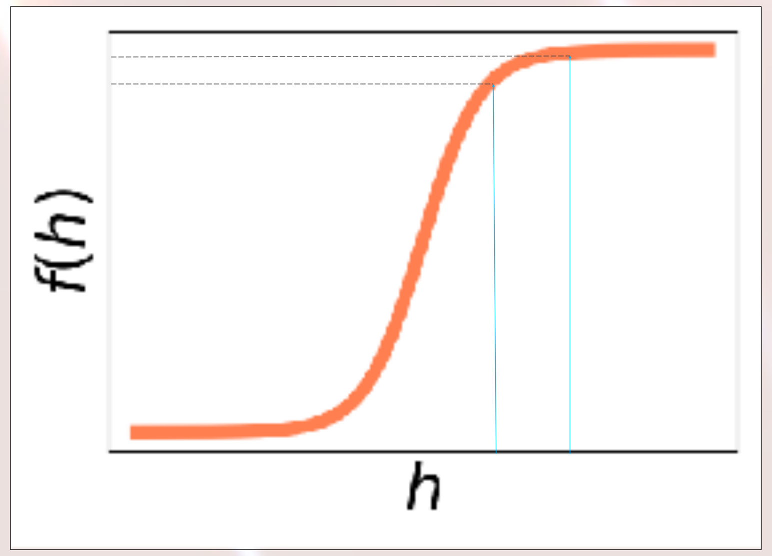
Artificial neural network



Perceptron

$$y^{out} = f\left(-\Delta + \sum_i J_i s_i\right)$$

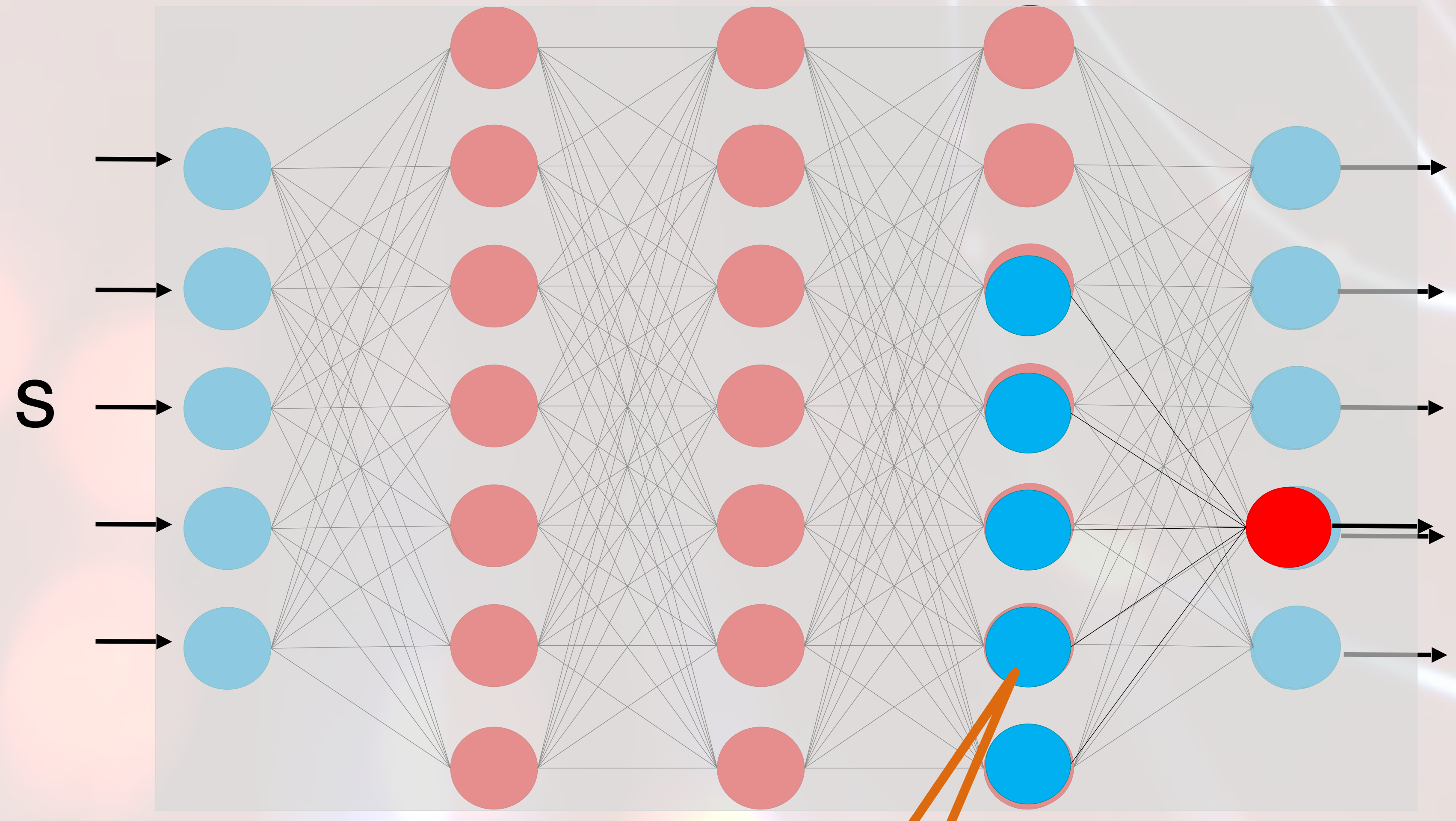
- Perceptrons are an oversimplistic model of neuronal computation.
- A perceptron cannot approximate all functions but many perceptrons together can[1].
- Their architecture makes them resilient to noise in the input.



[1]Hornik, K., et. al., ScienceDirect (1991)

The building blocks of neural networks

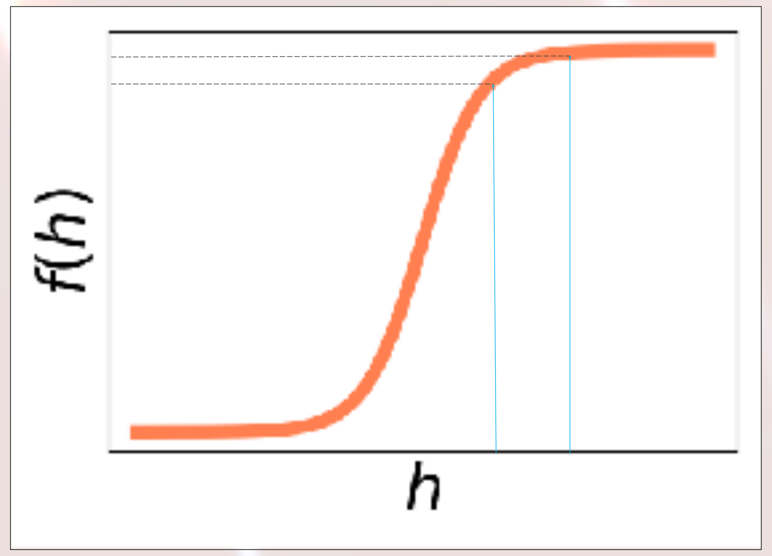
Artificial neural network



Perceptron

$$y^{out} = f\left(-\Delta + \sum_i J_i s_i\right)$$

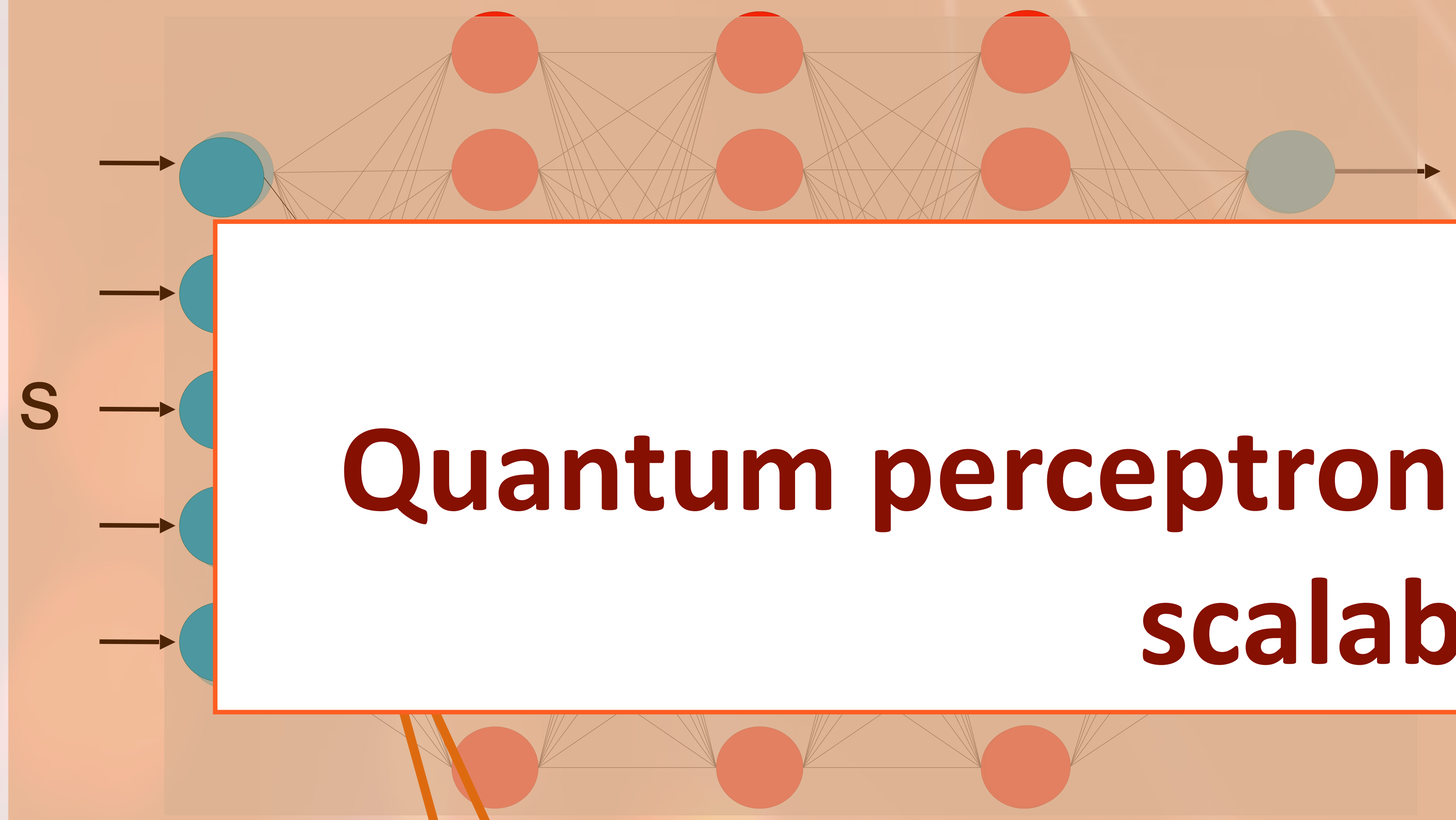
- Perceptrons are an oversimplistic model of neuronal computation.
- A perceptron cannot approximate all functions but many perceptrons together can[1].
- Their architecture makes them resilient to noise in the input.



[1]Hornik, K., et. al., ScienceDirect (1991)

The building blocks of neural networks

Artificial neural network

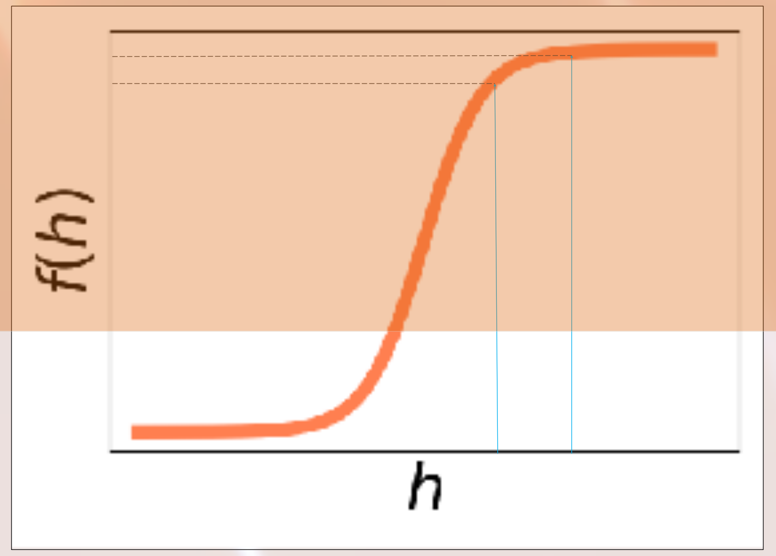


- Perceptrons are an oversimplistic model of neuronal computation.

IDEA:

Quantum perceptron as a simple building block of **scalable** algorithms

- Their architecture makes them resilient to noise in the input.



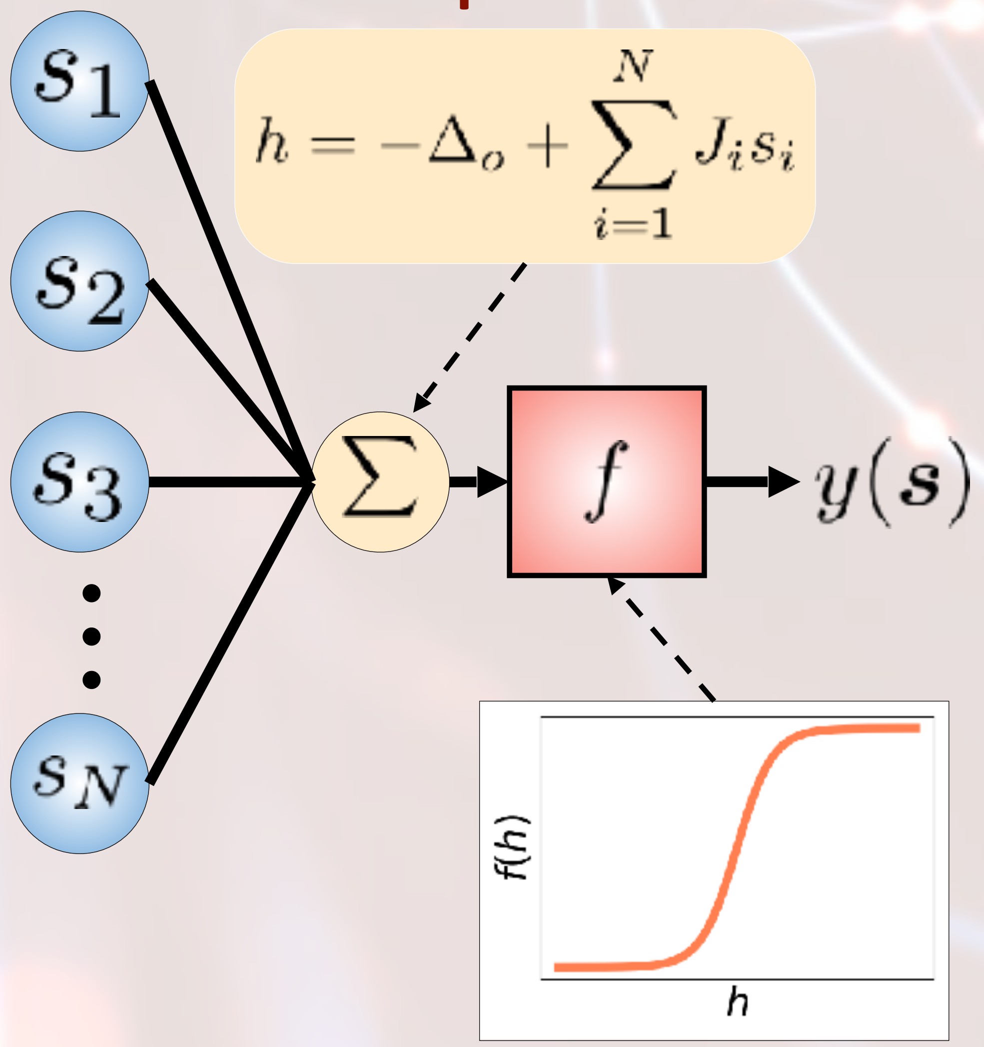
Perceptron

$$y^{out} = f\left(-\Delta + \sum_i J_i s_i\right)$$

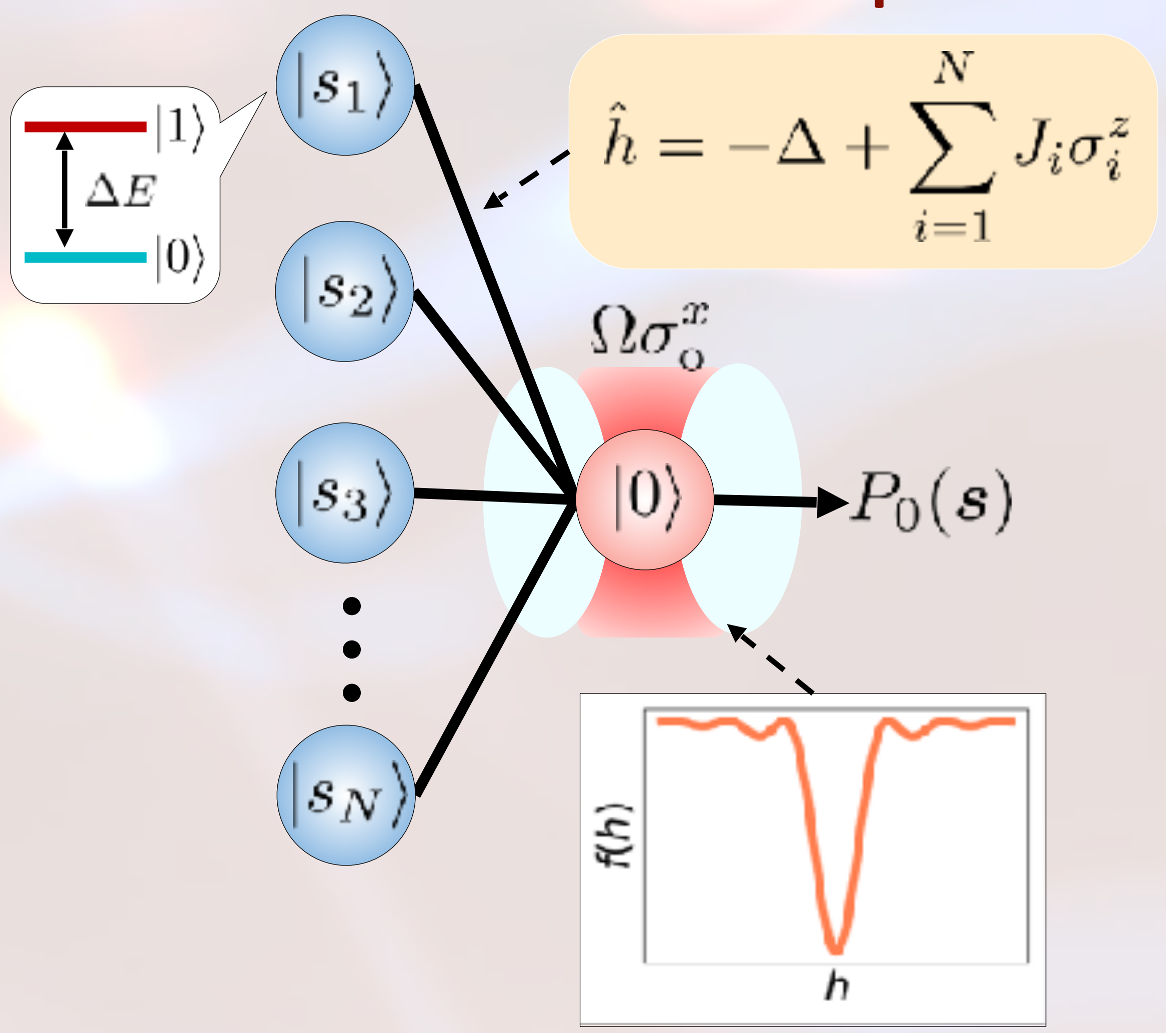
[1]Hornik, K., et. al., ScienceDirect (1991)

Quantum Perceptrons (QP)

Perceptron



Quantum Perceptron



- Inputs are summed to calculate h
- h is passed through a nonlinear function
- An output is produced

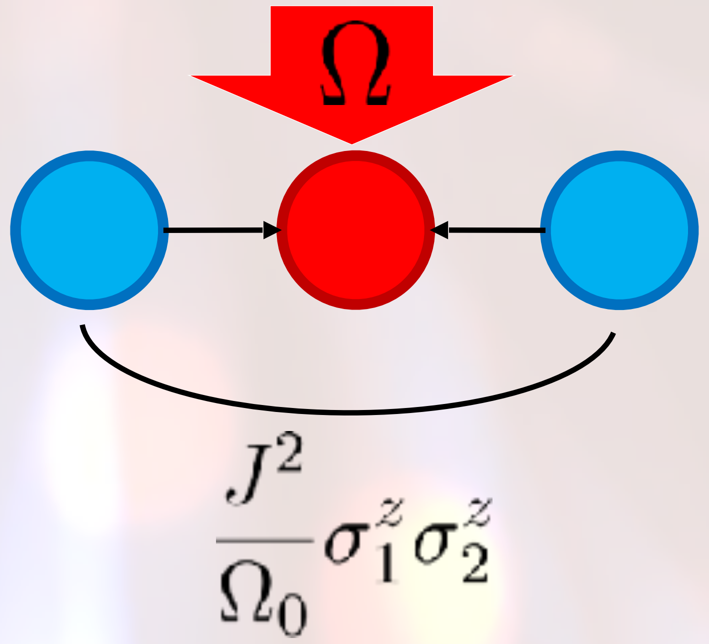
- Input qubits create an effective field h for output qubit
- An extra driving field Ω tries to rotate the output qubit
- The output qubit evolves under competing forces
- The final configuration is the output

A QP is a Universal Quantum Computer

A QP is a **universal quantum** computer if complemented by single-qubit rotations

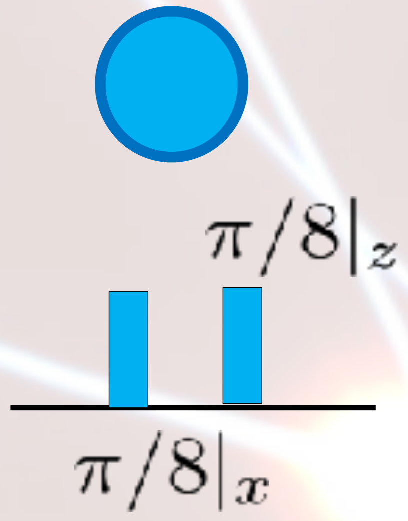
<u>Entangling gate</u>

Large Ω freezes red qubit and the blue qubits can now interact



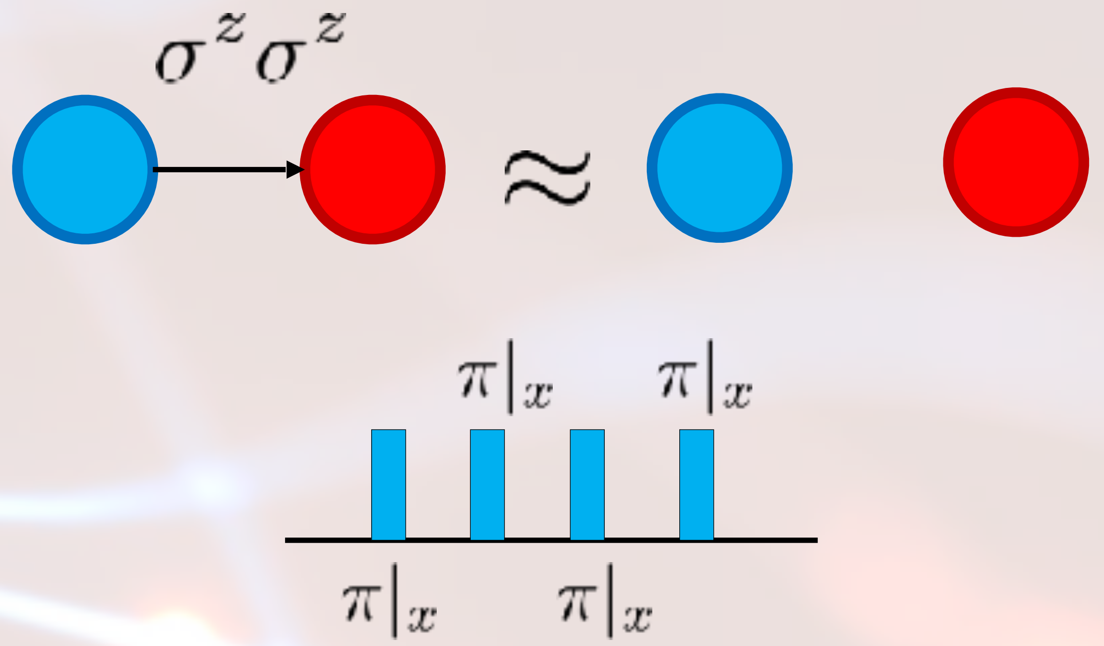
<u>Single-qubit gates</u>

Generated by small pulses on each qubit



<u>Identity gates</u>

Pulses on qubits decouple them from participating in the computation



The upshot: A QP is as powerful as a universal quantum computer
The drawback: Single-qubit pulse control is resource intensive

Araiza Bravo, Najafi, Patti, Gao, Yelin, arXiv:2211.07075; early work: Raussendorf, R., et. al., PRA (2003)

A QP is a Universal Quantum Computer

A QP is a **universal quantum** computer if compl... ...it rotations

Entangling gate

Si...

Large Ω freezes red qu...

and the blue qu...

now intera...

...e

...n

$\pi|_x$

$\pi|_x$ $\pi|_x$

Quantum reservoir computers are universal!

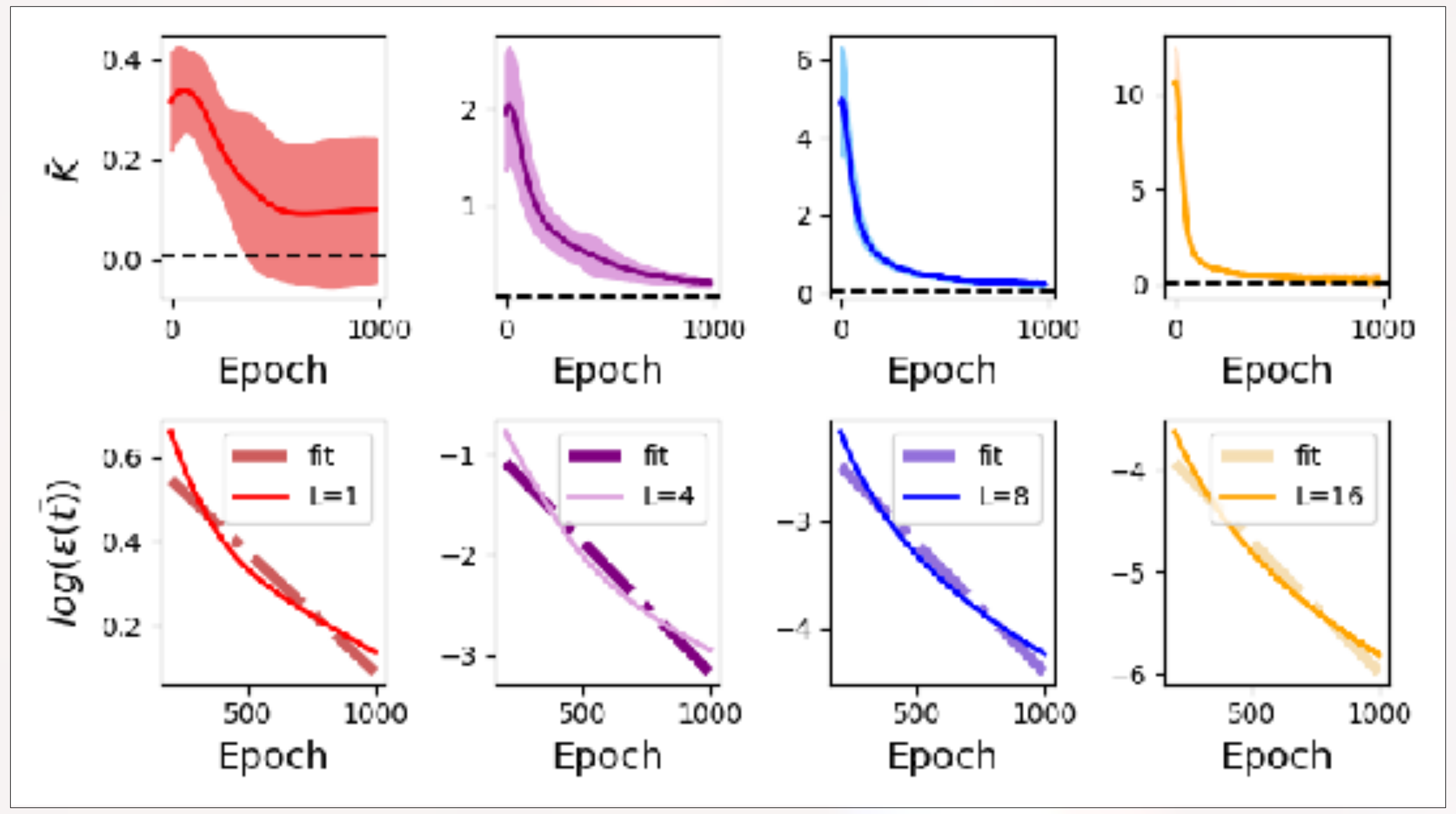
...as a universal ...mputer

...: single-qubit pulse control is resource intensive

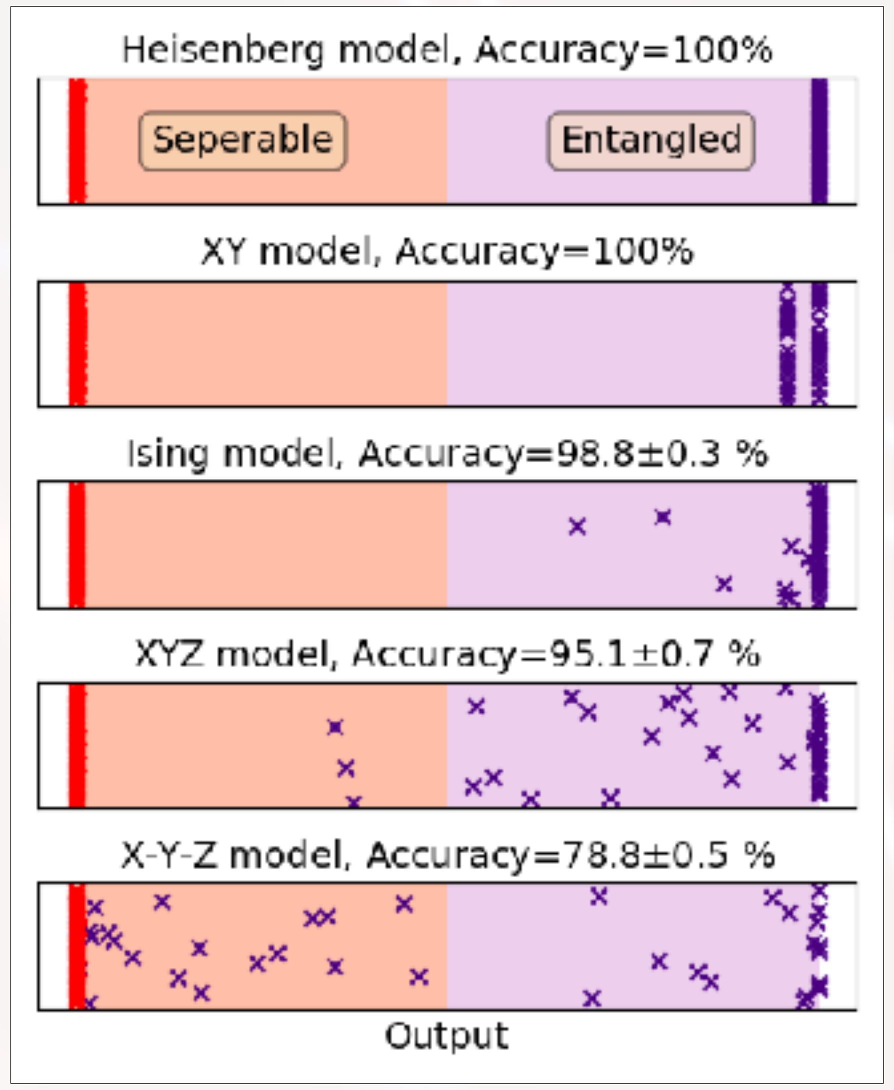
Araiza Bravo, Najafi, Patti, Gao, Yelin, arXiv:2211.07075; early work: Raussendorf, R., et. al., PRA (2003)

Simplest case: single QPs

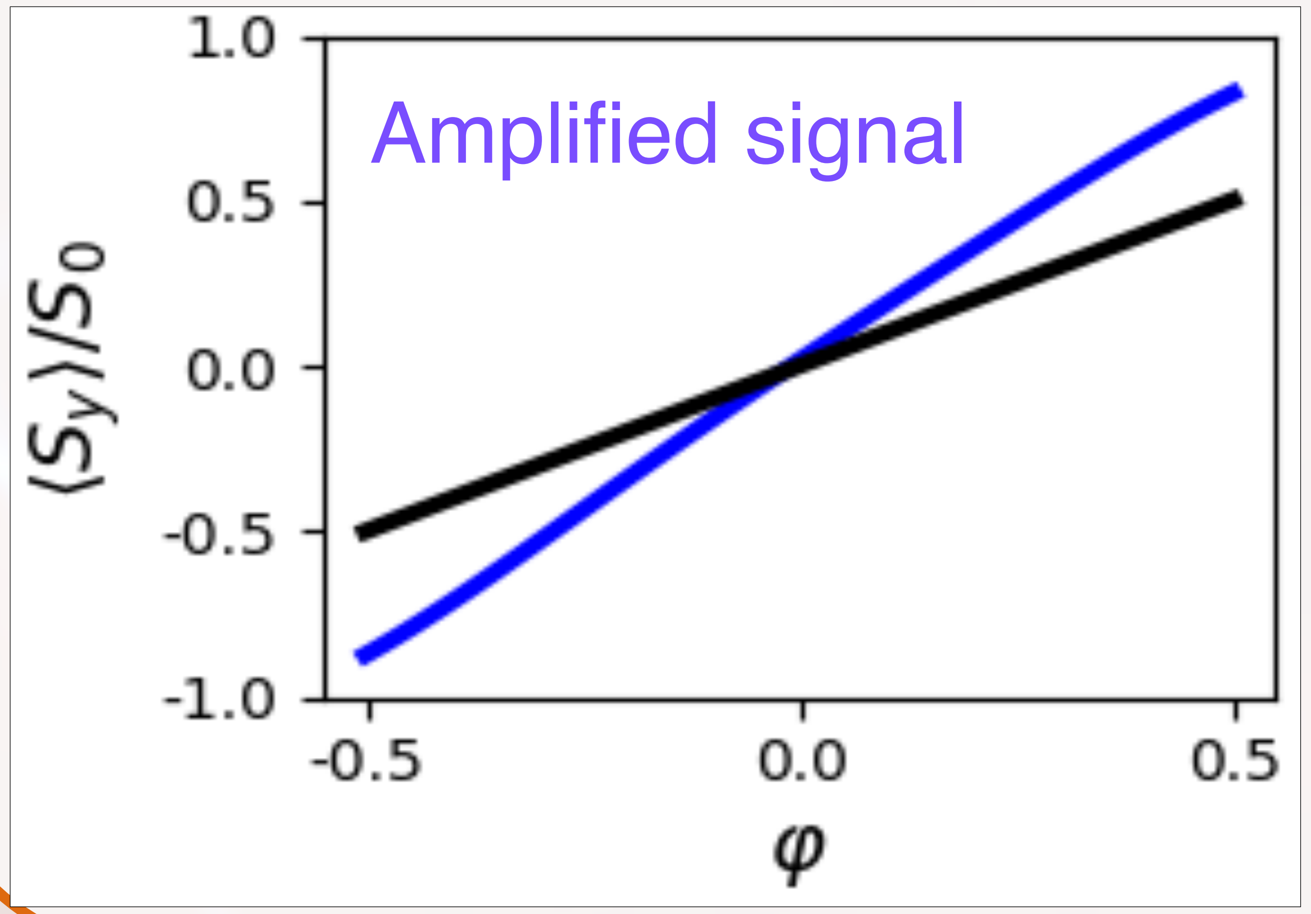
Training compares favorably to other quantum algorithms



Energy measurement and entanglement detection



Quantum metrology: measuring fundamental constants in the lab



Araiza Bravo, Najafi, Patti, Gao, Yelin, arXiv:2211.07075

Analog quantum-classical learning

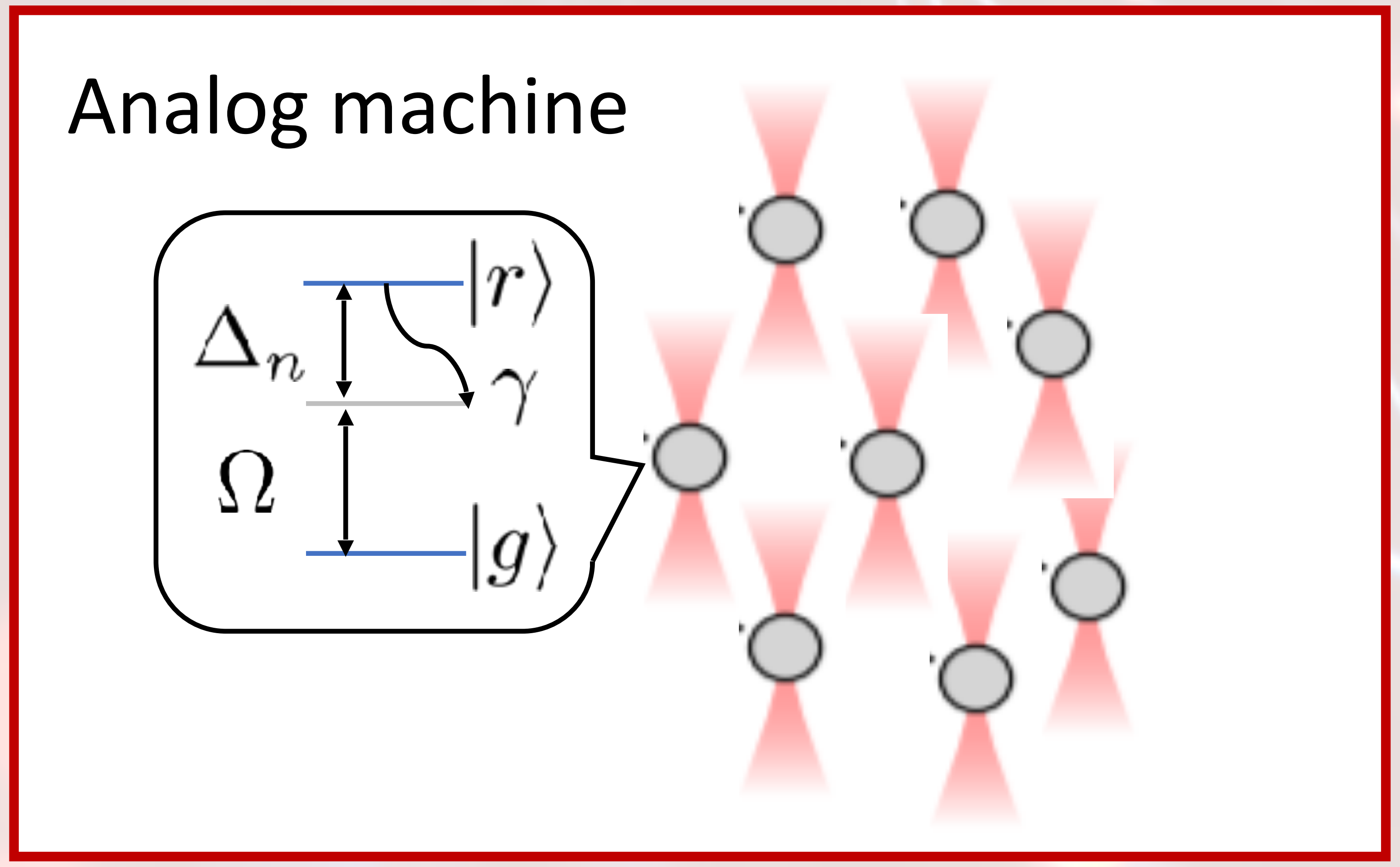
- Goal: Make the most of a quantum computer

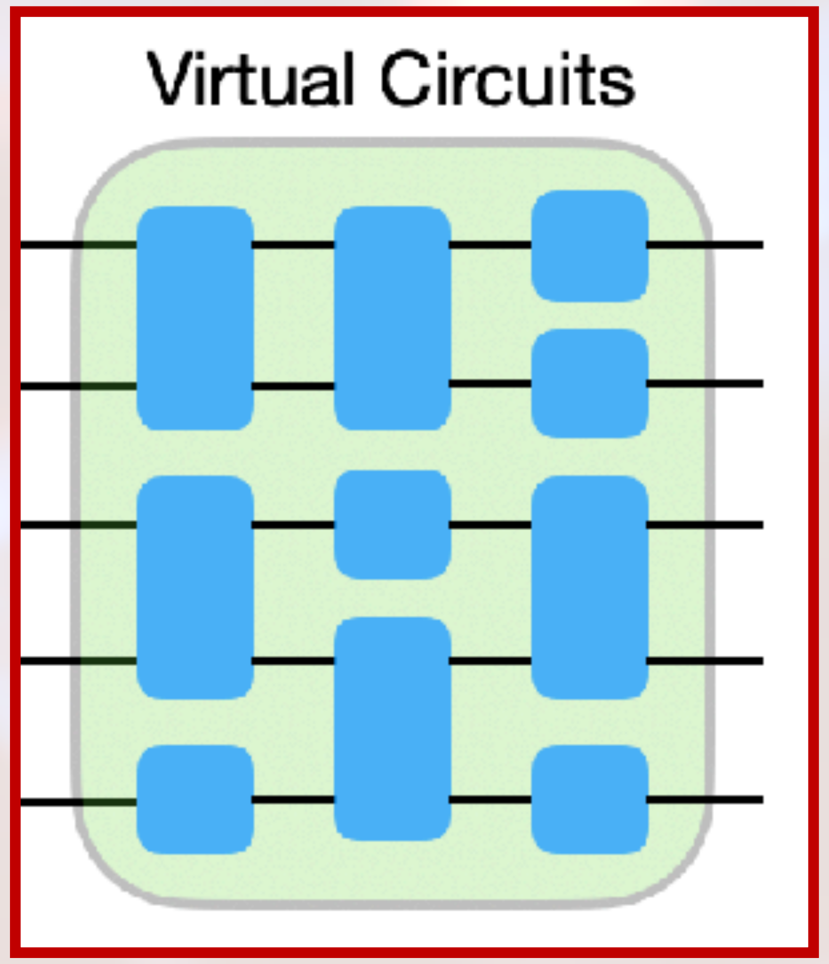
  (Here: analog quantum machine)

- How?
  - Classically simulable $\longrightarrow$ classical
  - Exponentially expensive $\longrightarrow$ quantum

Gu, Hu, Luo, Patti, Rubin, Yelin, arXiv:2308.11616 (2023).

Analog Quantum-Classical Hybrid Machine Learning



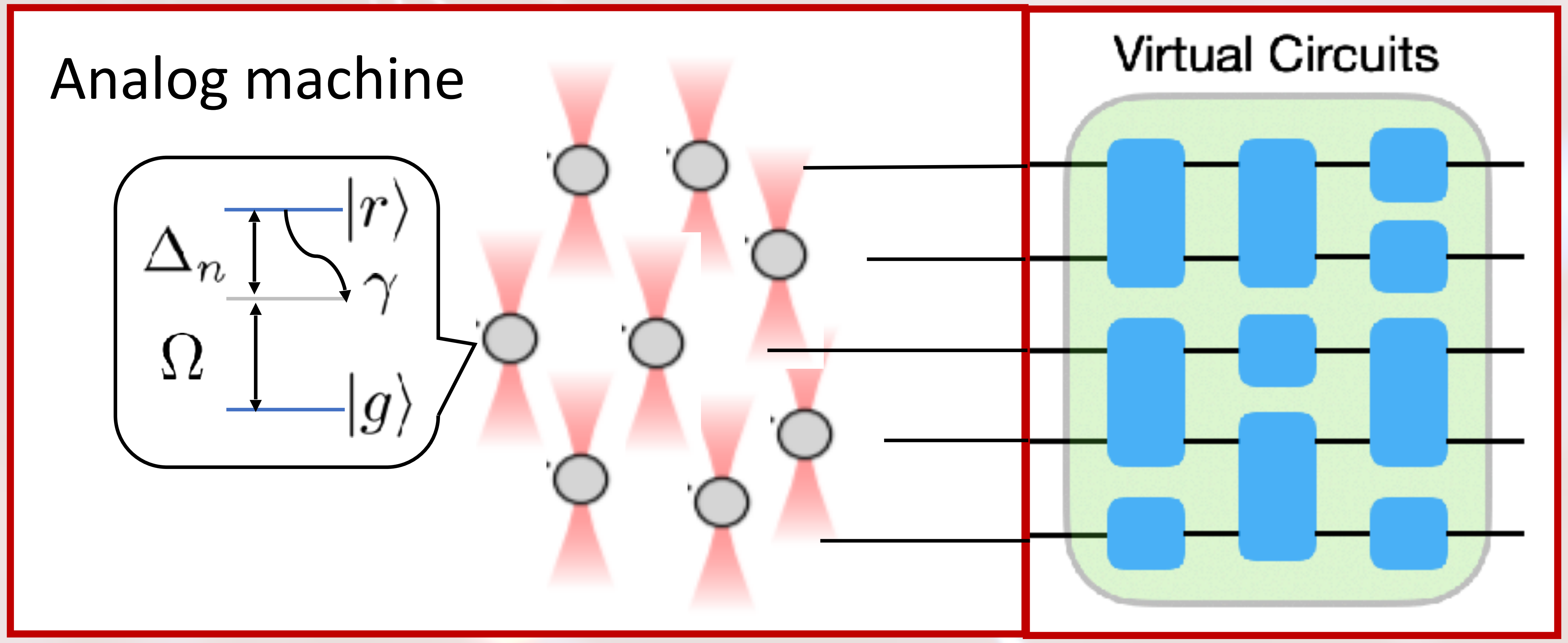


Example: Clifford circuits

- Limited tunability
- Highly expressible (quantum correlations)
- Evolution time limited by decoherence

- Tunable
- Not expressible (classically simulable)
- Long time evolution (no errors)

Analog Quantum-Classical Hybrid Machine Learning

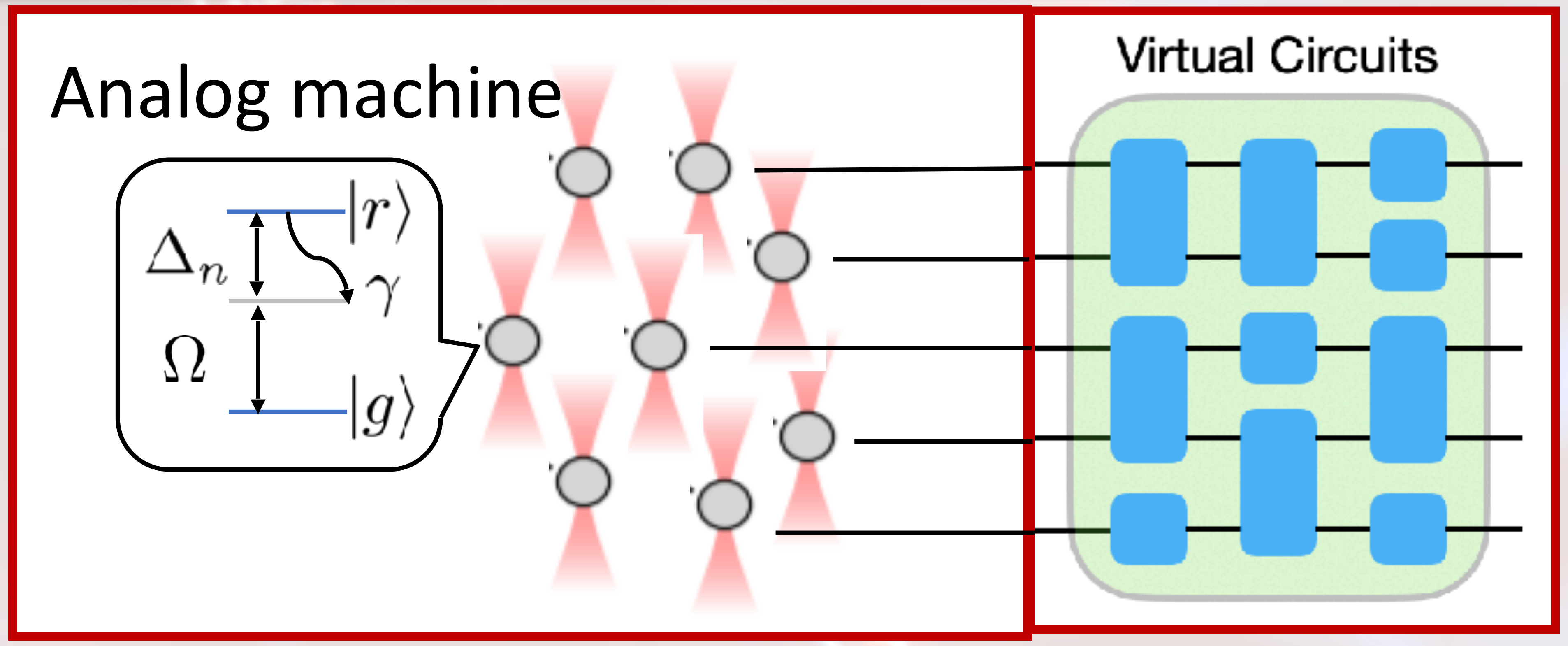


Example: Clifford circuits

- Limited tunability
- Highly expressible (quantum correlations)
- Evolution time limited by decoherence

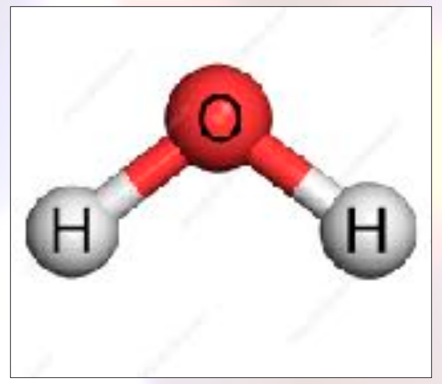
- Tunable
- Not expressible (classically simulable)
- Long time evolution (no errors)

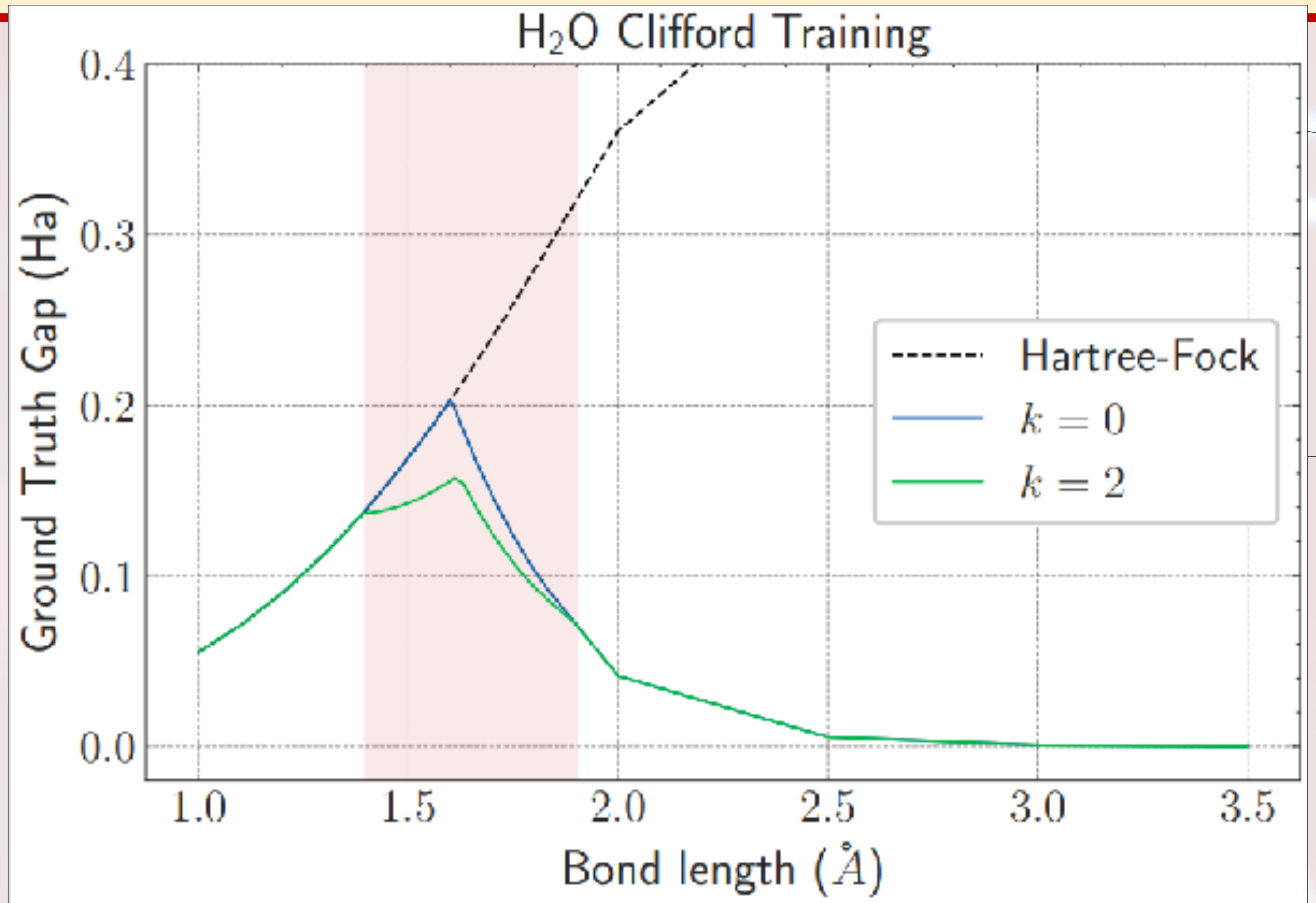
Analog Quantum-Classical Hybrid



Example:
Clifford
circuits

Combining these two approaches promises enhanced expressivity and trainability all while on a near-term device!



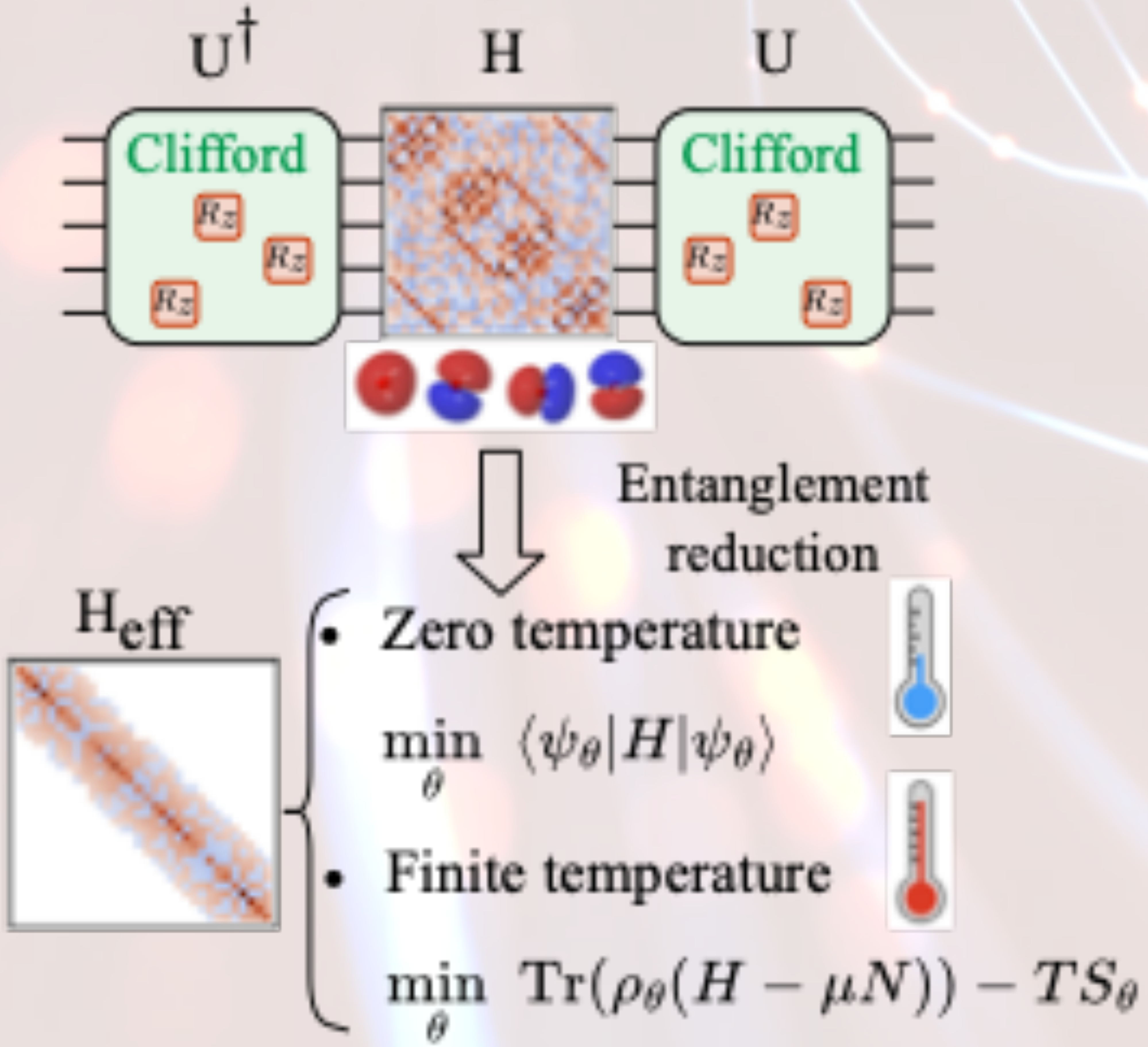


Classical simulation

Quantum simulations

Analog Quantum-Classical Hybrid Machine Learning

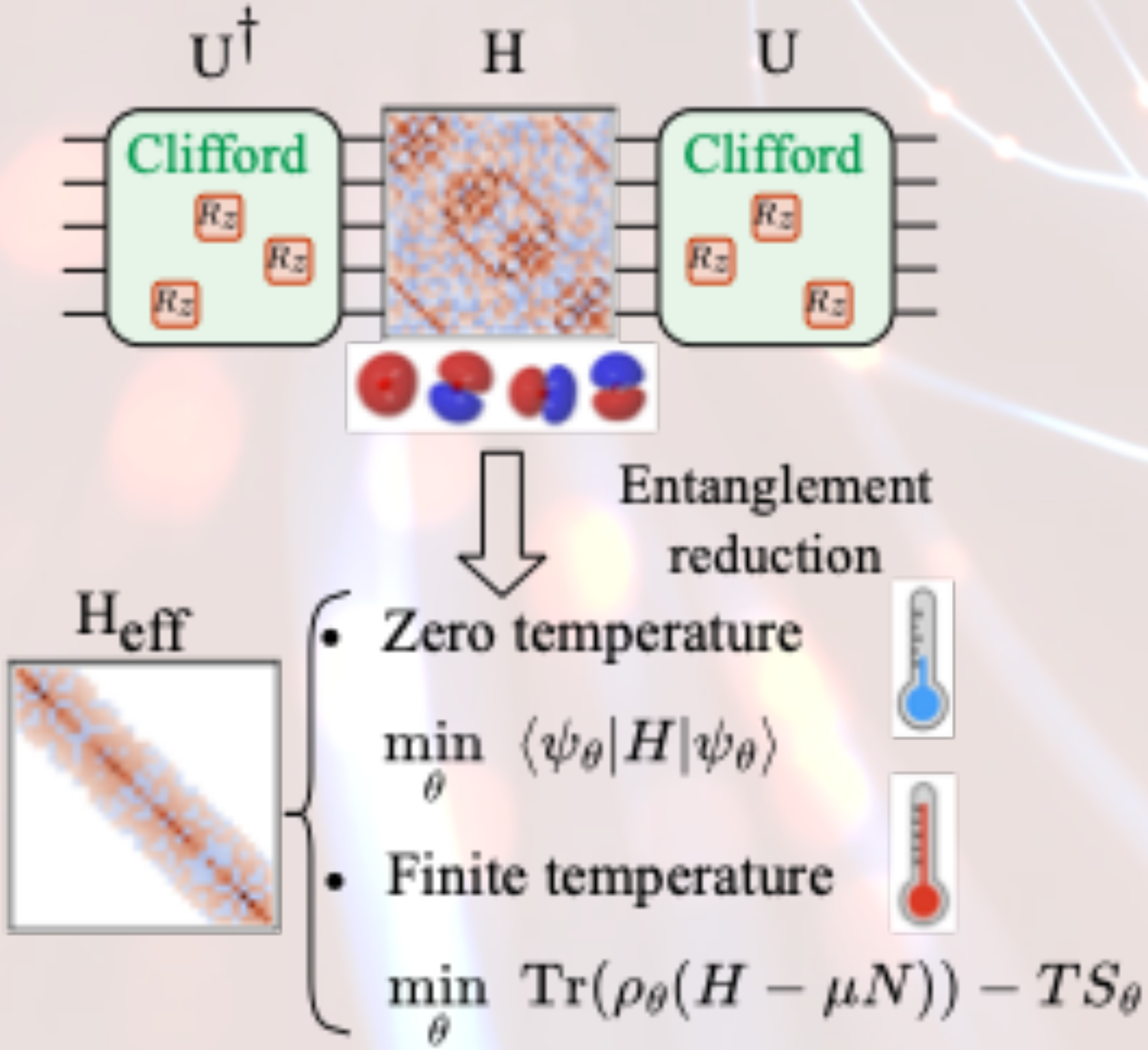
Clifford + T circuit acts as **basis transformation**: (Entanglement reduction)



Analog Quantum-Classical ...earning

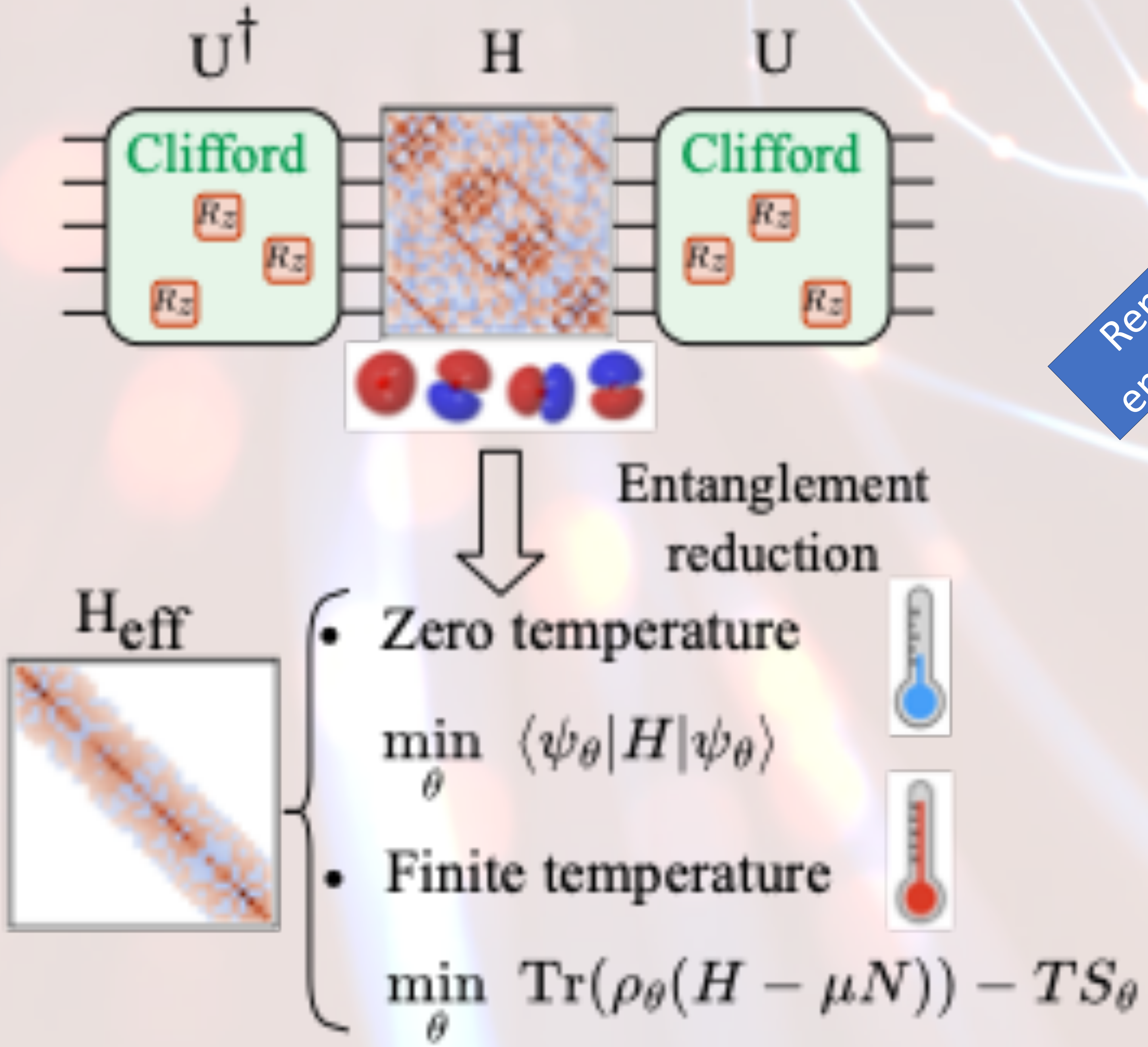
classically simulable (in polynomial time)

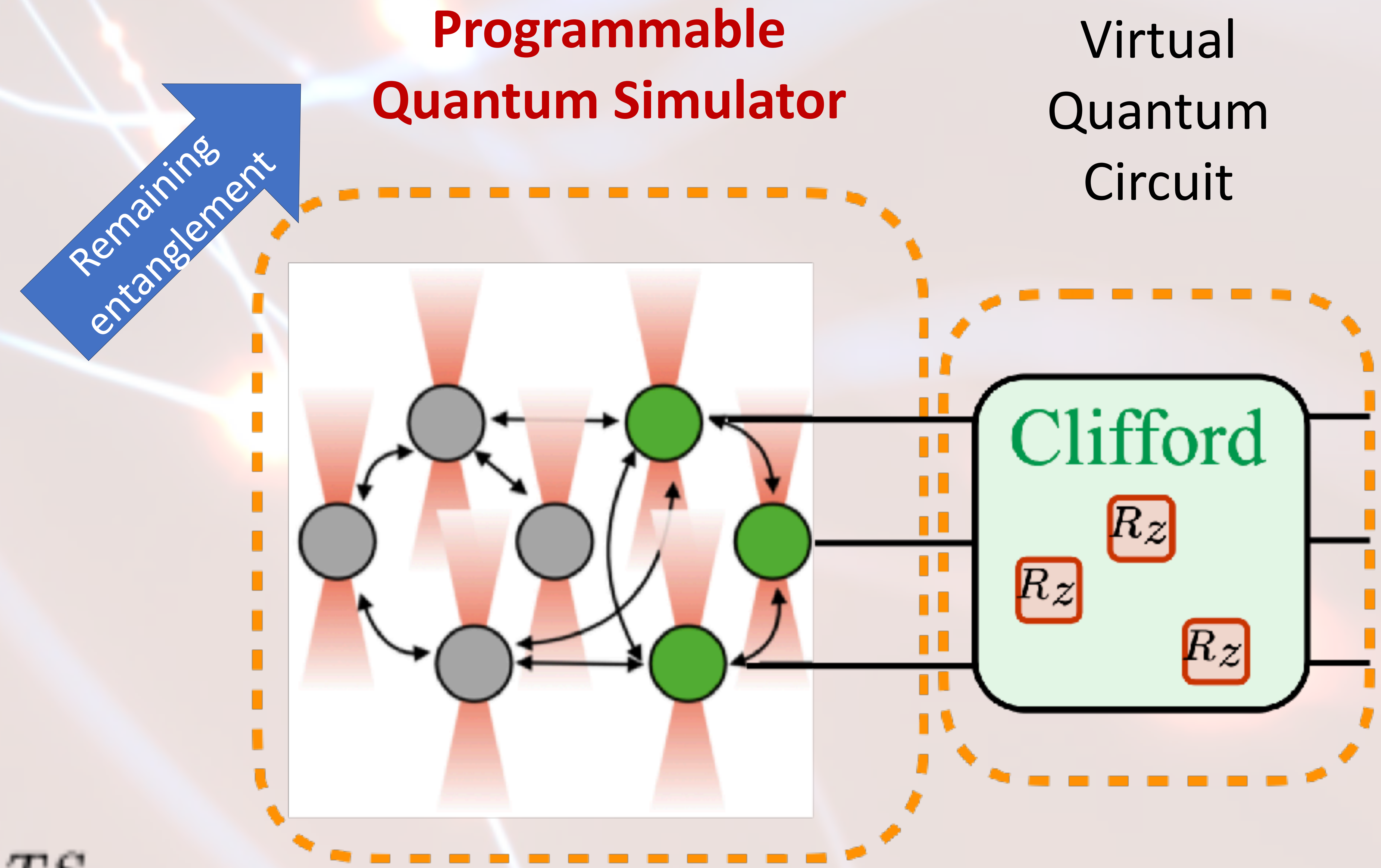
Clifford + T circuit acts as **basis transformation**: (Entanglement reduction)



Analog Quantum-Classical Hybrid Machine Learning

Clifford + T circuit acts as **basis transformation**: (Entanglement reduction)





Outlook

- Scale up systems
- Applications to quantum chemistry

Collaborators

Rodrigo Araiza Bravo

Taylor Patti

Xun Gao

Yidan Wang

Hong-Ye Hu

Andi Gu

Di Luo

Sona Najafi

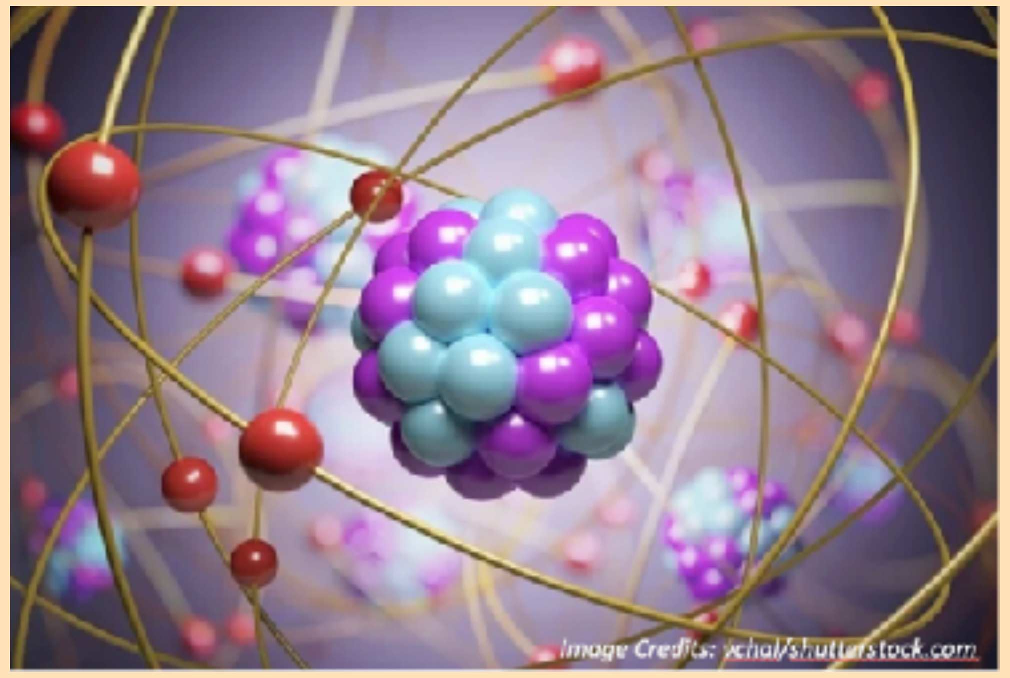
Oriol Rubies Bigorda

- $$$: NSF-CUA, DOE, HQI, NSF-QSEnSE, NSF-HDR

Quantum Chemistry and Material Science

Quantum Chemistry

= solving an electronic structure problem for a configuration of electrons and nuclei



Major thrust of quantum chemistry: quantitative prediction of material or molecular properties

Full Hamiltonian:

$$H = -\sum_{i} \frac{\hbar^2 \nabla^2}{2m_i} - \sum_{k \neq i} \frac{Z_k e^2}{|\mathbf{R}_k - \mathbf{r}_i|} + \sum_{i \neq j} \frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|}$$

Challenge: Simulating systems with **strong correlations**

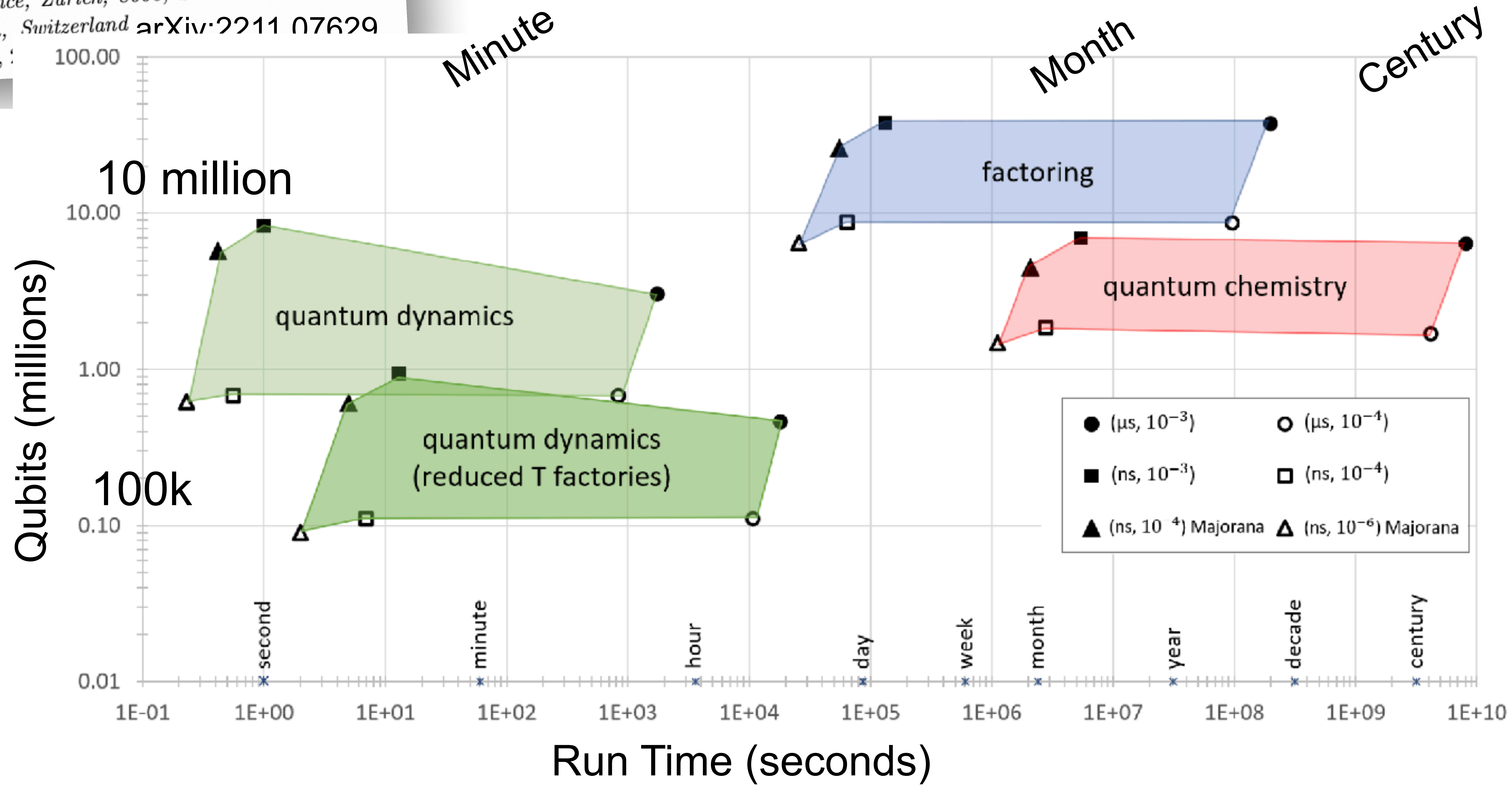
➤ Unfavorable Hilbert space scaling motivates use of **quantum computers**

Quantum Chemistry on Quantum Computers

Assessing requirements to scale to practical quantum advantage

M. E. Beverland,[1] P. Murali,[1] M. Troyer,[1] K. M. Svore,[1] T. Hoefler,[2]
V. Kliuchnikov,[1] G. H. Low,[1] M. Soeken,[3] A. Sundaram,[1] and A. Vaschillo[1]
[1] *Microsoft Quantum, Redmond, WA 98052, USA*
[2] *ETH Zurich, Department of Computer Science, Zürich, 8006, Switzerland*
[3] *Microsoft Quantum, Zurich, Switzerland*
(Dated: November 19,

arXiv:2211.07629

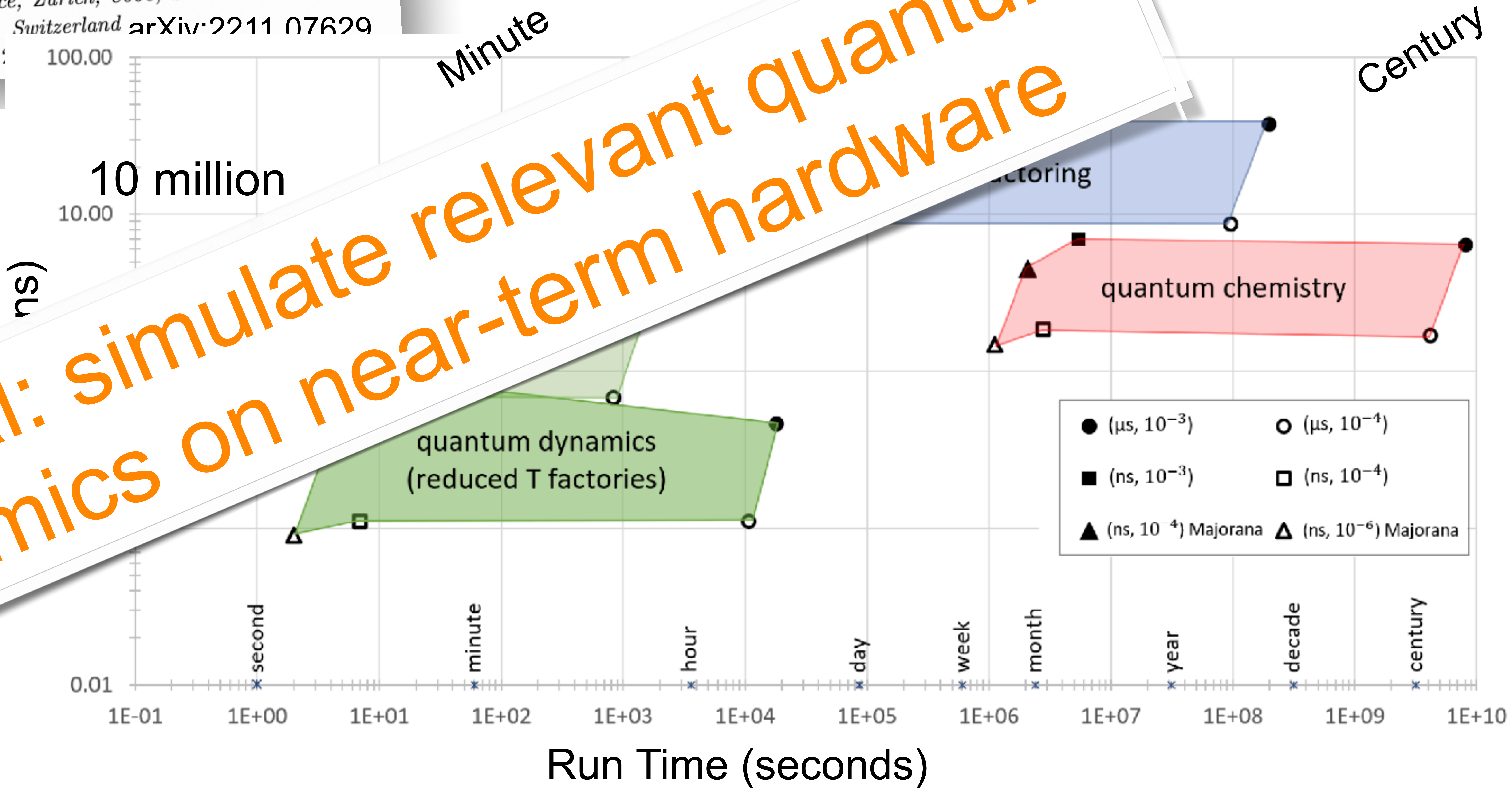


Quantum Chemistry on Quantum Computers

Assessing requirements to scale to practical quantum advantage

M. E. Beverland,[1] P. Murali,[1] M. Troyer,[1] K. M. Svore,[1] T. Hoefler,[2]
V. Kliuchnikov,[1] G. H. Low,[1] M. Soeken,[3] A. Sundaram,[1] and A. Vaschillo[1]
[1] *Microsoft Quantum, Redmond, WA 98052, USA*
[2] *ETH Zurich, Department of Computer Science, Zürich, 8006, Switzerland*
[3] *Microsoft Quantum, Zurich, Switzerland*
(Dated: November 19,

arXiv:2211.07629



Our goal: simulate relevant quantum dynamics on near-term hardware

Advancing Computational Quantum Chemistry

Our approach:

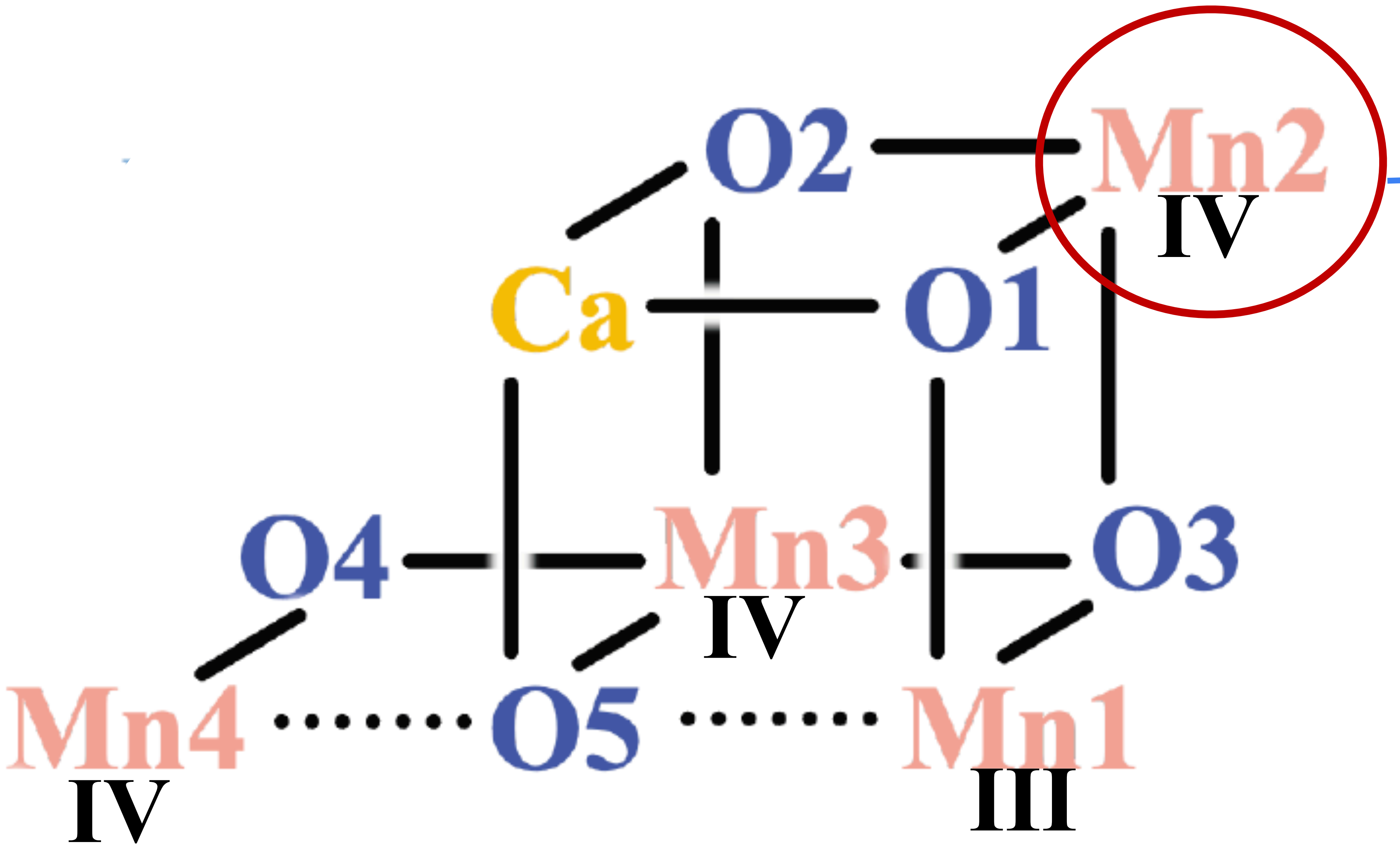
- Leverage insights obtained from state-of-the art **classical computational algorithms**.
- Use state-of-the art **programmable quantum simulators** (e.g., Rydberg atom arrays)
- Focus on **hardware-efficient** implementations on near term devices.

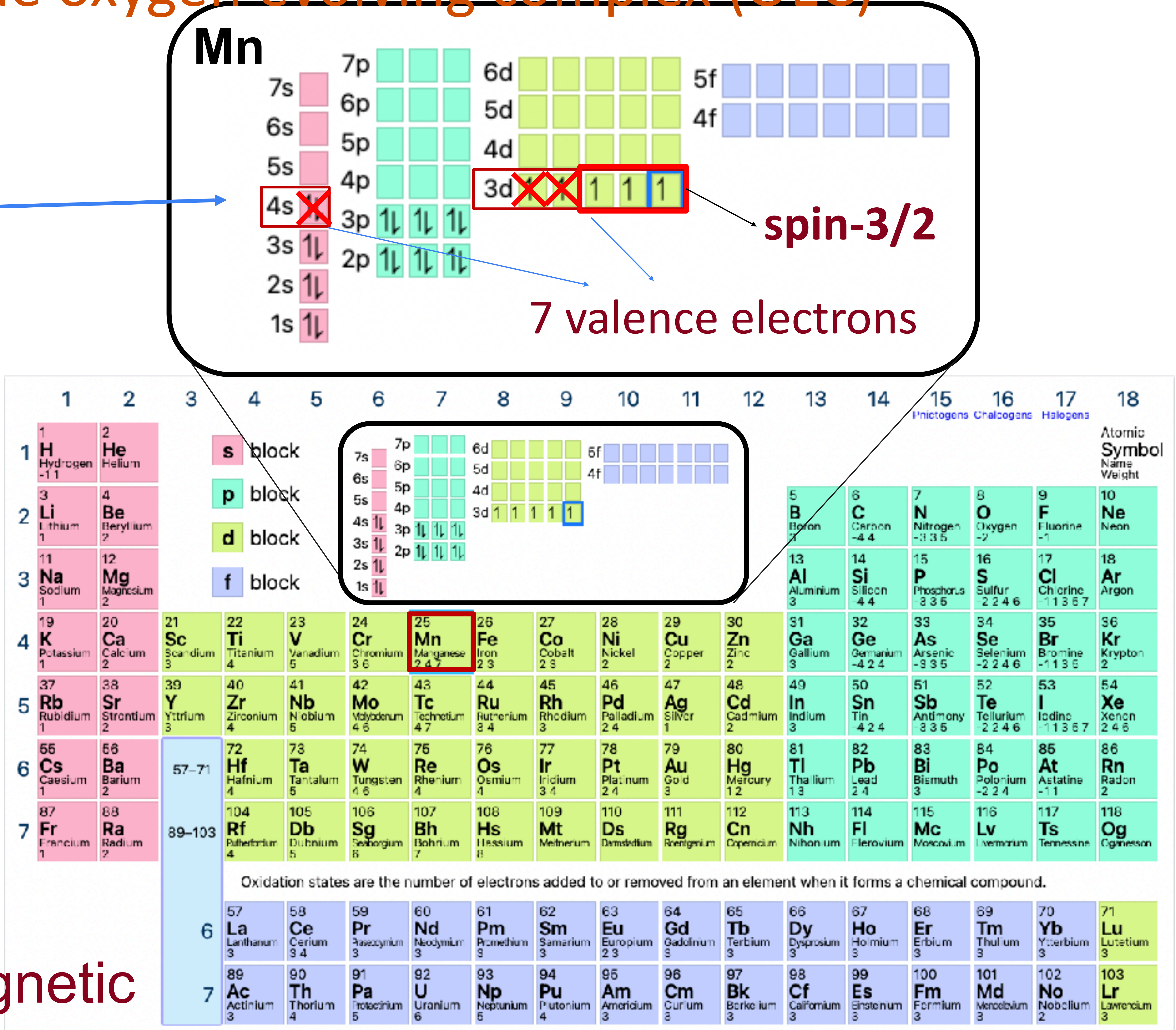
What problems do need a quantum computer?

problems with strong correlations

Model Hamiltonians

Example: Biochemical catalyst involved in the oxygen evolving complex (OEC)



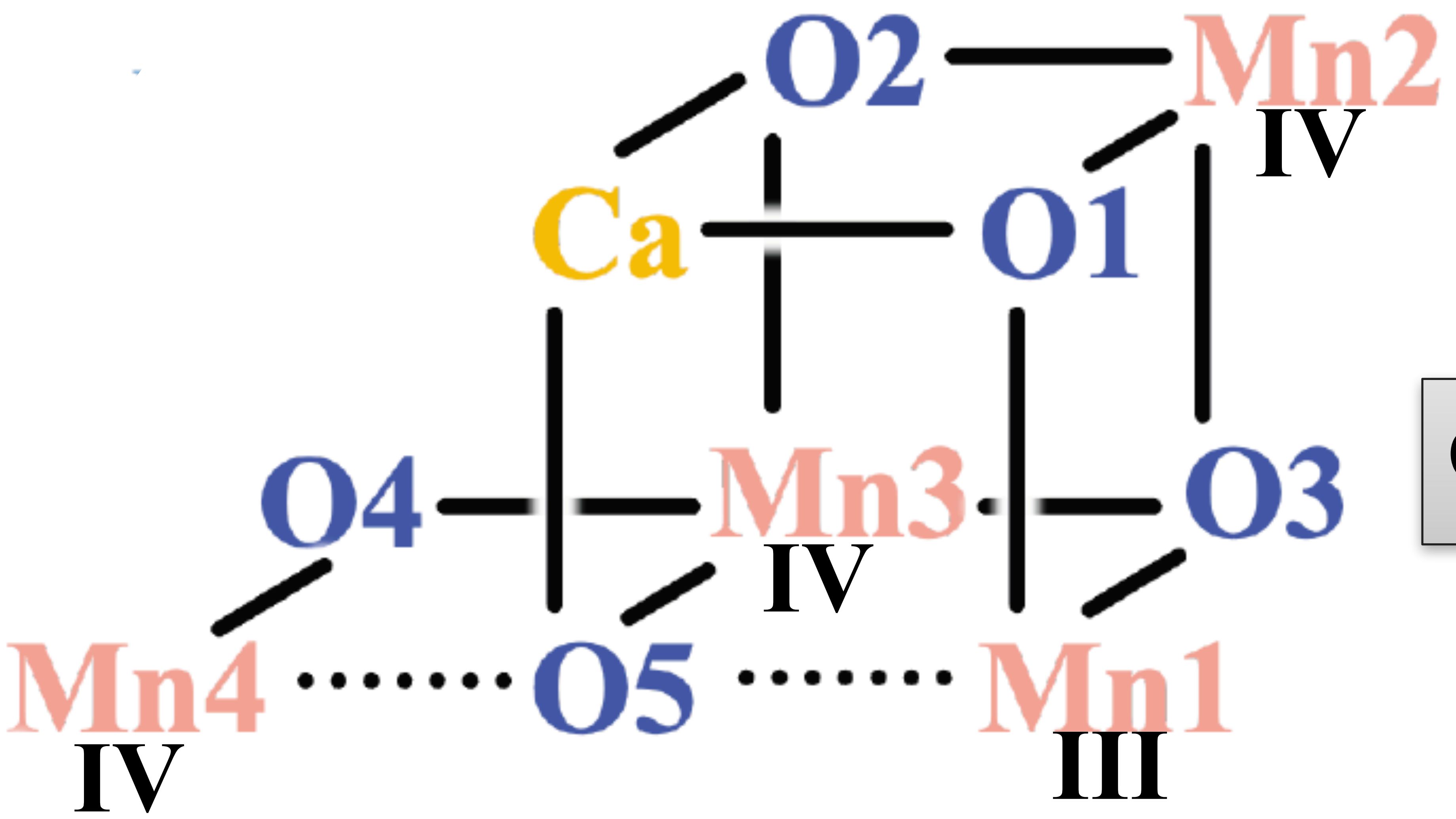


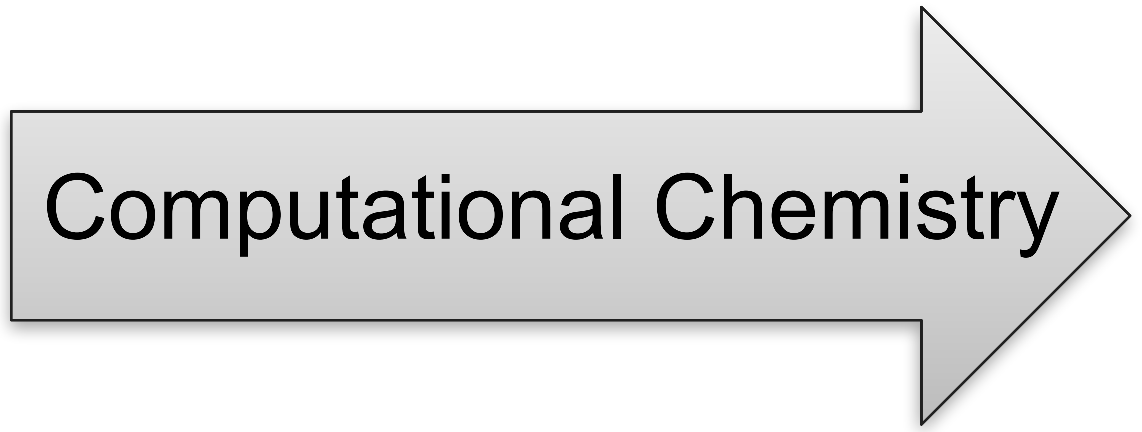
→ Mn2 has 3 **active** electrons

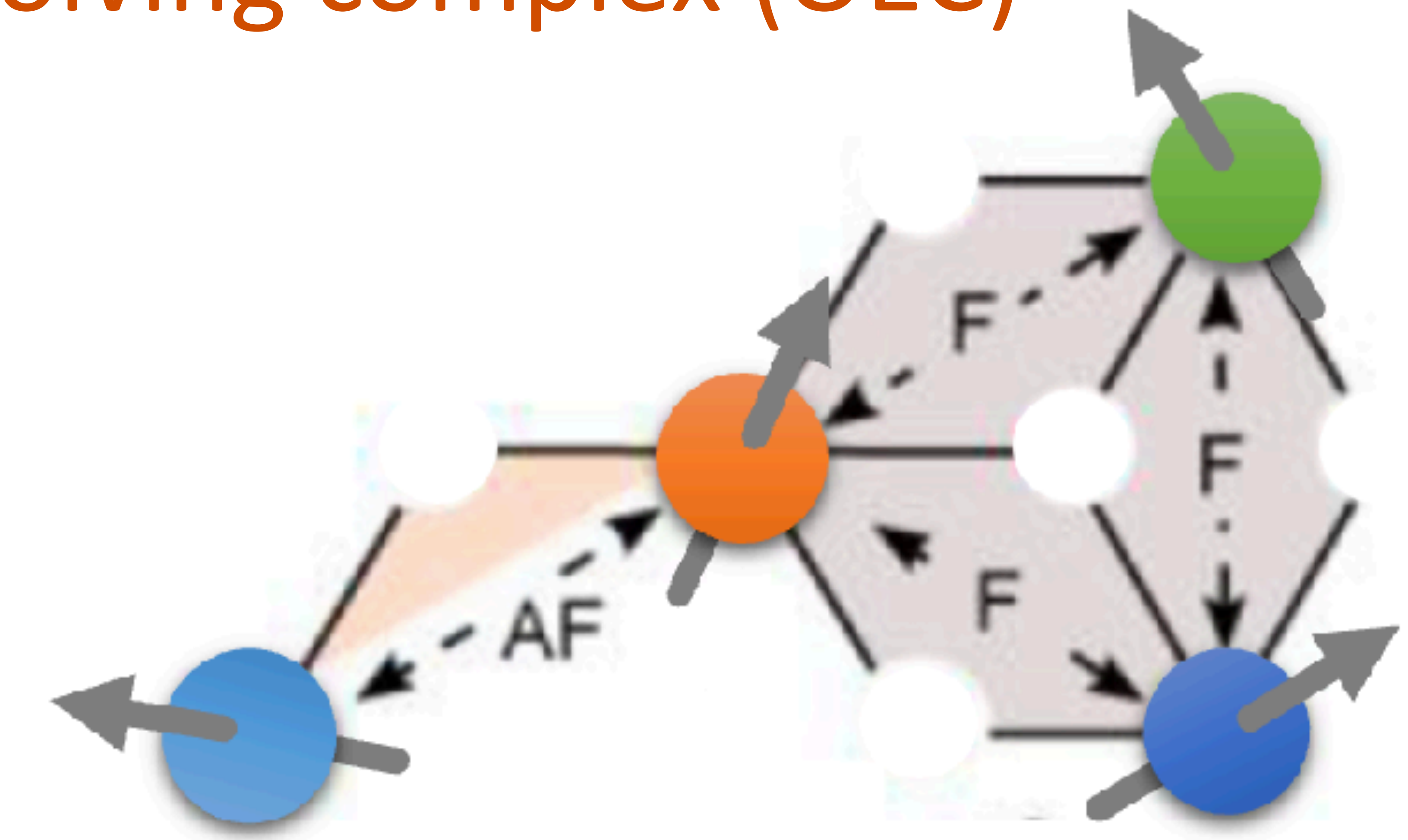
- Coulomb interaction - localizes electrons
- Exchange interaction - ferromagnetic
- Super-exchange interaction - anti-ferromagnetic

Model Hamiltonians

Example: Biochemical catalyst involved in the oxygen evolving complex (OEC)







→ Mn2 has 3 **active** electrons

Hilbert space scaling:

$\propto (2S+1)^N$

V. Krewald, M. Retegan, F. Neese, W. Lubitz,
D. A. Pantazis, N. Cox, Inorg. Chem. 55, 488–501 (2016)

Approach

- Represent high spins …

➡ How to implement high spins?

- … and let them interact

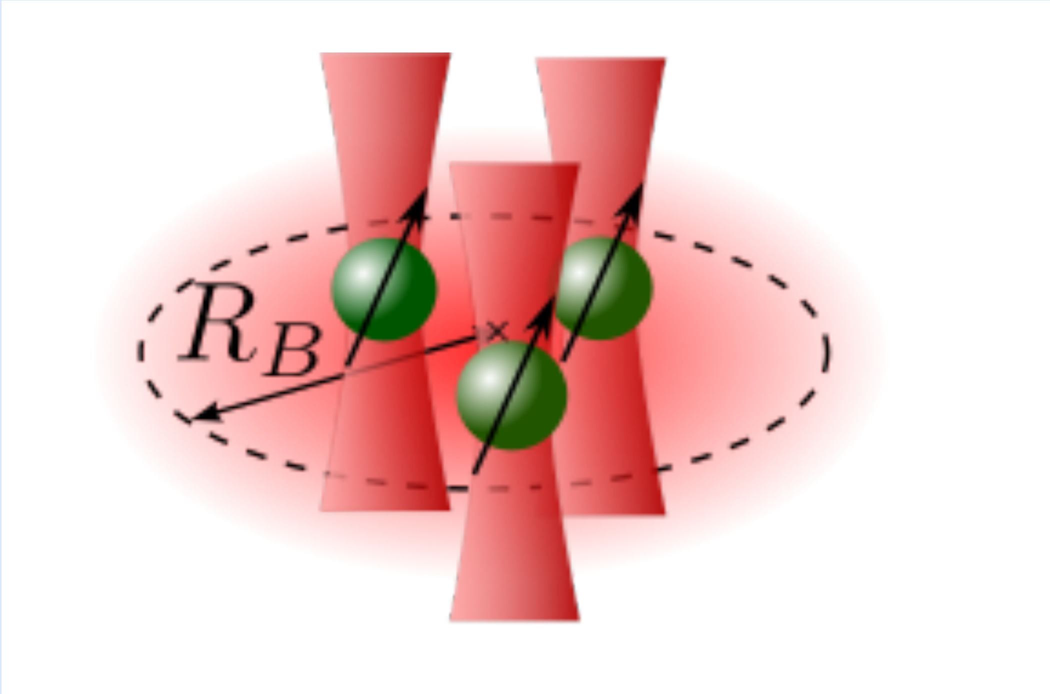
➡ How to implement non-local connectivity?

- Read out chemically relevant quantities

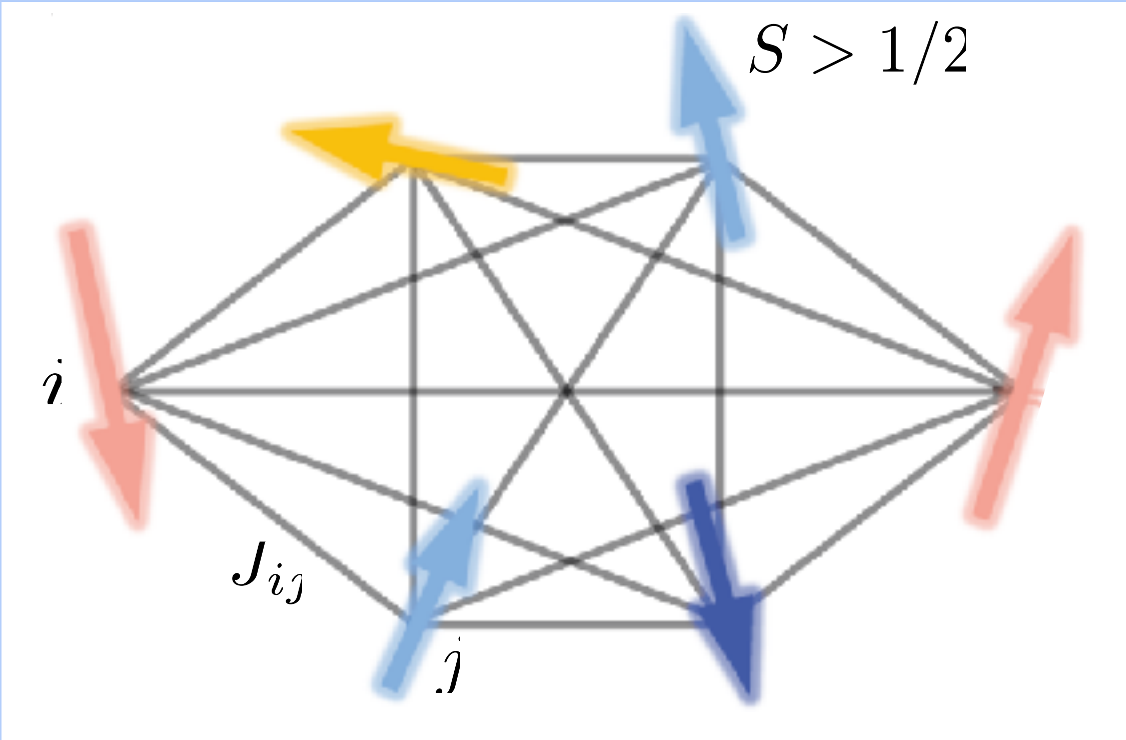
➡ Quantum-classical co-processing

Necessary Ingredients

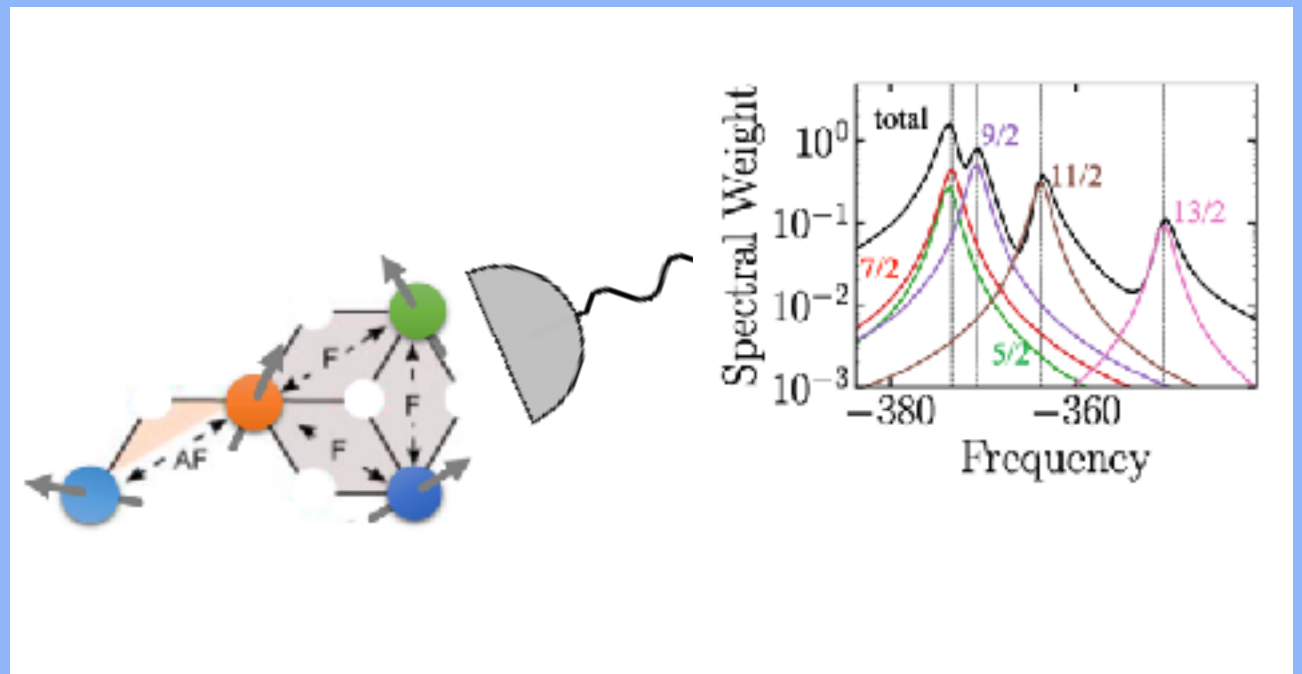
Control high spin (S>1/2)



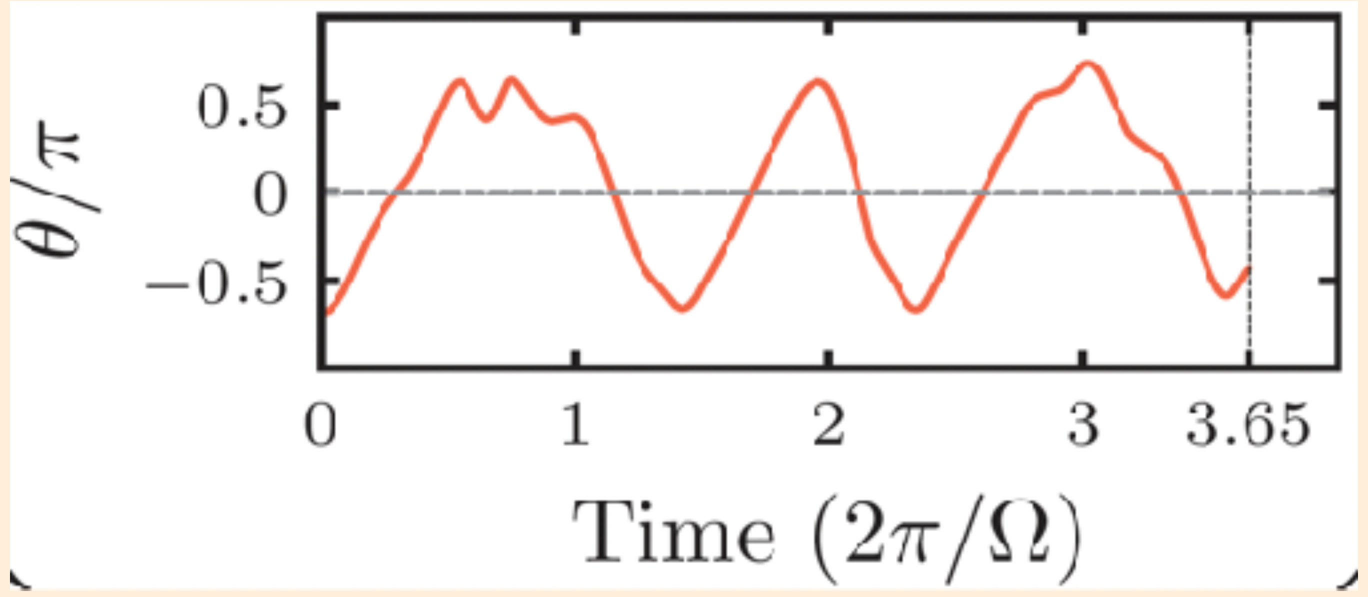
Non-local connectivity



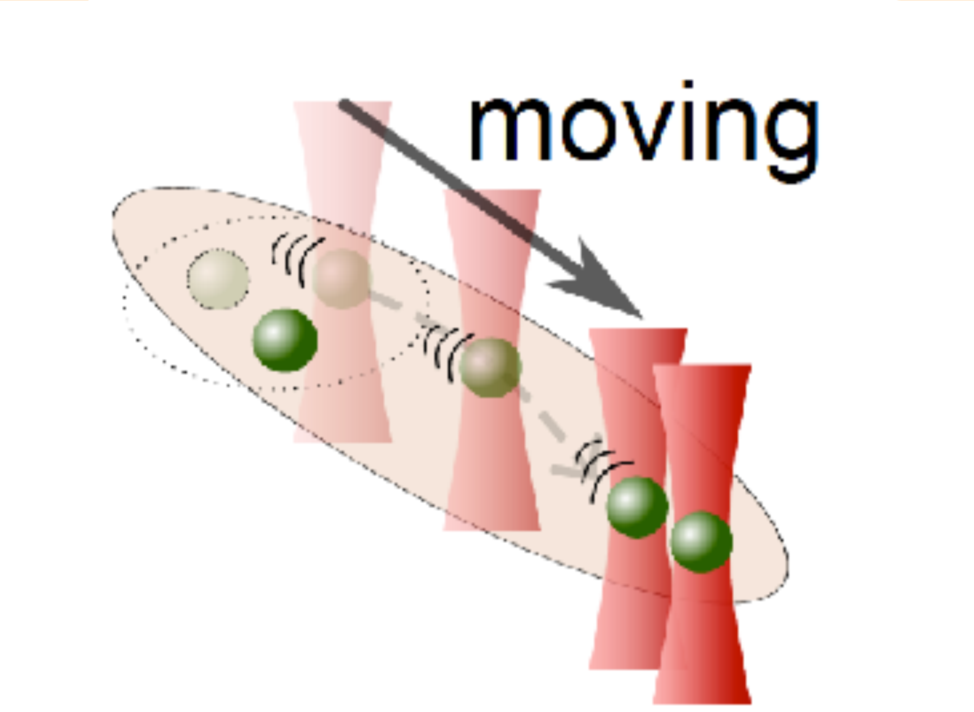
Solution read-out



Native multi-qubit gates

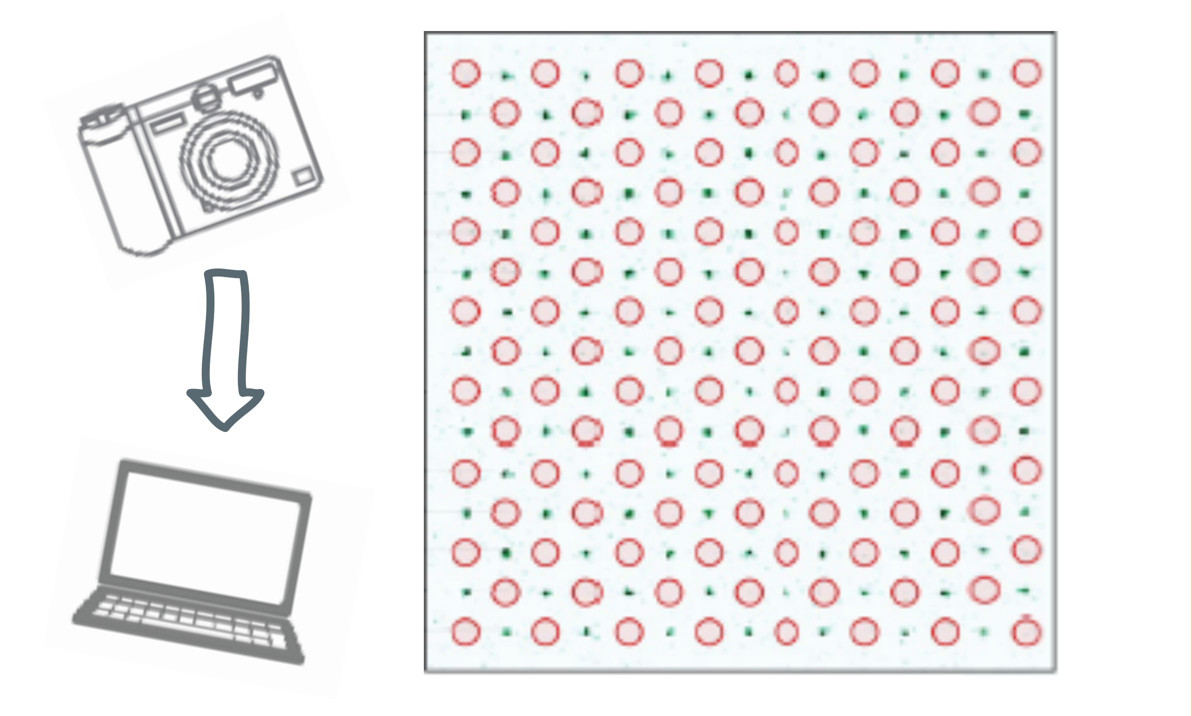


Atom moving



D. Bluvstein et al., Nature **604**, 451–456 (2022)

Co-processing

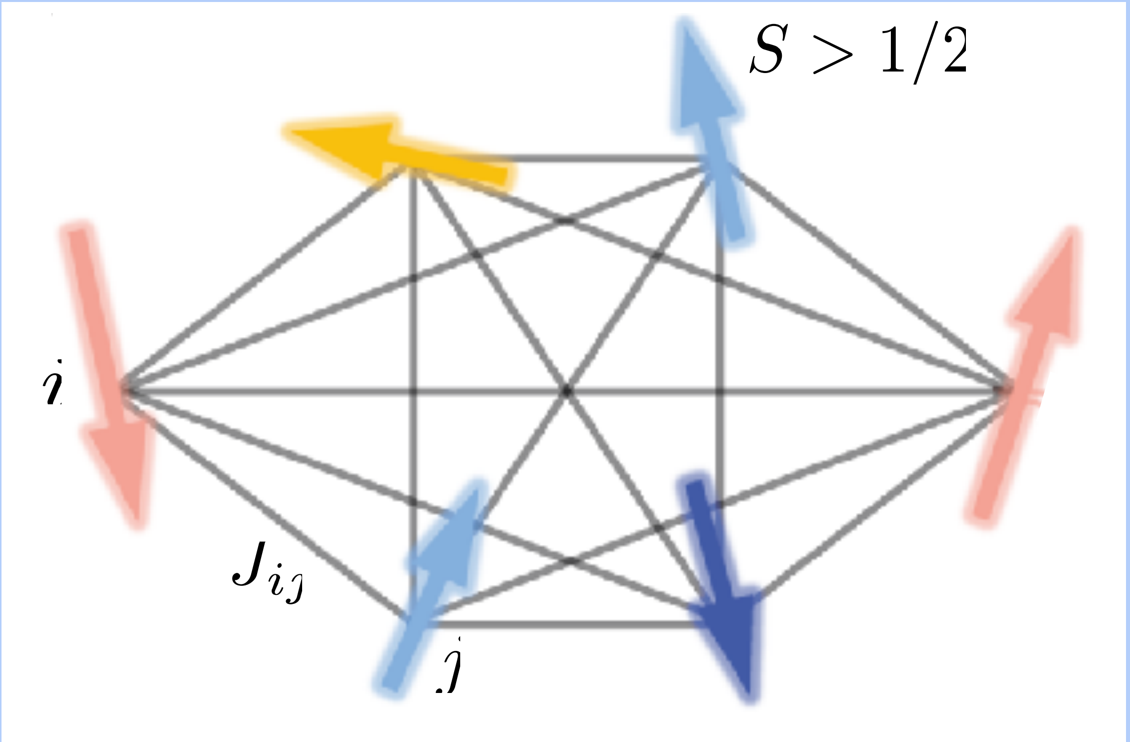


H.-Y. Huang et al., Nat. Phys. 16, 1050–1057 (2020)

Necessary Ingredients

Control high spin (S>1/2)

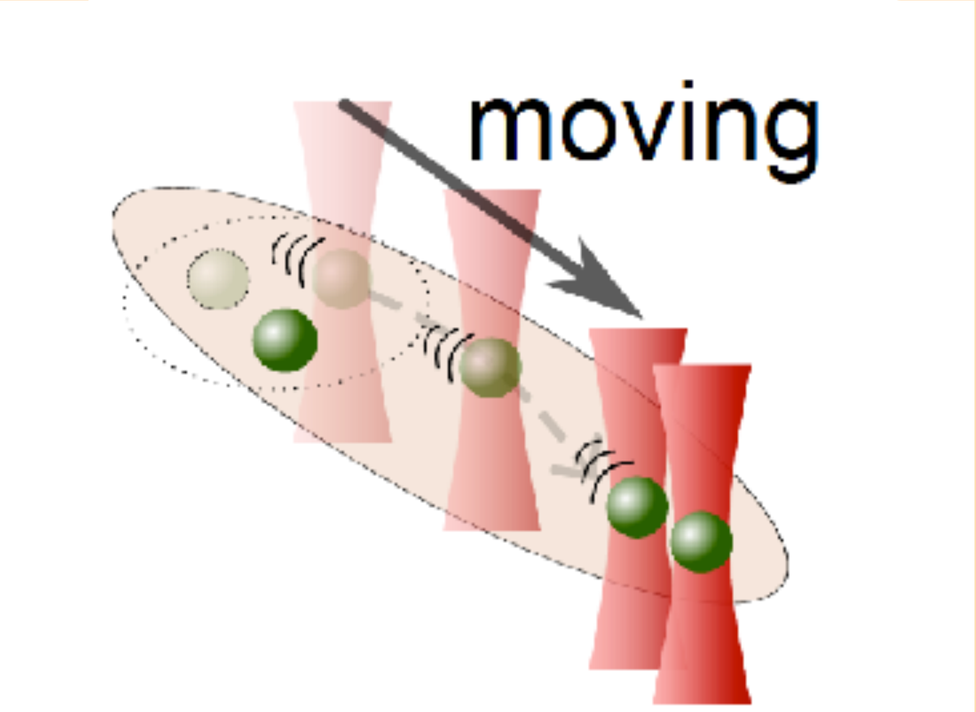
Non-local connectivity



Solution read-out

Native multi-qubit gates

Atom moving



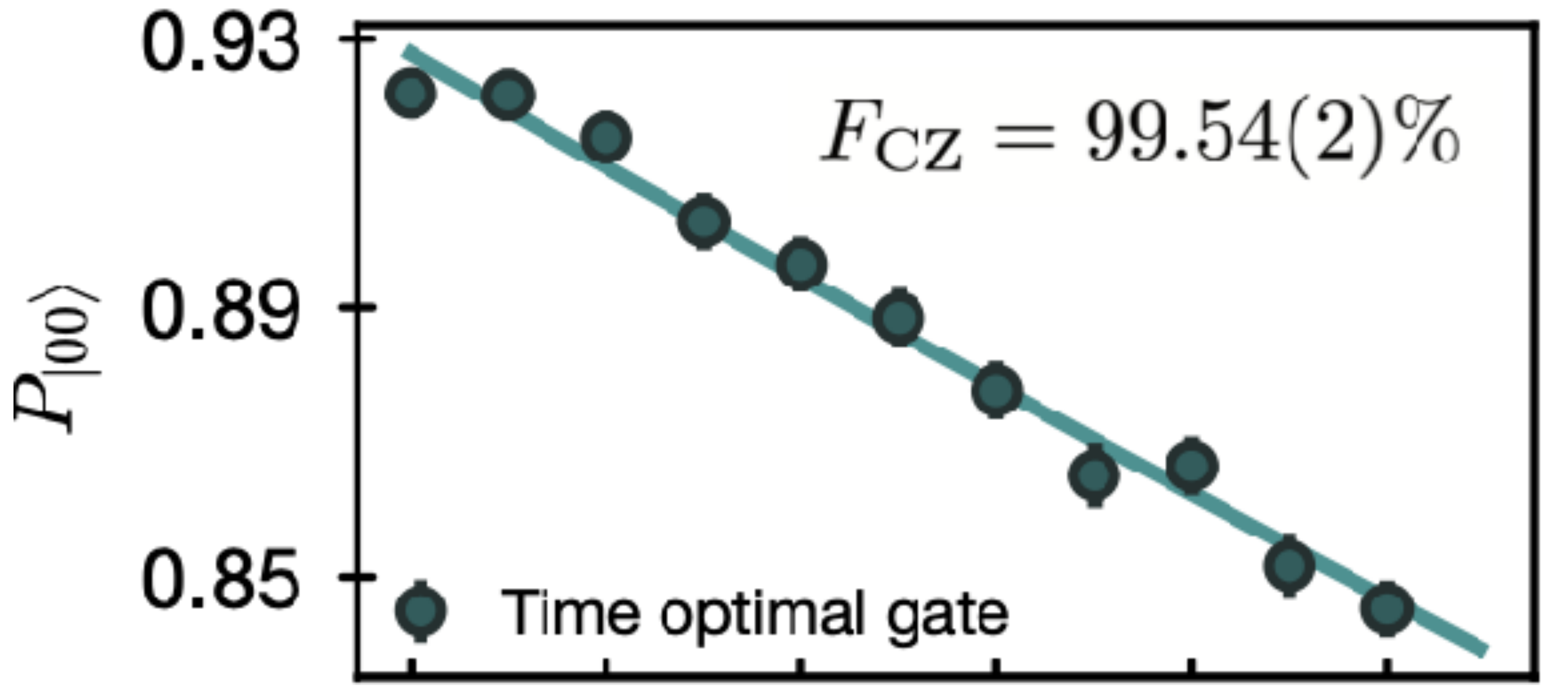
D. Bluvstein et al., Nature **604**, 451–456 (2022)

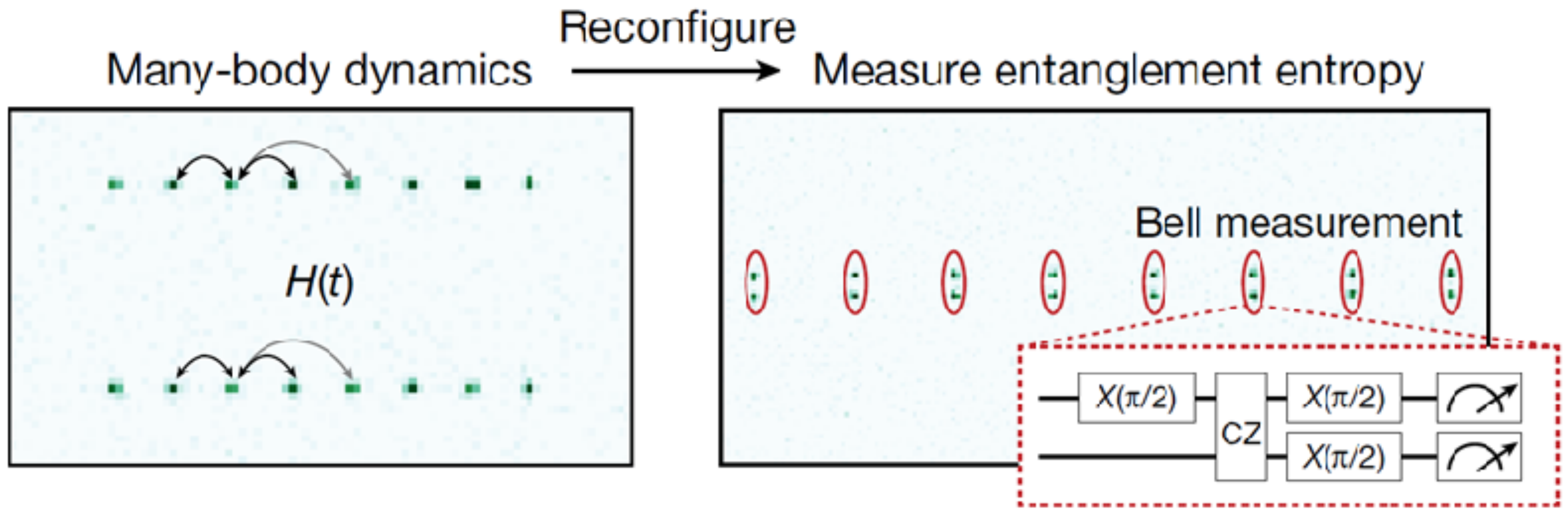
Co-processing

H.-Y. Huang et al., Nat. Phys. 16, 1050–1057 (2020)

Atom Array Platform in Analog-Digital Mode

Reconfigurable architecture + High-fidelity gates + Hybrid analog-digital control





High degree of programmability

Interactions manipulated via geometric configuration

+

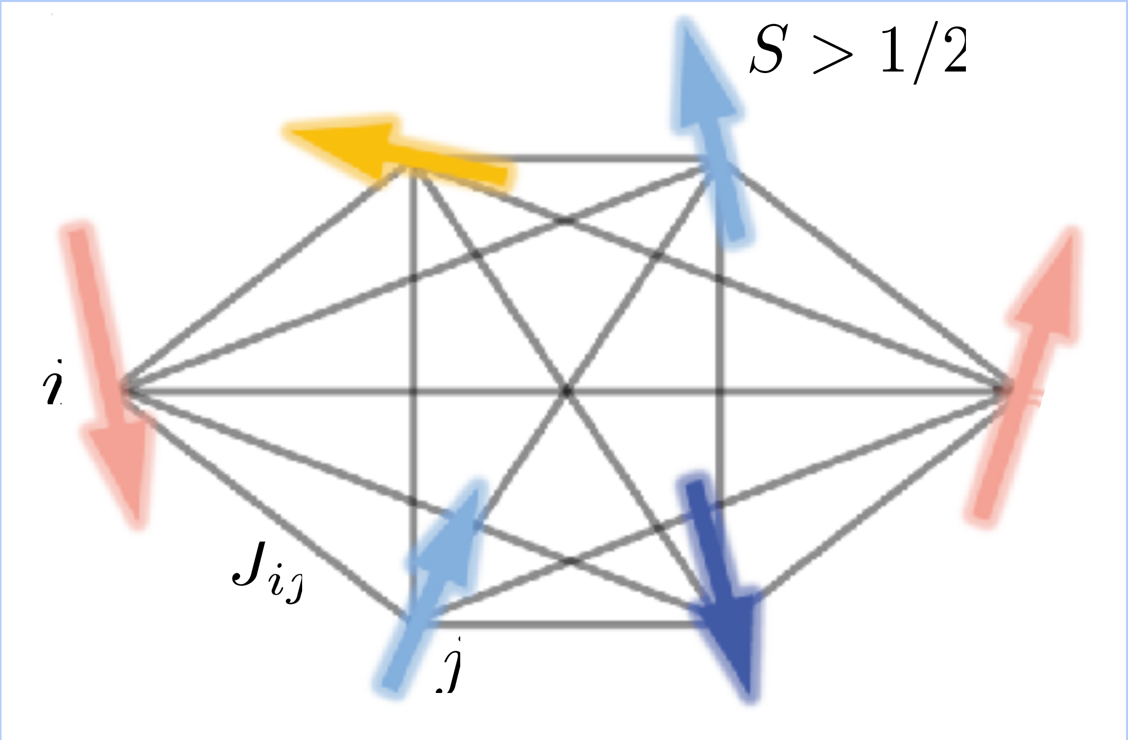
Global control pulses

Bluvstein *et al. Nature* **604**, 451 (2022).
Evered, *et al. Nature* **622**, 268n (2023)

Necessary Ingredients

Control high spin (S>1/2)

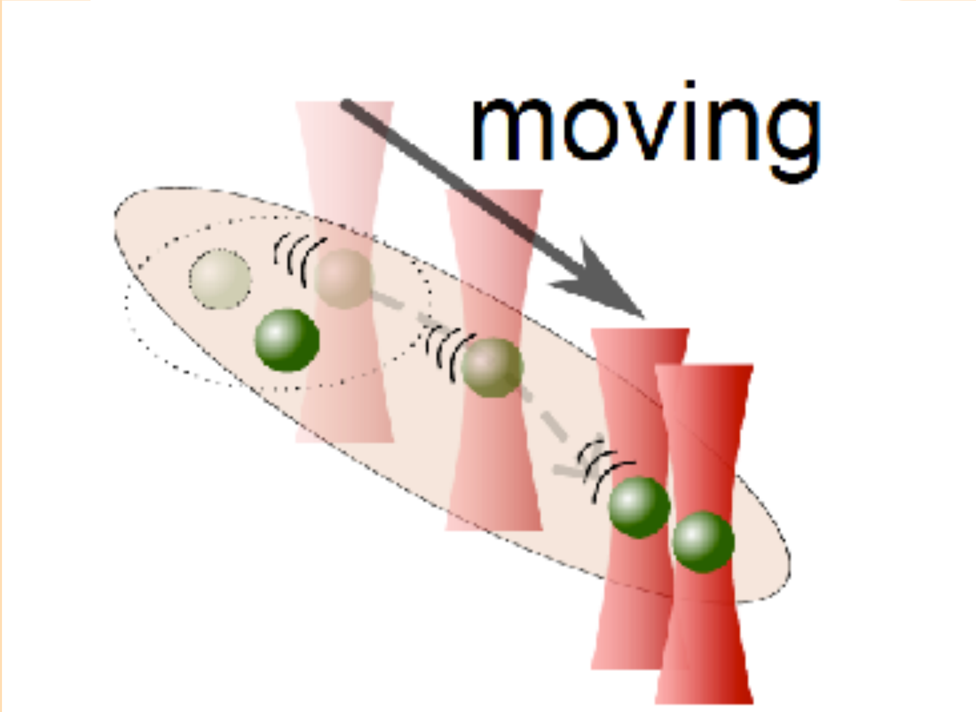
Non-local connectivity



Solution read-out

Native multi-qubit gates

Atom moving



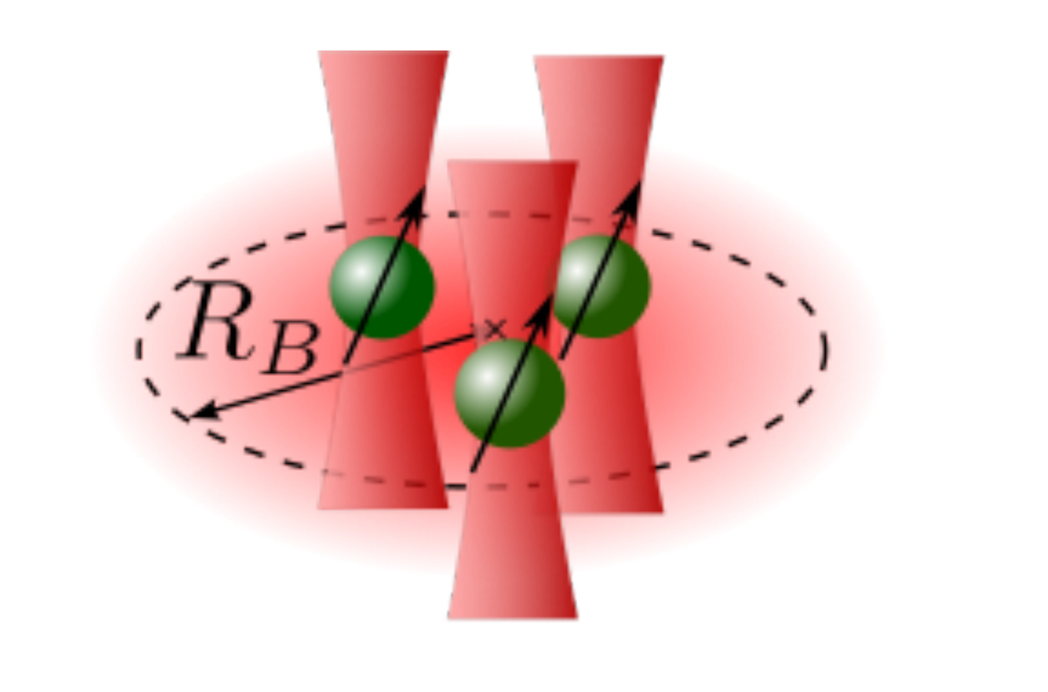
D. Bluvstein et al., Nature **604**, 451–456 (2022)

Co-processing

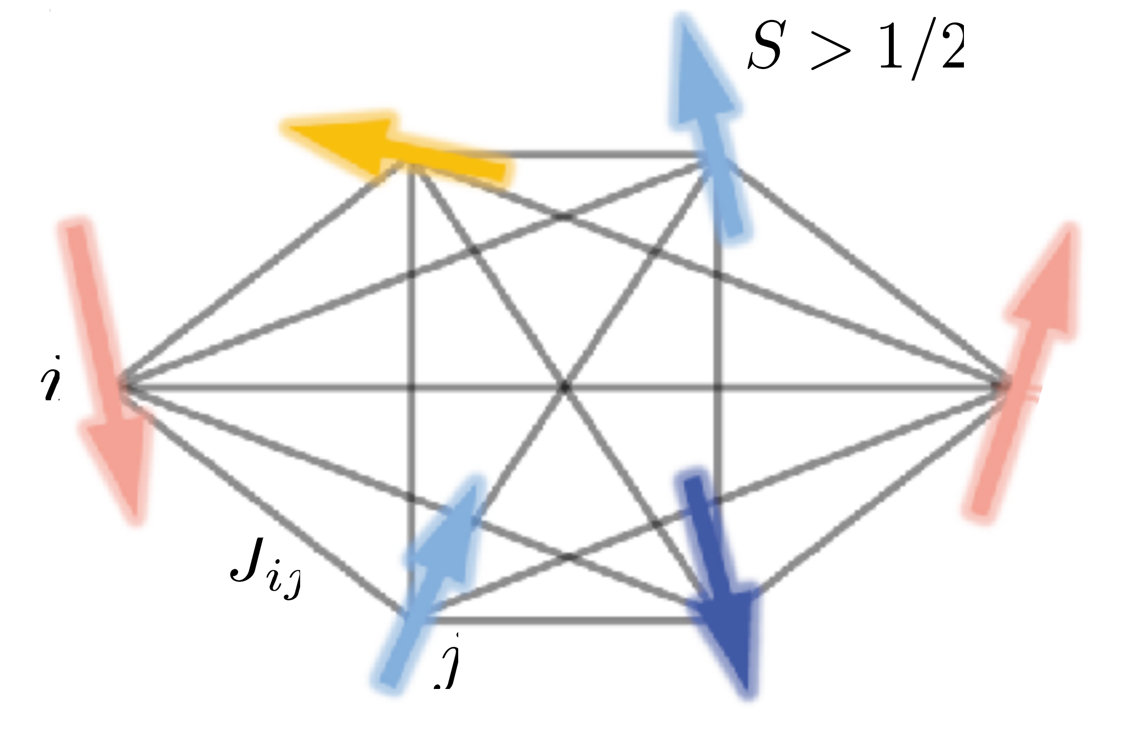
H.-Y. Huang et al., Nat. Phys. 16, 1050–1057 (2020)

Necessary Ingredients

Control high spin (S>1/2)

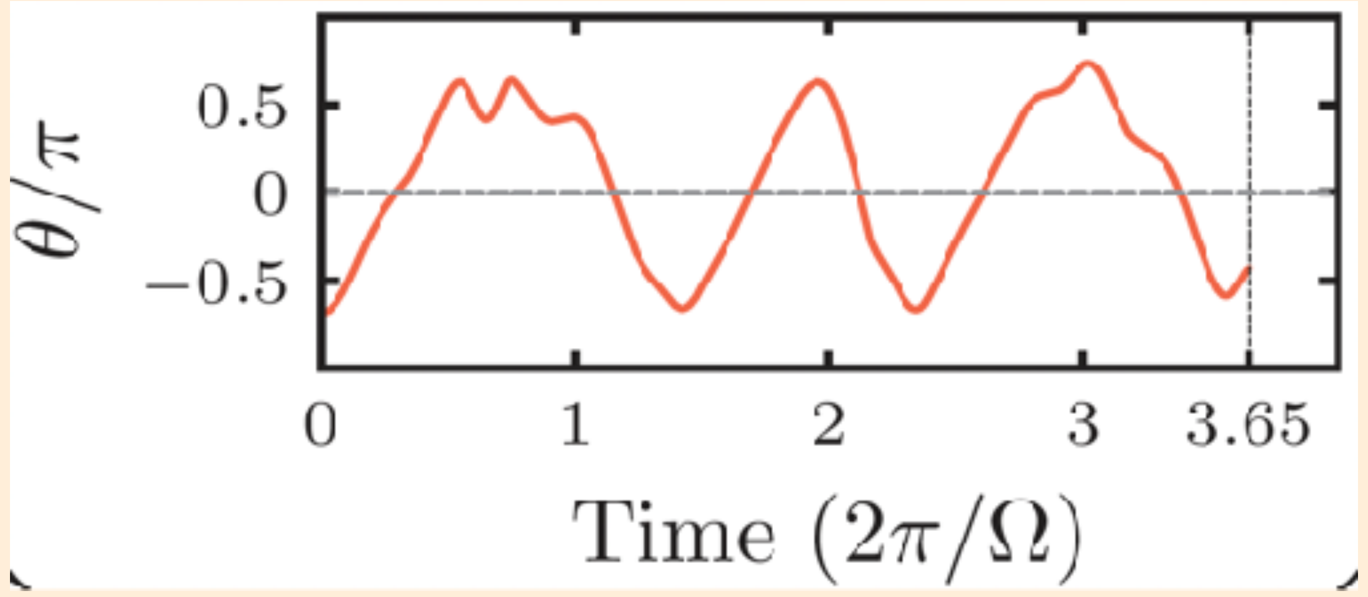


Non-local connectivity

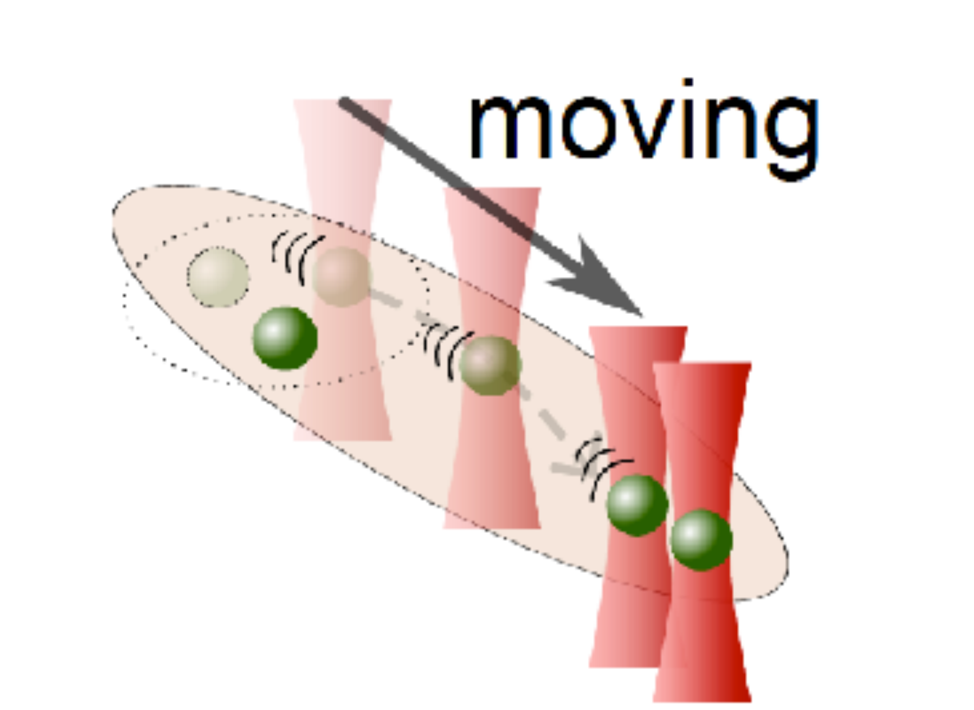


Solution read-out

Native multi-qubit gates



Atom moving



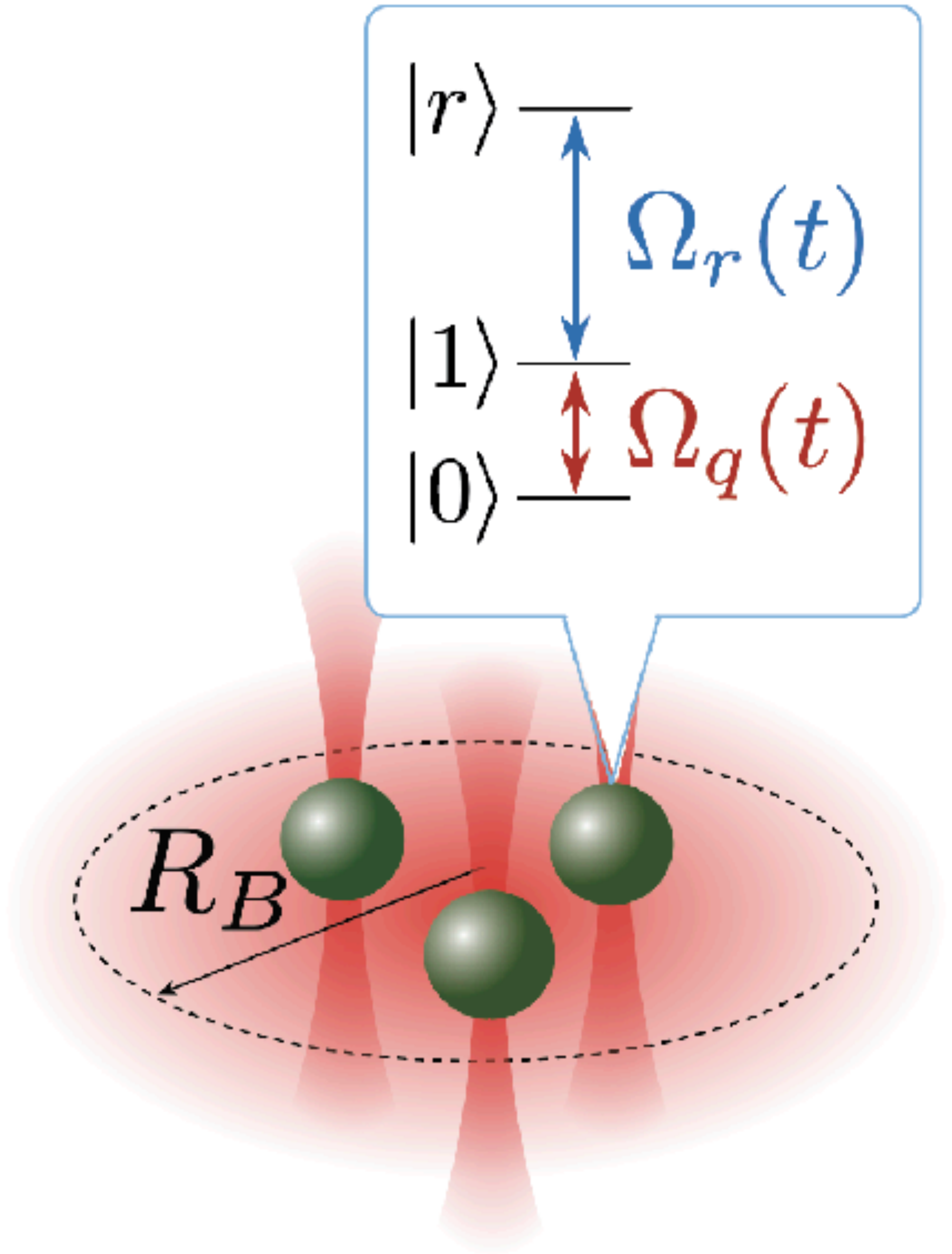
D. Bluvstein et al., Nature **604**, 451–456 (2022)

Co-processing

H.-Y. Huang et al., Nat. Phys. 16, 1050–1057 (2020)

Engineering Large Spins on Rydberg Platform

Hardware Efficient Multi-Qubit Operations with a global drive



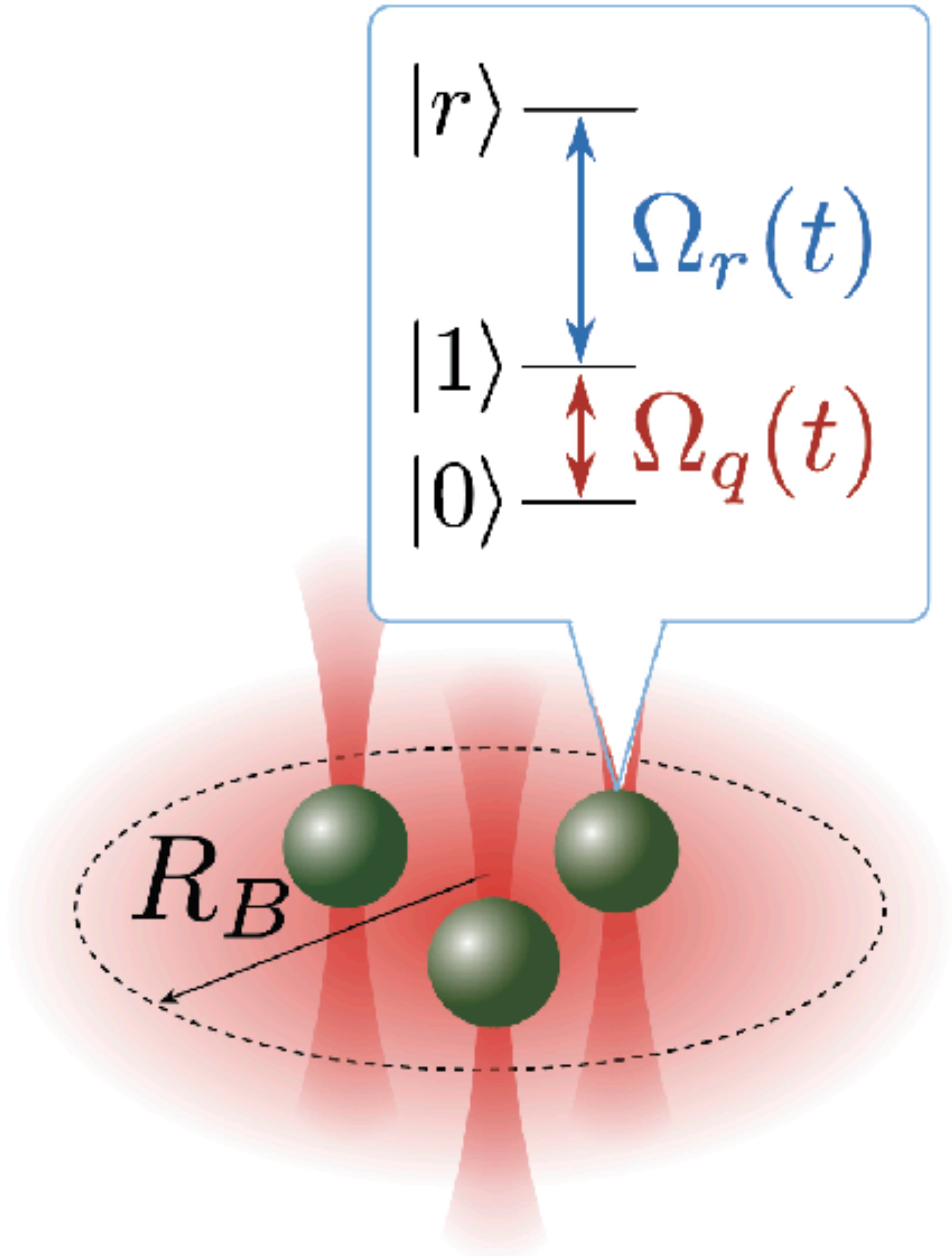
Encode spin-S variables into 2S (spin-1/2) qubits ("clusters"):
$\Rightarrow$ valid spin-S states: $\langle \hat{\mathbf{S}}_i^2 \rangle = S_i(S_i + 1)$

Engineer two-field pulses (using Rydberg blockade) to implement any 2S-qubit gate!

Realize multi-qubit gates via time-dependent global control:
use GrAPE (Gradient Ascent Pulse Engineering)

Engineering Large Spins on Rydberg Platform

Hardware Efficient Multi-Qubit Operations with a global drive



Encode spin-S variables into 2S (spin-1/2) qubits (“clusters”):

$\Rightarrow$ valid spin-S states: $\langle \hat{S}_i^2 \rangle = S_i(S_i + 1)$

Engineer two-field pulses (using Rydberg blockade) to implement any 2S-qubit gate!

Khaneja, et al., J. Magn. Reson. **172**, 296 (2005)
Jandura et al., Quantum **6**, 712 (2022)
Evered et al., Nature **622**, 268 (2023)
Katz, et al., Nat. Phys. **19**, 1452 (2023)

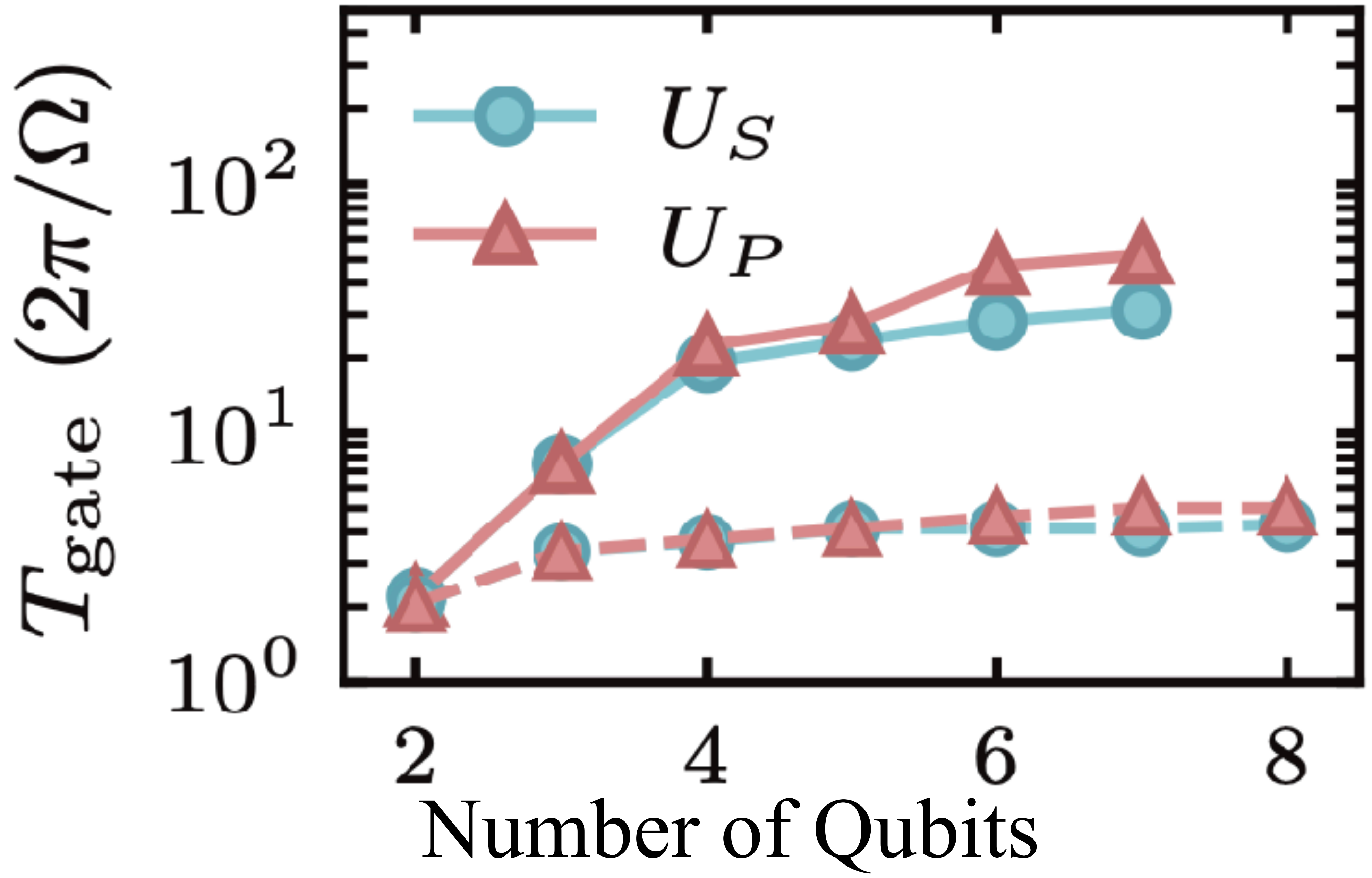
Realize multi-qubit gates via time-depende[...]
use GrAPE (Gradient Ascent Pulse E[...]

Important Metric: Gate Times

The shorter the gates, the more sequences one can run (until system decoheres)

Multi-Qubit Gates via Global Drive

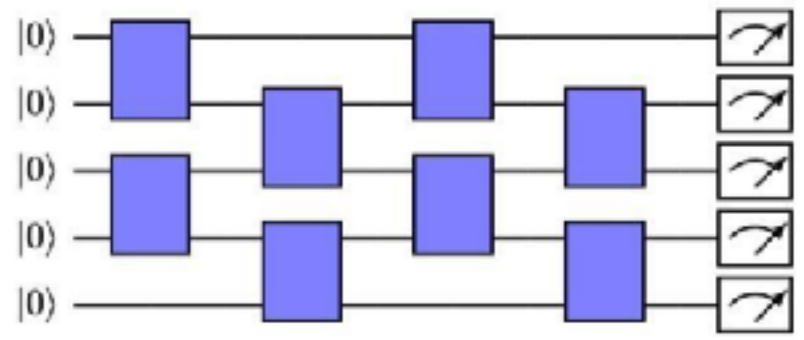
(for error $\epsilon = 10^{-3}$)

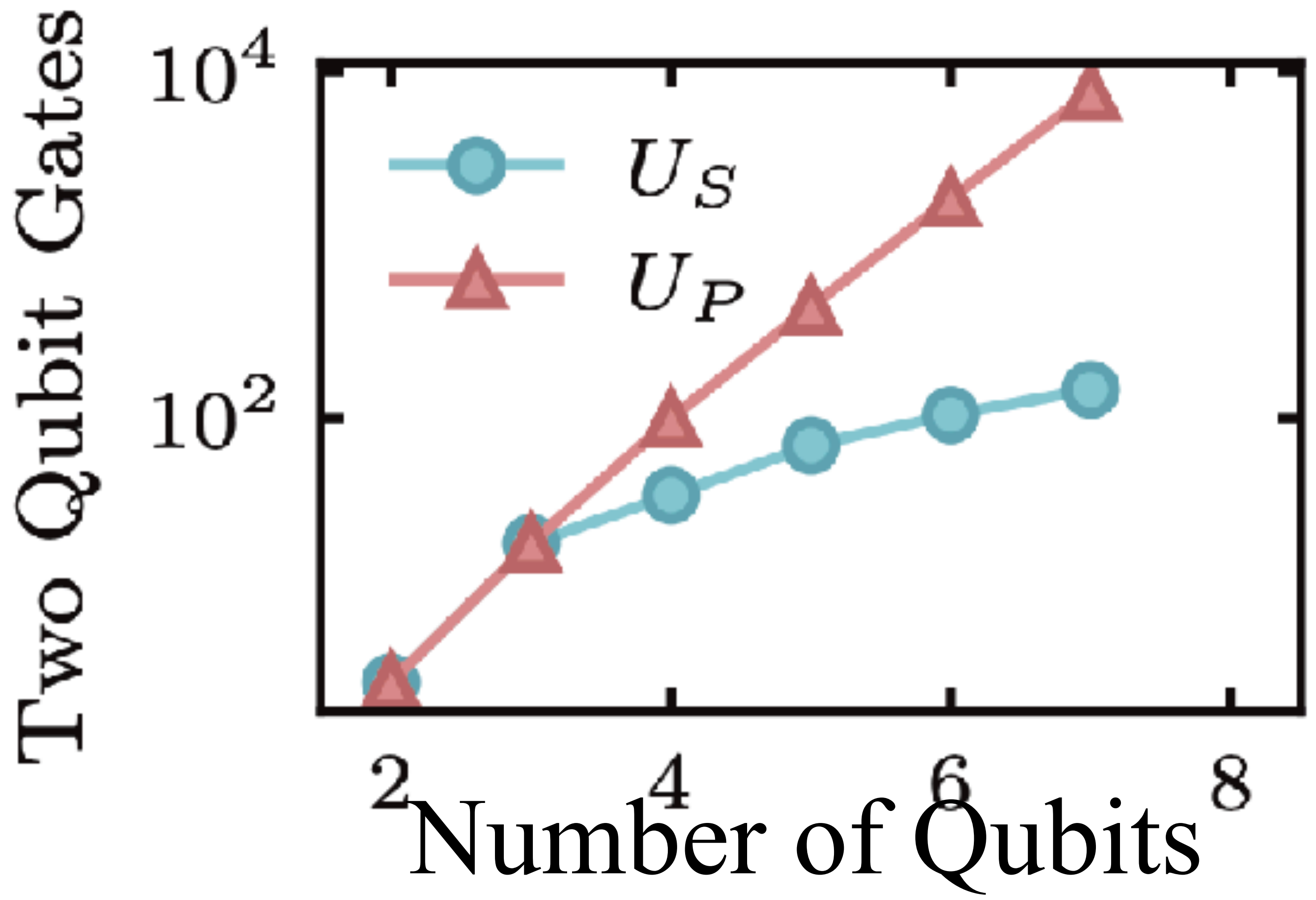


⇒ Almost no scaling with cluster size

Comparison: two-qubit operations

(for error $\epsilon = 10^{-3}$)



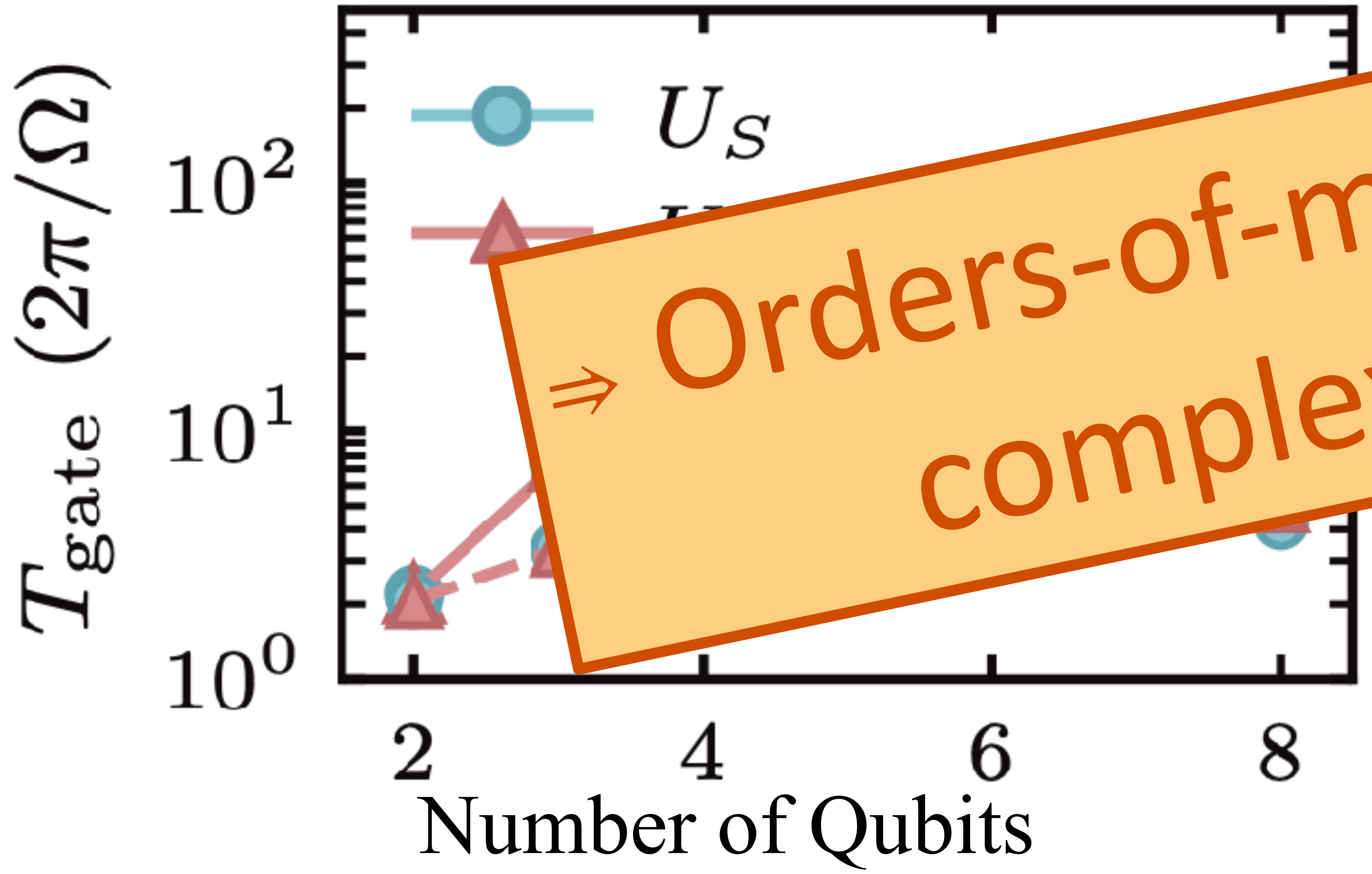


Important Metric: Gate Times

The shorter the gates, the more sequences one can run (until system decoheres)

Multi-Qubit Gates via Global Drive

(for error $\epsilon = 10^{-3}$)

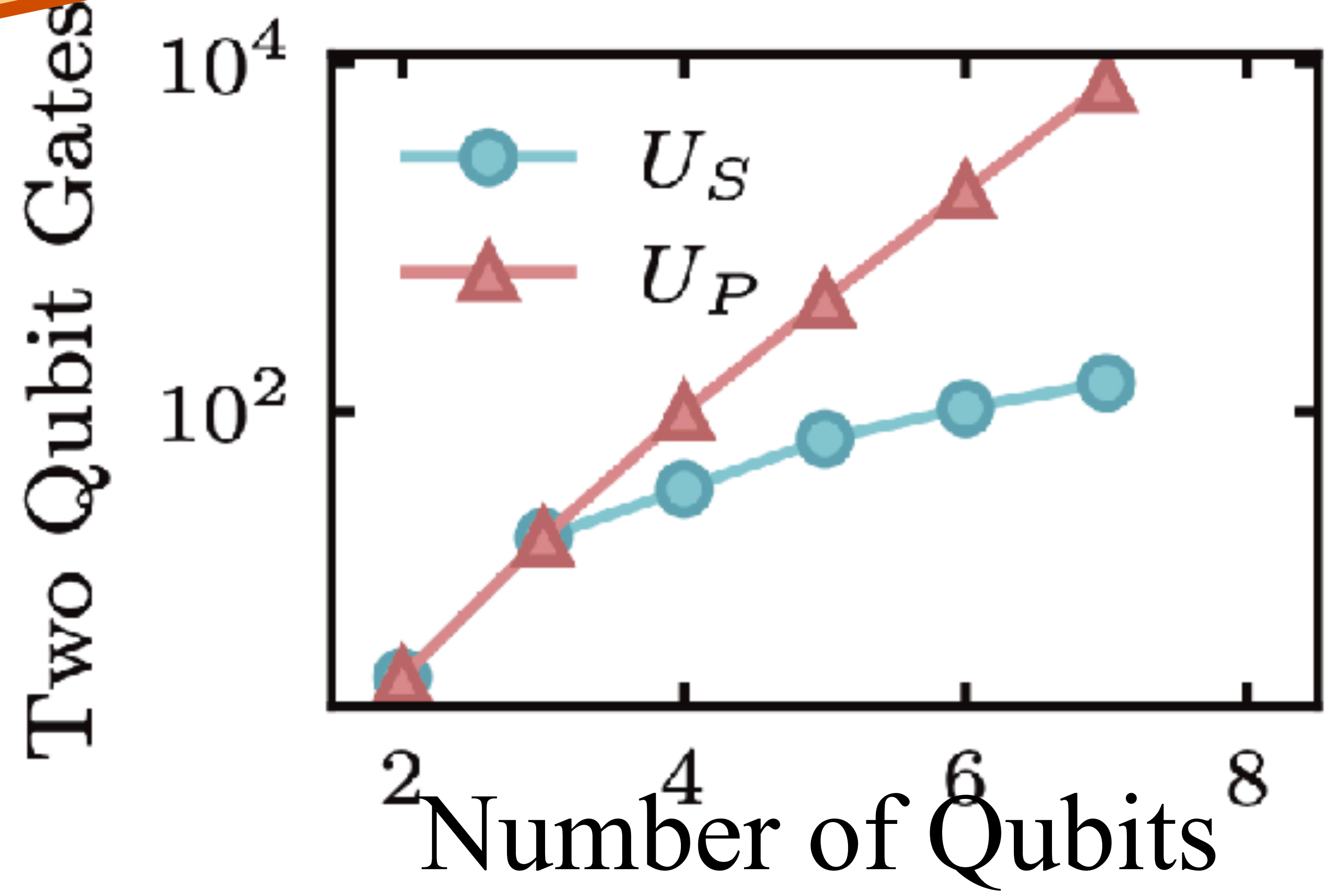


⇒ Almost no scaling with cluster size

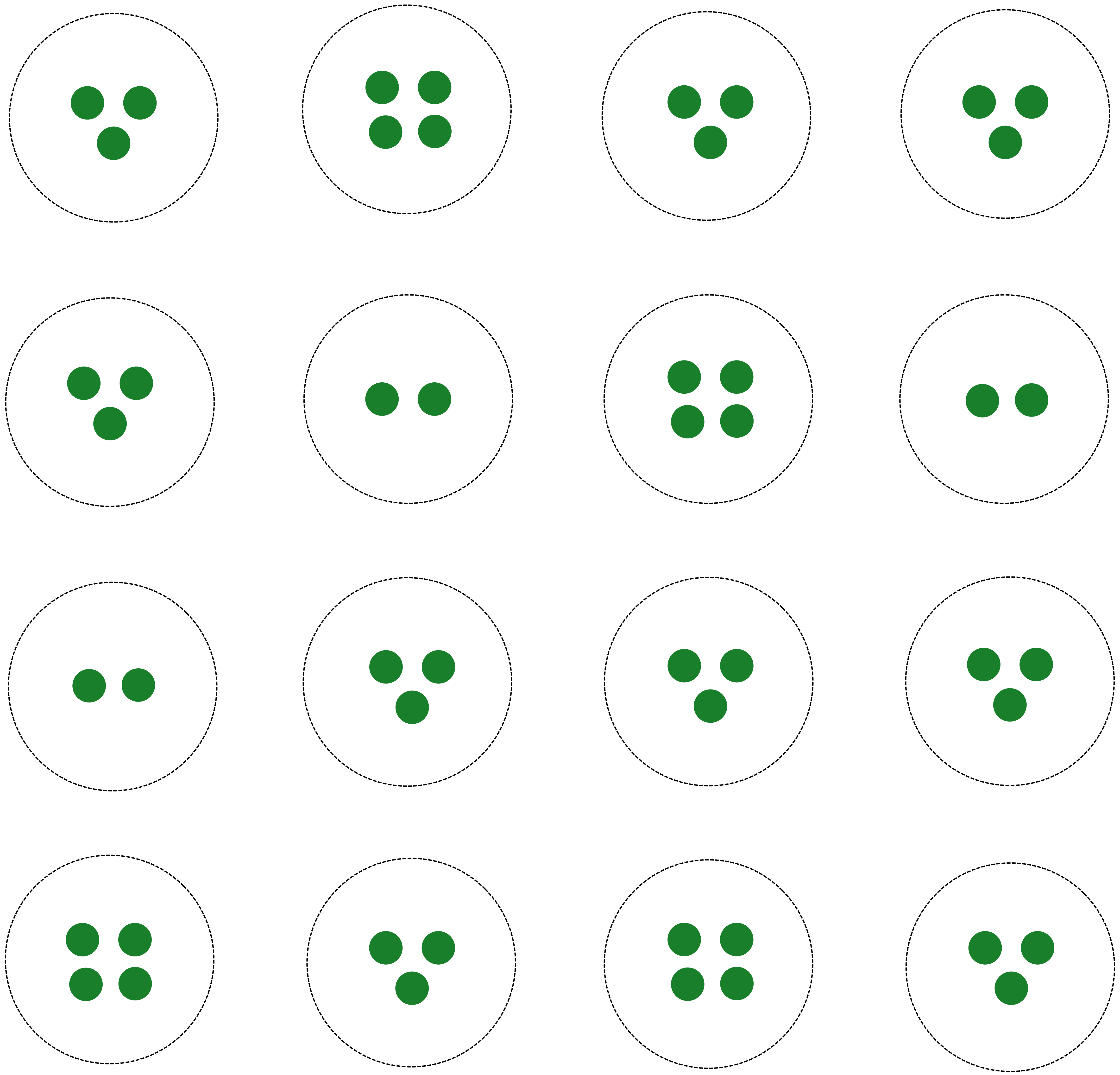
Comparison: two-qubit operations

(for error ϵ

⇒ Orders-of-magnitude speedup for complex spin-operations



Engineering Interactions



Target Hamiltonian:

$$H_{\mathrm{target}} = \sum_{\mathrm{i,j}} J_{\mathrm{ij}}^{\alpha\beta}\, \hat{S}_{\mathrm{i}}^{\alpha}\, \hat{S}_{\mathrm{j}}^{\beta}$$

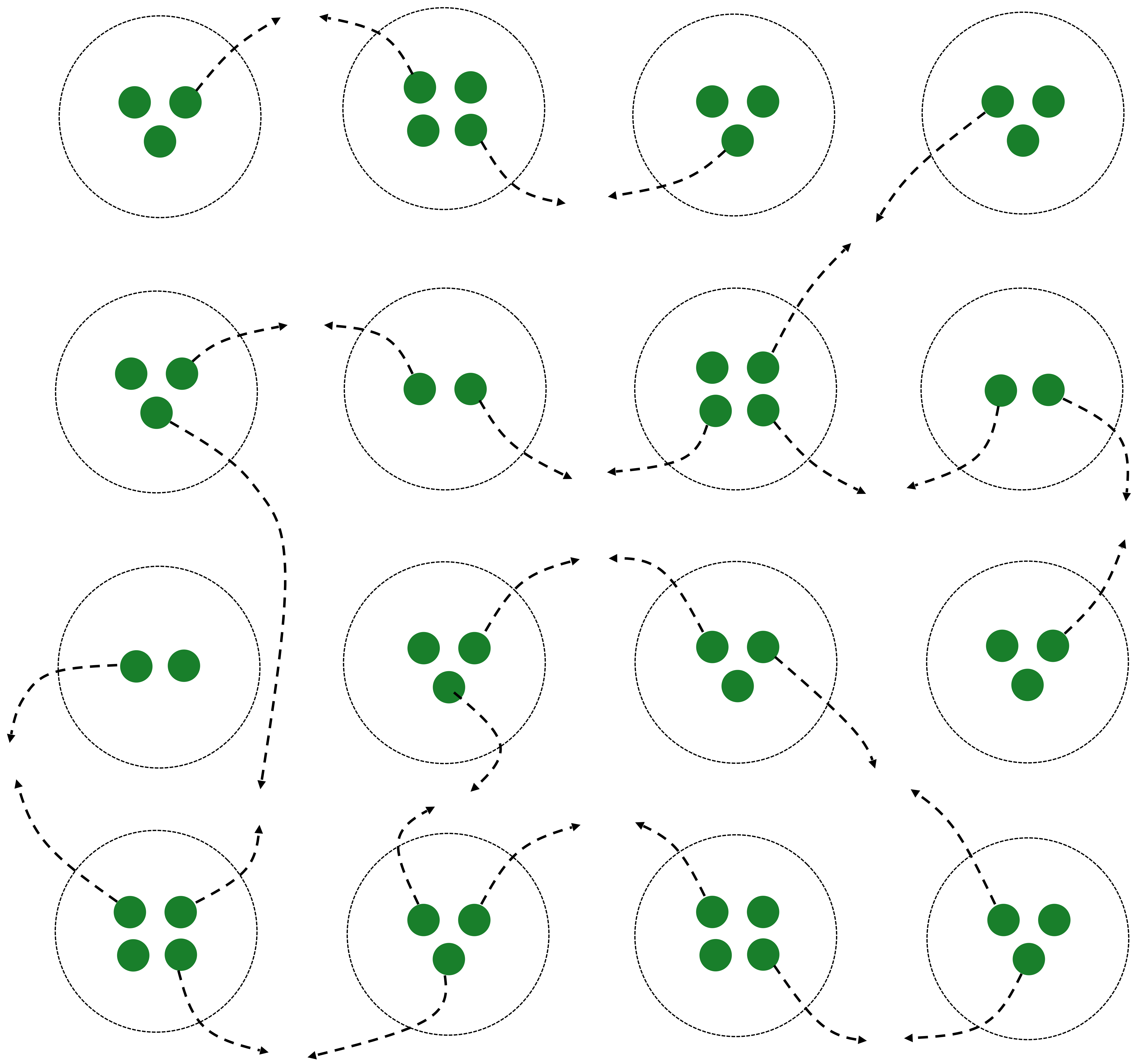
Highly programmable!

Example:

$$J_{\mathrm{ij}}^{\alpha\beta} = \underbrace{J_{\mathrm{ij}}\delta_{\alpha\beta}}_{\text{Heisenberg}} + \underbrace{D_{\mathrm{ij}}^{\gamma}\,\epsilon_{\alpha\beta}^{\gamma}}_{\text{DM}}$$

(e.g., Malrieu et al, Chem. Rev **114**, 429 (2014))

Engineering Interactions



Target Hamiltonian:

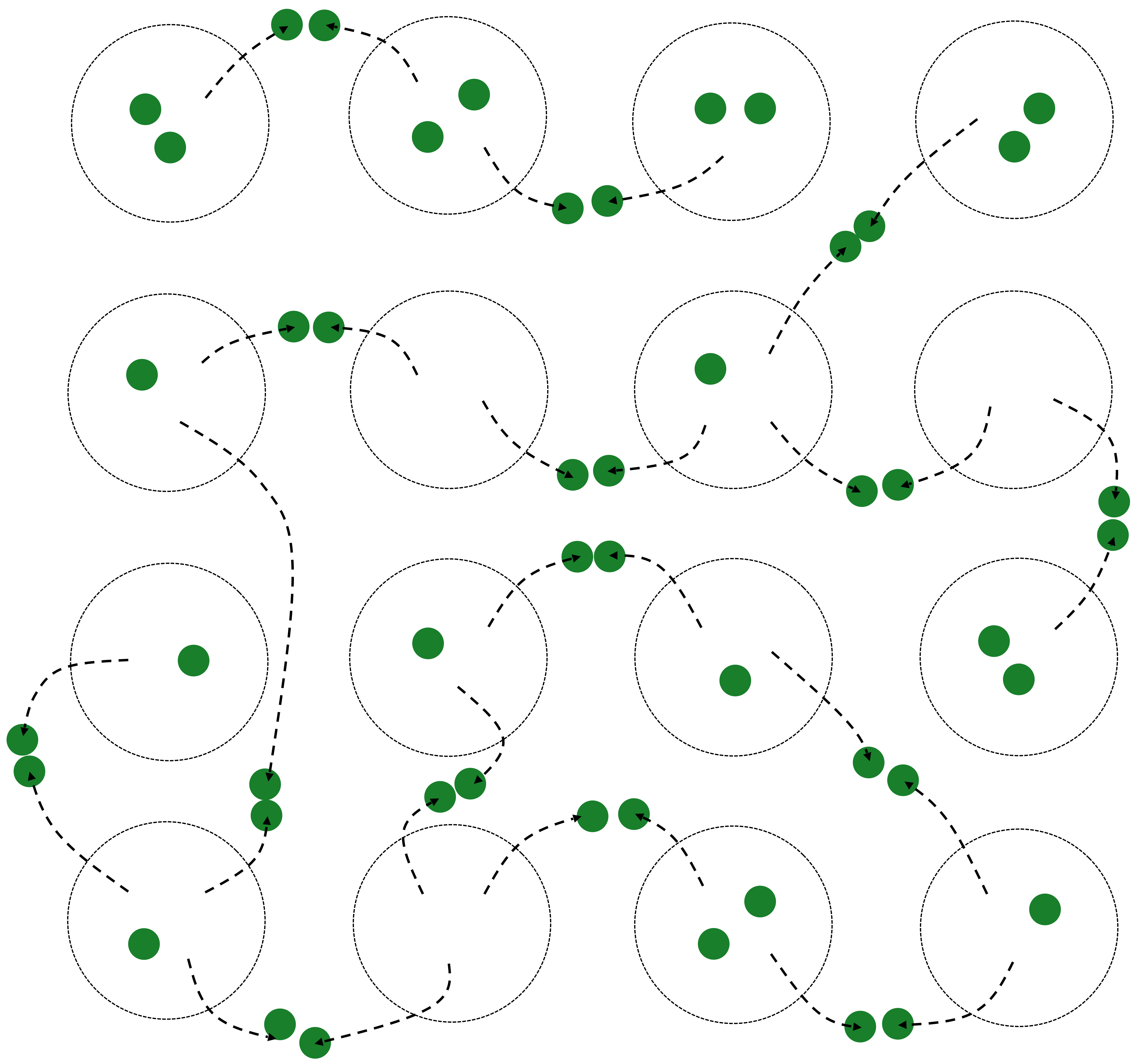
$$H_{target} = \sum_{i,j} J_{ij}^{\alpha\beta}\, \hat{S}_i^{\alpha}\, \hat{S}_j^{\beta}$$

1. Reconfigure

Engineering Interactions

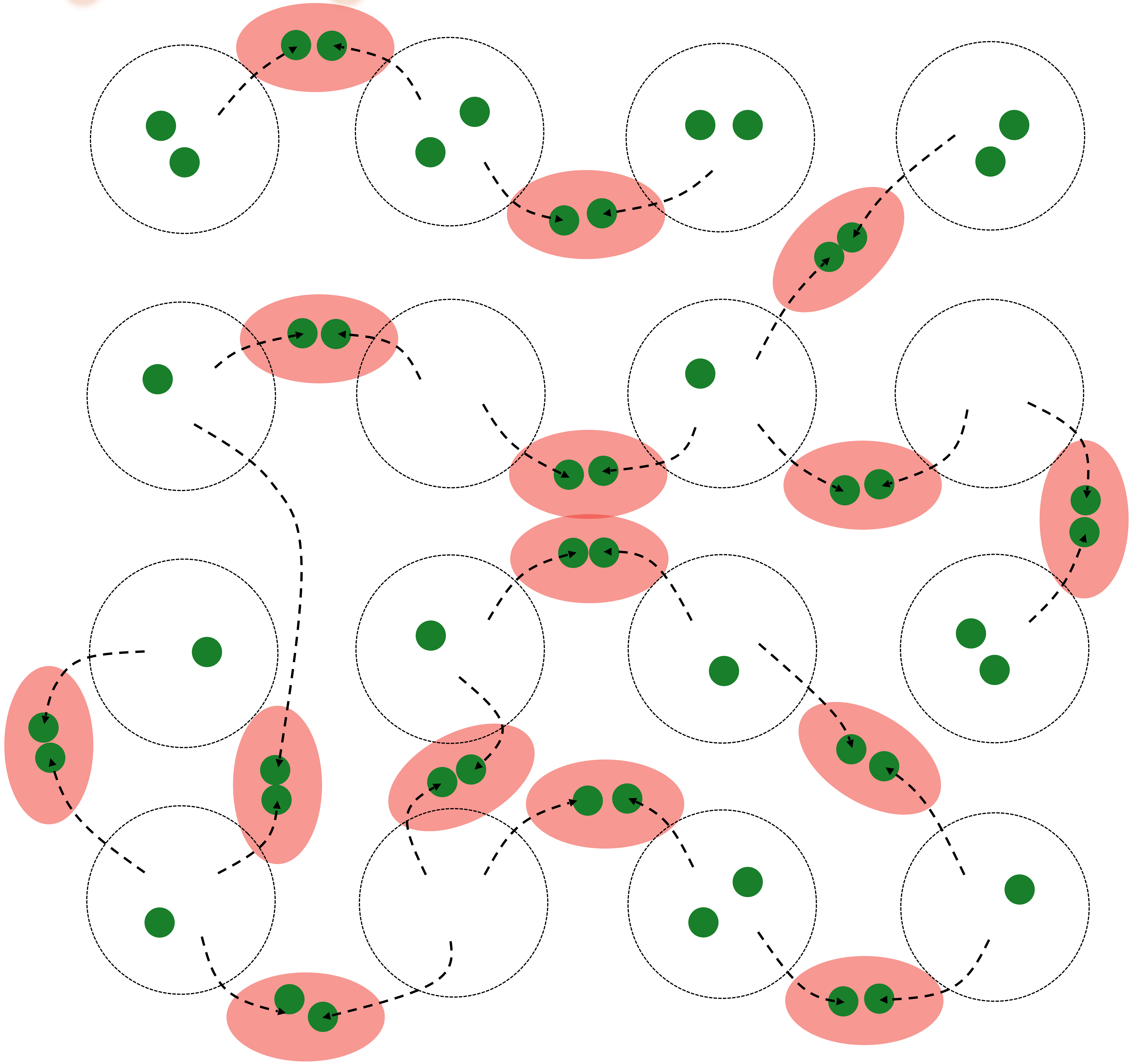
Target Hamiltonian:

$$H_{\mathrm{target}} = \sum_{i,j} J_{ij}^{\alpha\beta}\, \hat{S}_i^{\alpha}\, \hat{S}_j^{\beta}$$



1. Reconfigure

Engineering Interactions



Target Hamiltonian:

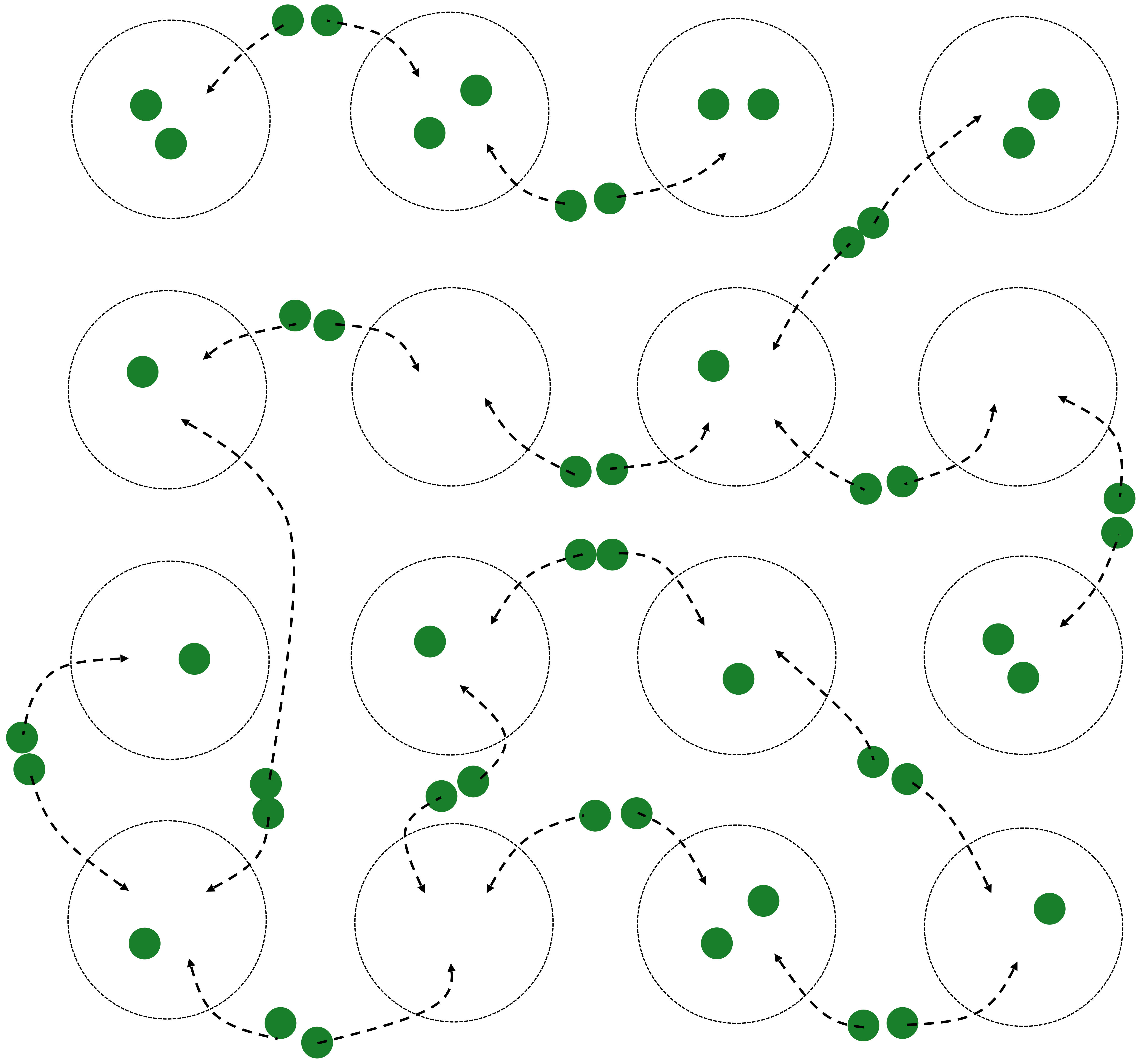
$$H_{target} = \sum_{i,j} J_{ij}^{\alpha\beta} \hat{S}_i^{\alpha} \hat{S}_j^{\beta}$$

1. Reconfigure

2. Inter-cluster gate: $H_I = \sum_{ij} J_{ij}^{\alpha\beta} \hat{s}_{i,a_{ij}}^{\alpha} \hat{s}_{j,b_{ij}}^{\beta}$

- Spin-1/2 gates and local rotations
- Mediates generic, long-range connectivity
- Violates large-spin encoding

Engineering Interactions



Target Hamiltonian:

$$H_{target} = \sum_{i,j} J_{ij}^{\alpha\beta} \hat{S}_i^{\alpha} \hat{S}_j^{\beta}$$

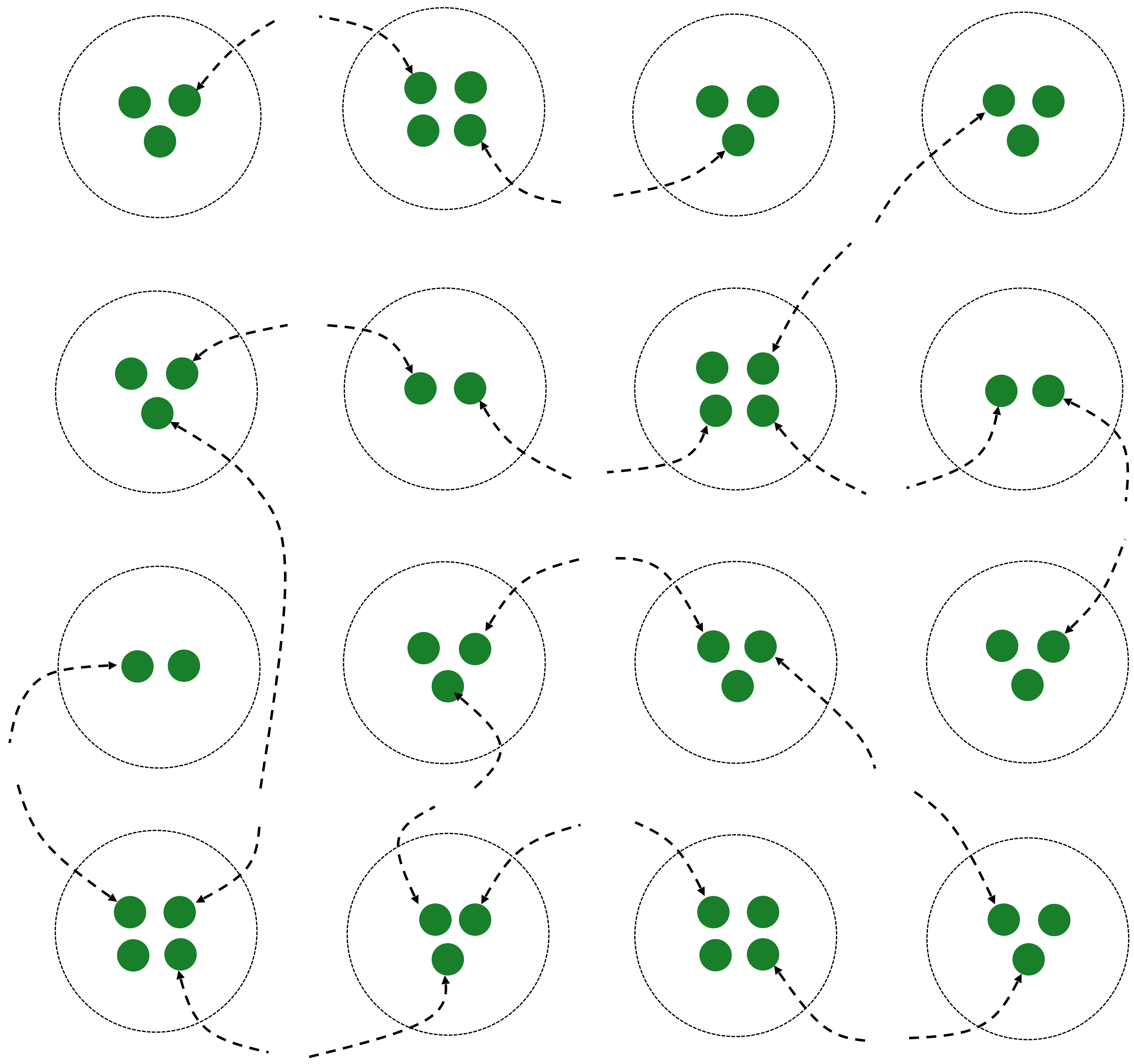
1. Reconfigure

2. Inter-cluster gate: $$H_I = \sum_{ij} J_{ij}^{\alpha\beta} \hat{s}_{i,a_{ij}}^{\alpha} \hat{s}_{j,b_{ij}}^{\beta}$$

- Spin-1/2 gates and local rotations
- Mediates generic, long-range connectivity
- Violates large-spin encoding

3. Reconfigure

Engineering Interactions



Target Hamiltonian:

$$H_{target} = \sum_{i,j} J_{ij}^{\alpha\beta} \hat{S}_i^{\alpha} \hat{S}_j^{\beta}$$

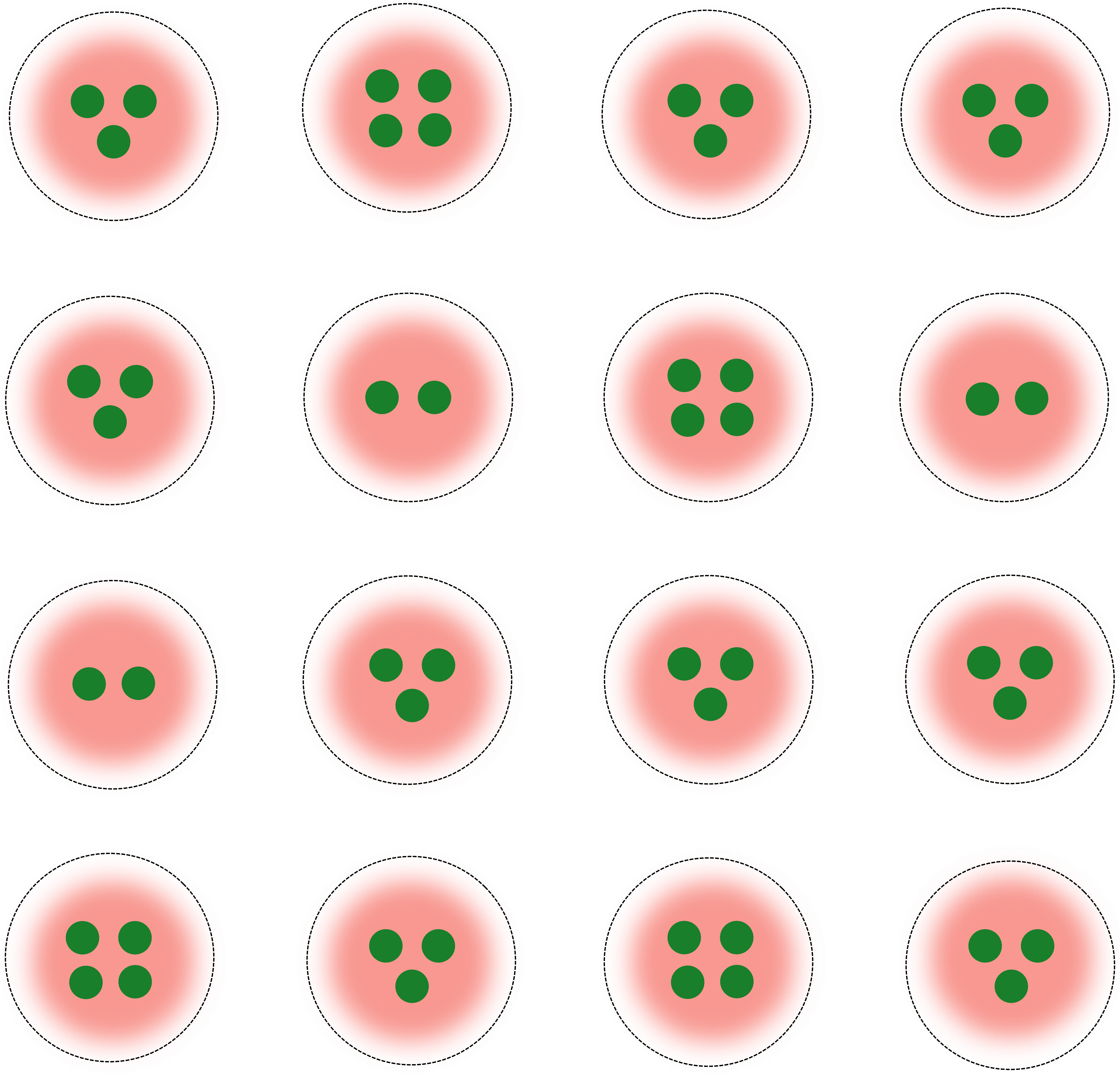
1. Reconfigure

2. Inter-cluster gate: $$H_I = \sum_{ij} J_{ij}^{\alpha\beta} \hat{s}^{\alpha}_{i,a_{ij}} \hat{s}^{\beta}_{j,b_{ij}}$$

- Spin-1/2 gates and local rotations
- Mediates generic, long-range connectivity
- Violates large-spin encoding

3. Reconfigure

Engineering Interactions



Target Hamiltonian:

$$H_{target} = \sum_{i,j} J_{ij}^{\alpha\beta}\, \hat{S}_i^{\alpha}\, \hat{S}_j^{\beta}$$

1. Reconfigure

2. Inter-cluster gate: $$H_I = \sum_{ij} J_{ij}^{\alpha\beta}\, \hat{s}_{i,a_{ij}}^{\alpha}\, \hat{s}_{j,b_{ij}}^{\beta}$$

- Spin-1/2 gates and local rotations
- Mediates generic, long-range connectivity
- Violates large-spin encoding

3. Reconfigure

4. Intra-cluster gate: $$H_C = -\sum_i P_{\mathrm{sym}}[(\vec{S}_i)^2]$$

- Encoding space is gapped ground state
- Applies phase to encoding violating terms

Floquet Sequence to Implement Model Hamiltonian

Effective evolution operator:

$$U_F = \prod_k e^{-i\theta_k H_C} e^{-i\tau H_I}$$

Floquet Sequence to Implement Model Hamiltonian

Effective evolution operator:

$$U_F = \prod_k e^{-i\theta_k H_C} e^{-i\tau H_I}$$

can be large-angle rotations

Floquet Sequence to Implement Model Hamiltonian

Effective evolution operator:

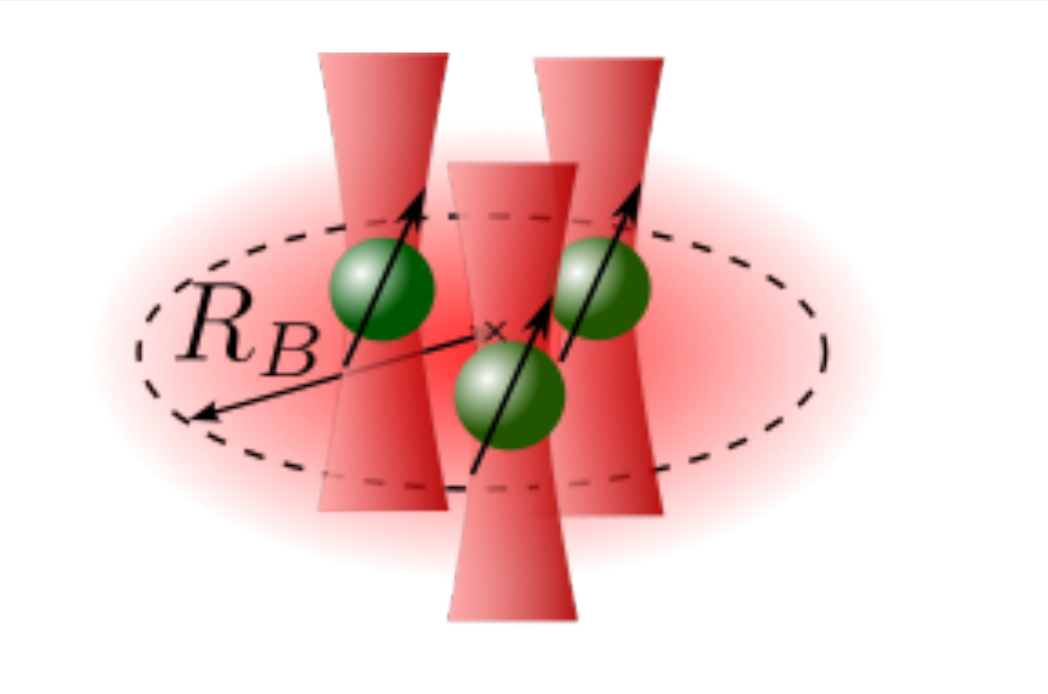
$$U_F = \prod_k e^{-i\theta_k H_C} e^{-i\tau H_I}$$

$\Rightarrow$ Realizes target Hamiltonian on average

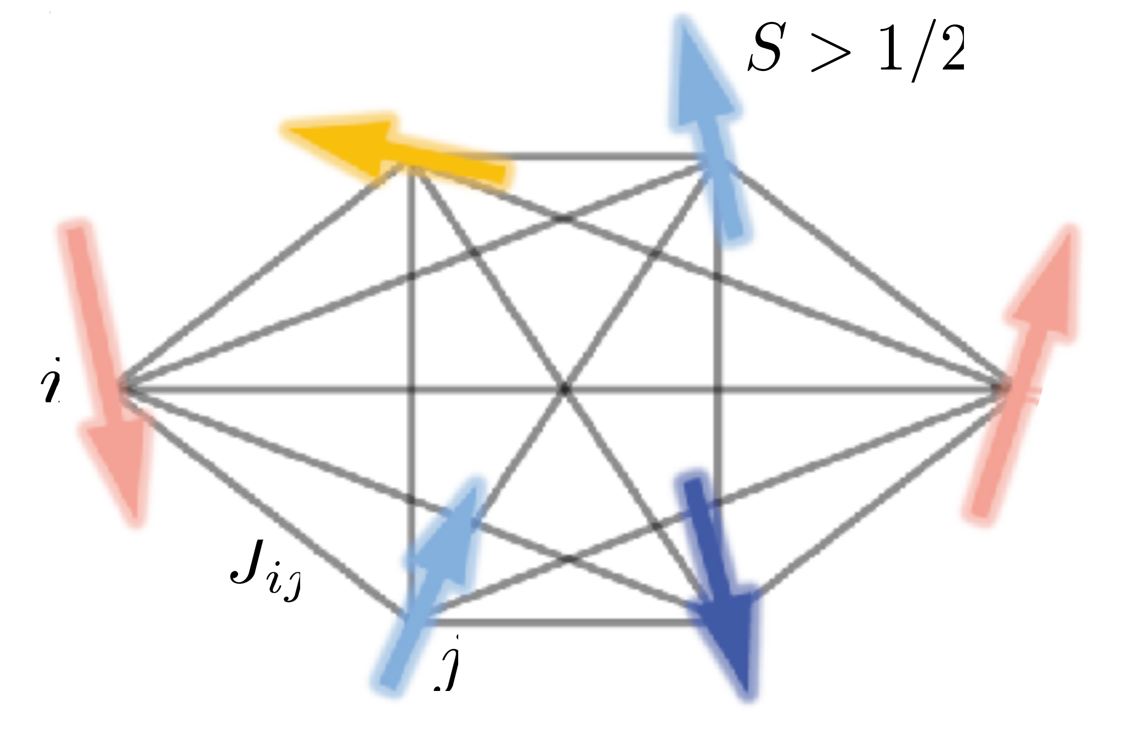
Higher-order errors can be cancelled out, or controlled via Floquet engineering.

Necessary Ingredients

Control high spin (S>1/2)

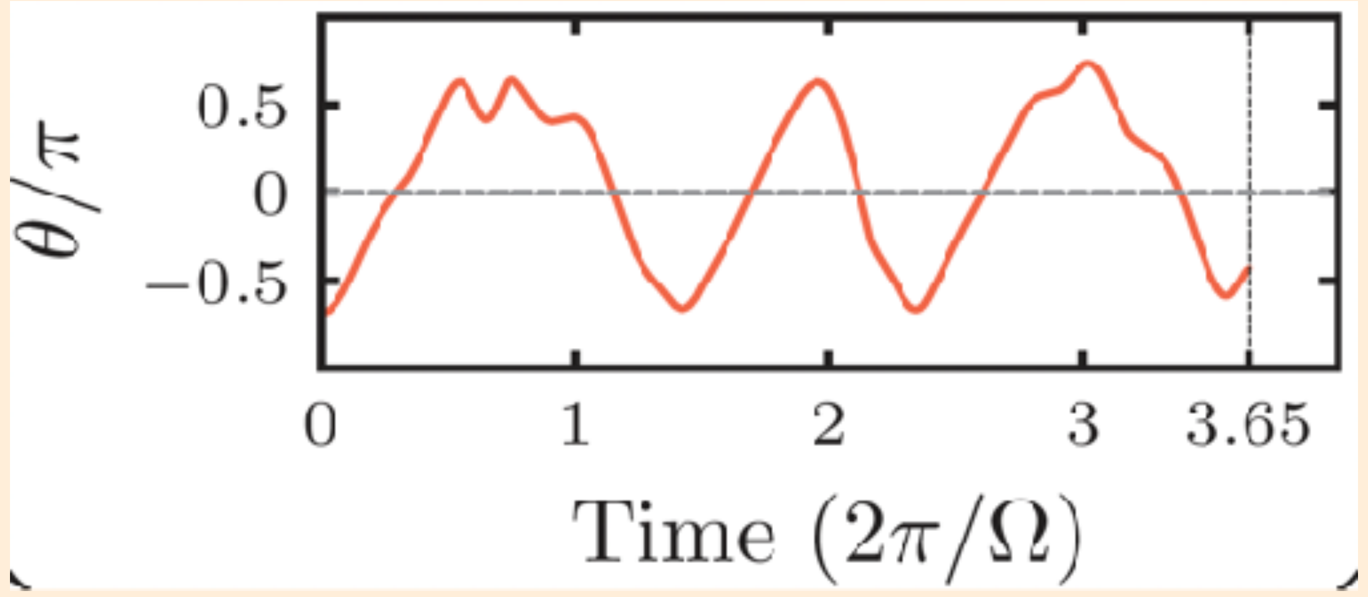


Non-local connectivity

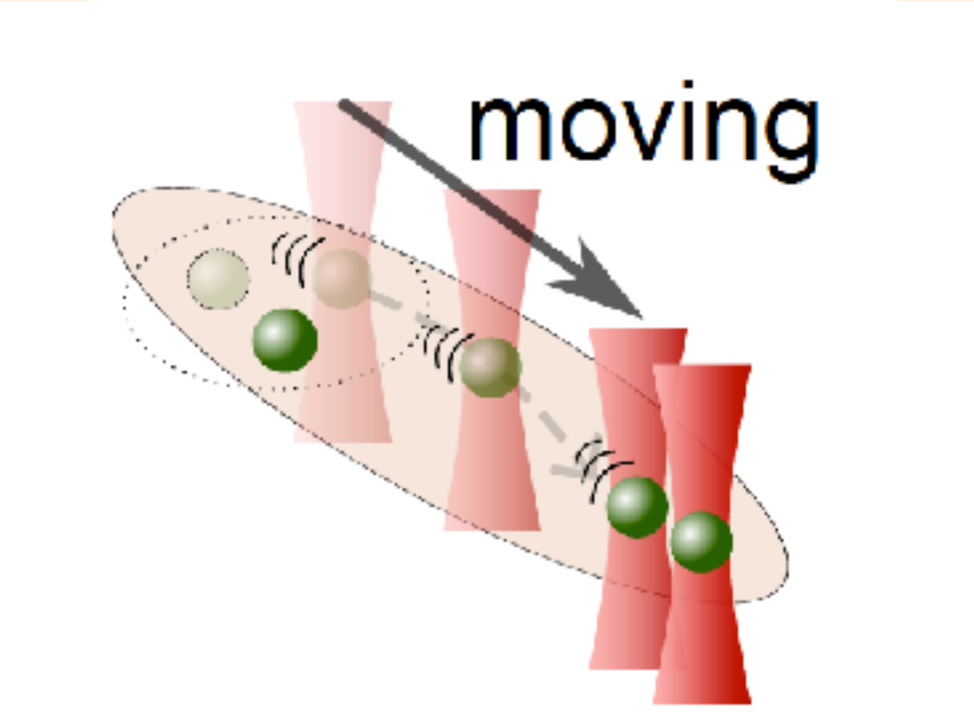


Solution read-out

Native multi-qubit gates



Atom moving



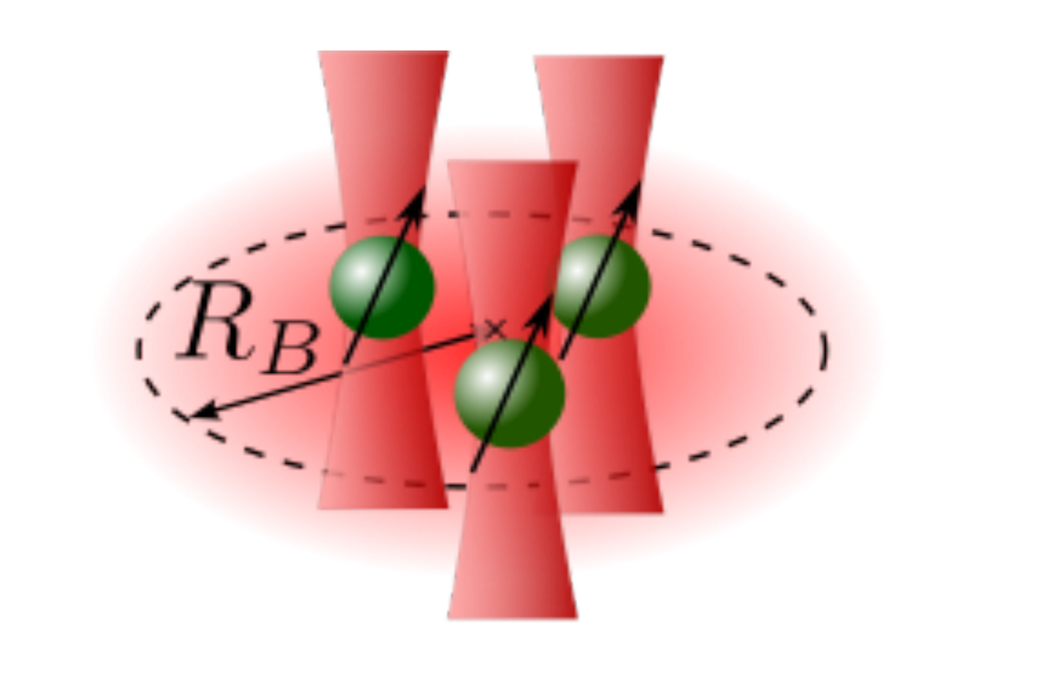
D. Bluvstein et al., Nature **604**, 451–456 (2022)

Co-processing

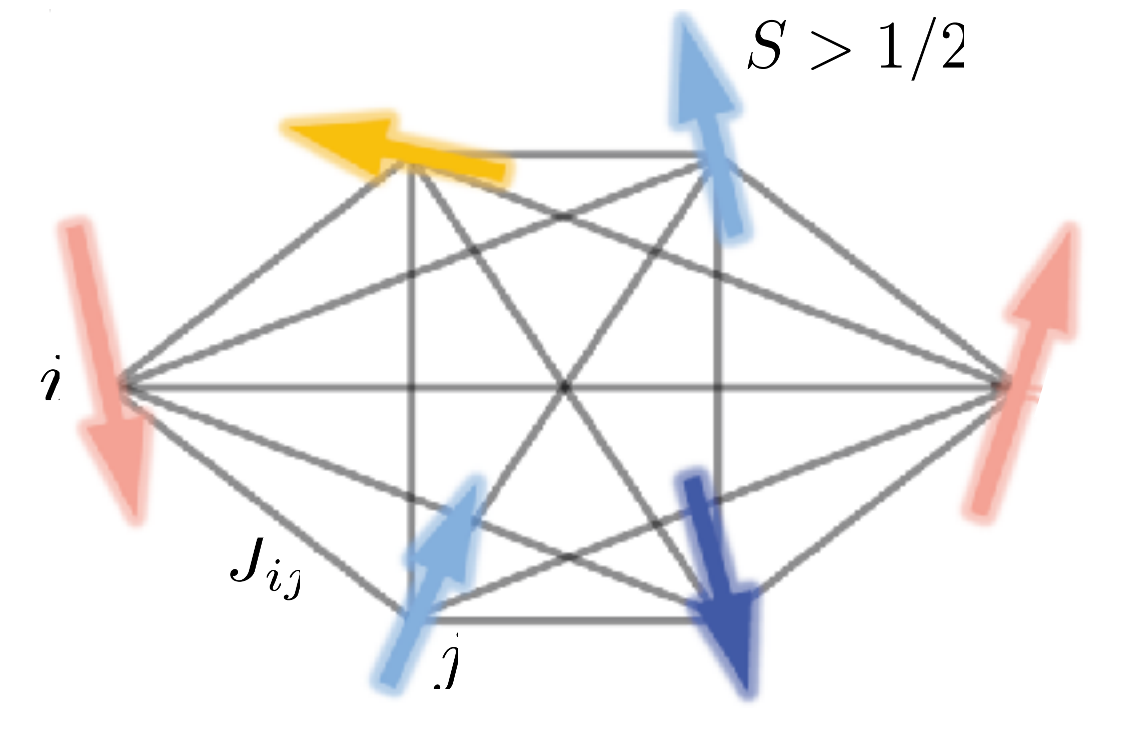
H.-Y. Huang et al., Nat. Phys. 16, 1050–1057 (2020)

Necessary Ingredients

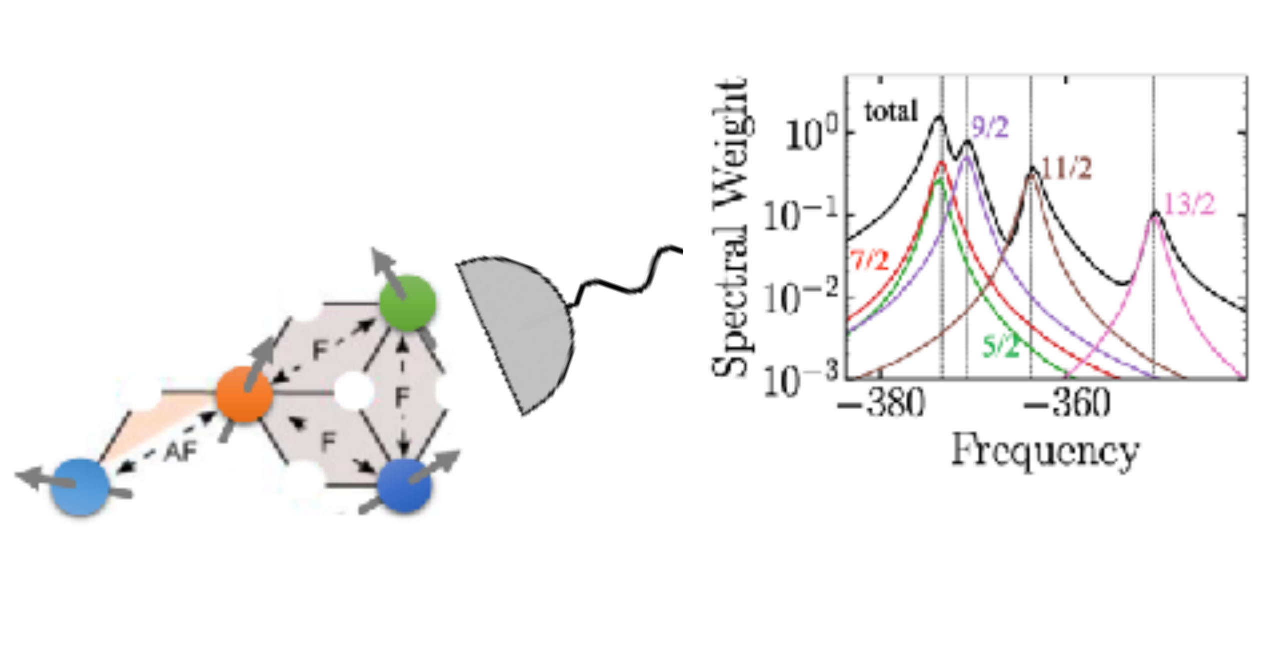
Control high spin (S>1/2)



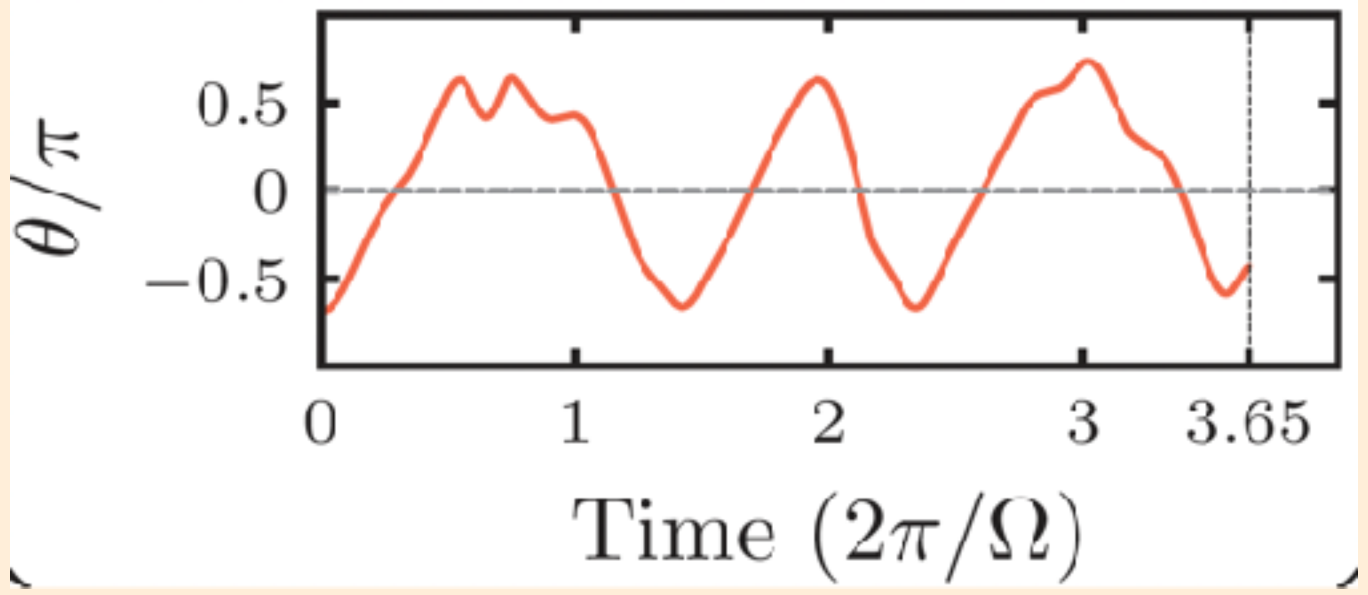
Non-local connectivity



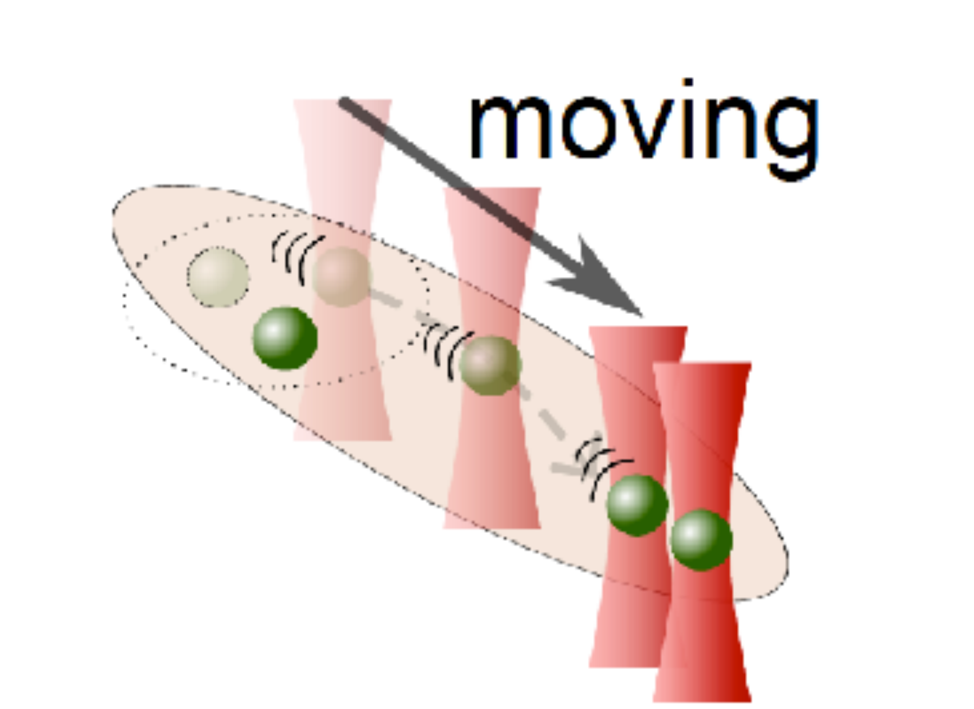
Solution read-out



Native multi-qubit gates

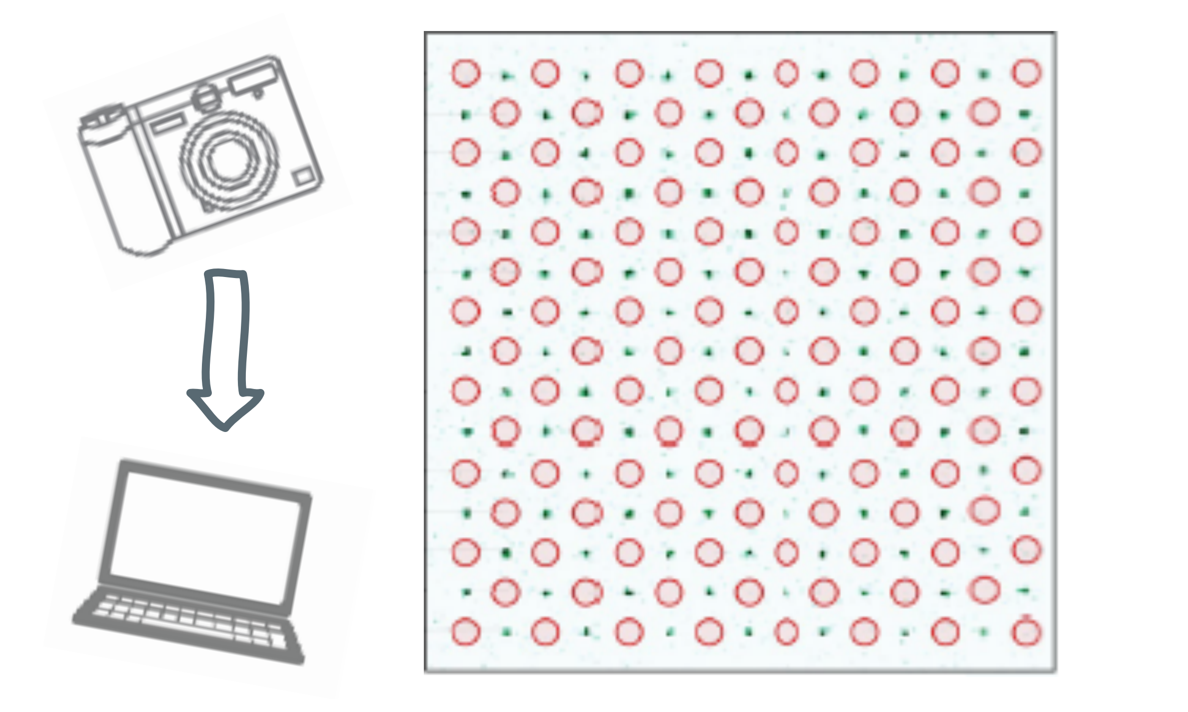


Atom moving



D. Bluvstein et al., Nature **604**, 451–456 (2022)

Co-processing



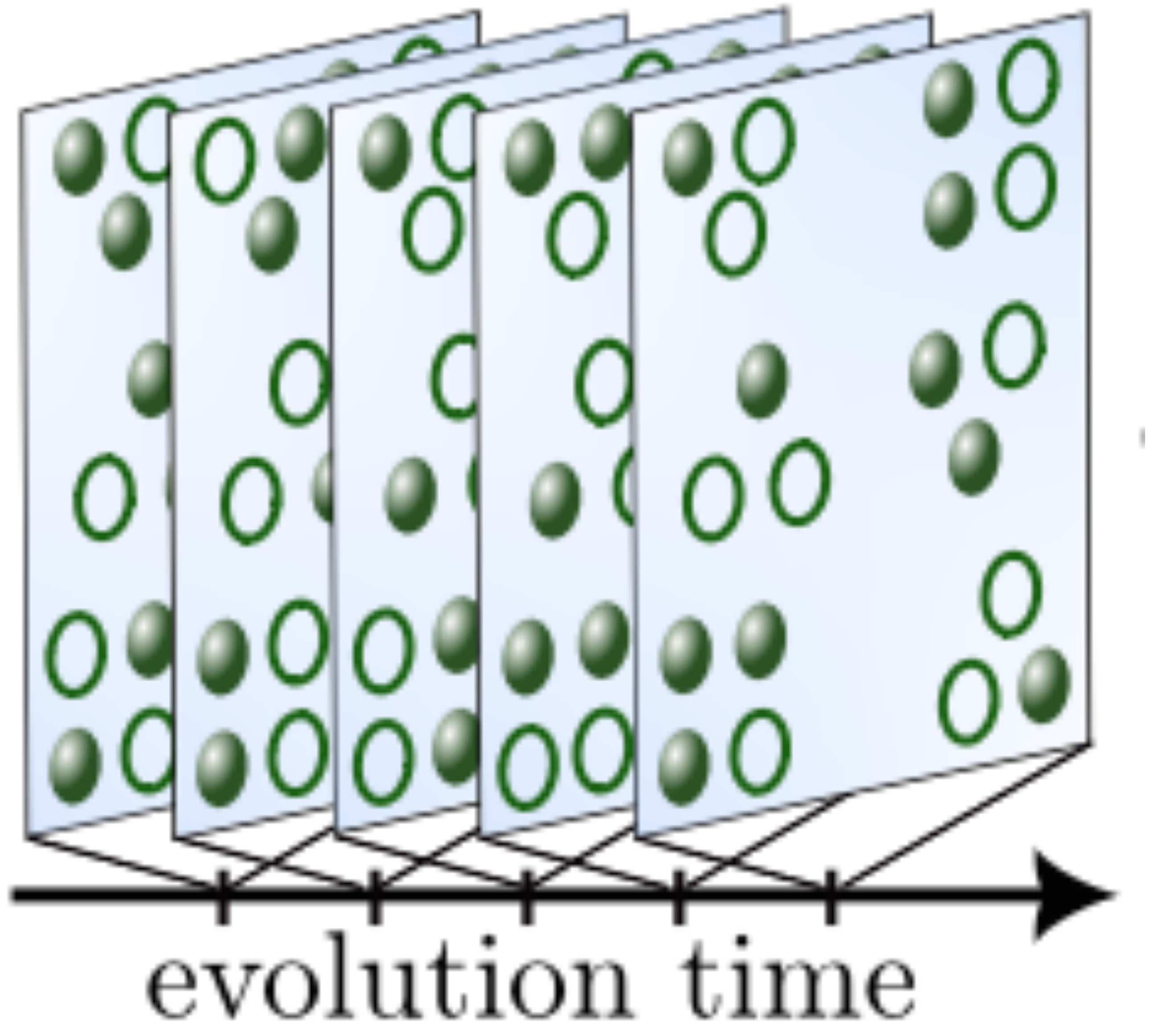
H.-Y. Huang et al., Nat. Phys. 16, 1050–1057 (2020)

Efficient Read-Out Based on Snapshots

Leverage

– ability for efficient time evolution of Rydberg simulator

– ability to perform snapshot measurements



Such measurements are very information dense!

Efficient Read-Out Based on Snapshots

Leverage

– ability for efficient time evolution of Rydberg simulator

– ability

Key quantity: Operator-resolved density of states

$$D_A(\omega) = \sum_n \langle n | A | n \rangle \, \delta(\omega - \epsilon_n)$$

properties of eigenstates

peaks at state energies

Such measurements are ve ion den

Efficient Read-Out Based on Snapshots

Leverage

– ability for efficient time evolution of Rydberg simulator

– ability

Key quantity: Operator-resolved density of states

$$D_A(\omega) = \sum_n \langle n | A | n \rangle \, \delta(\omega - \epsilon_n)$$

How to obtain?

peaks at state energies

Such measurements are ve den

Efficient readout: Quantum-Classical Co-Processing

Hardware-efficient toolbox to compute spectral functions

$$D_A(\omega) = \sum \langle n | A | n \rangle \, \delta(\omega - \epsilon_n) = $$

$$\boxed{\int \mathrm{Corr}_{\mathrm{classical}} \; \mathrm{Corr}_{\mathrm{quantum}}}$$

evolution times
state samples
observables

Efficient readout: Quantum-Classical Co-Processing

Hardware-efficient toolbox to compute spectral functions

$$D_A(\omega) = \sum \langle n | A | n \rangle \, \delta(\omega - \epsilon_n) =$$

$$\int \quad \mathrm{Corr}_{\mathrm{classical}} \; \mathrm{Corr}_{\mathrm{quantum}}$$

evolution times

sta

o

time evolution of known reference state

Efficient readout: Quantum-Classical Co-Processing

Hardware-efficient toolbox to compute spectral functions

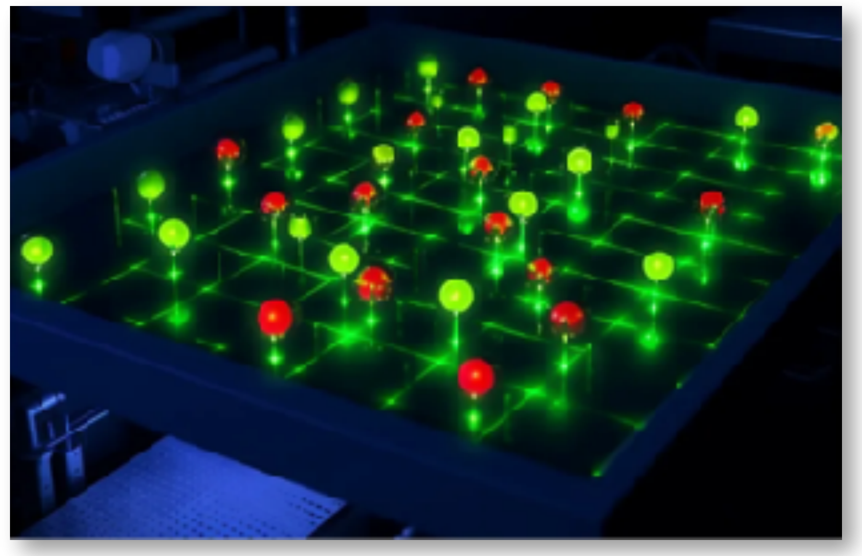
$$D_A(\omega) = \sum \langle n | A | n \rangle \, \delta(\omega - \epsilon_n) = \int_{\text{evolution times}} \mathrm{Corr}_{\text{classical}} \; \mathrm{Corr}_{\text{quantum}}$$

sta

o

time evolution of known reference state

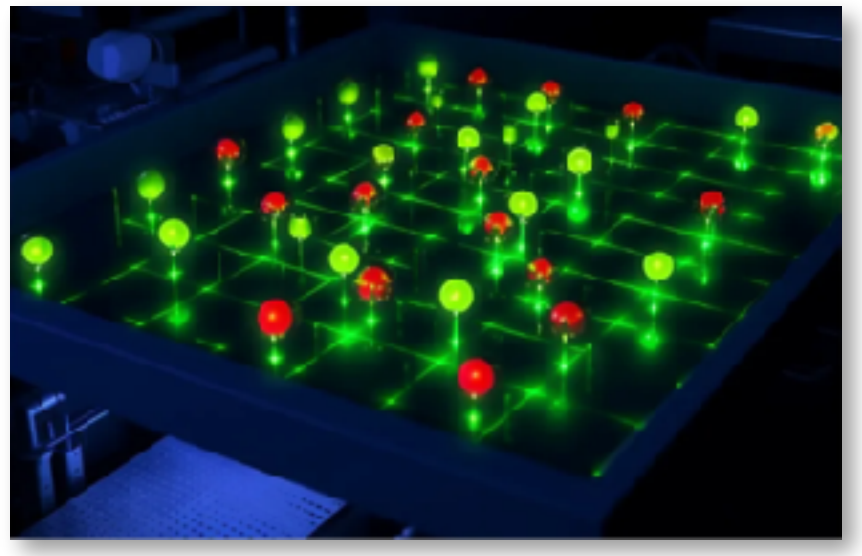
time evolution of unknown superposition state



Efficient readout: Quantum-Classical Co-Processing

Hardware-efficient toolbox to compute spectral functions

$$D_A(\omega) = \sum \langle n|A|n\rangle \, \delta(\omega - \epsilon_n) =$$



$$\int \quad \mathrm{Corr}_{\mathrm{classical}} \; \mathrm{Corr}_{\mathrm{quantum}}$$

evolution times

sta

ol

time evolution of known reference state

time evolution of unknown superposition state

$D^A(\omega) = \int d$ … $O_s U(t)|$ …

$\simeq C^*_{O_s,R}(0)e^{-iE_S t}$ $\quad C_{O_s,R}(t) + O(1/\sqrt{M})$

Uniformly sample perturbed states

$$\mathbb{E}_{R\sim\mathcal{R}}\,[R|S\rangle\langle S|R] = 2^{-n} I$$

Sum over observables

$$\sum [O_s|S\rangle\langle S|O_s] = I$$

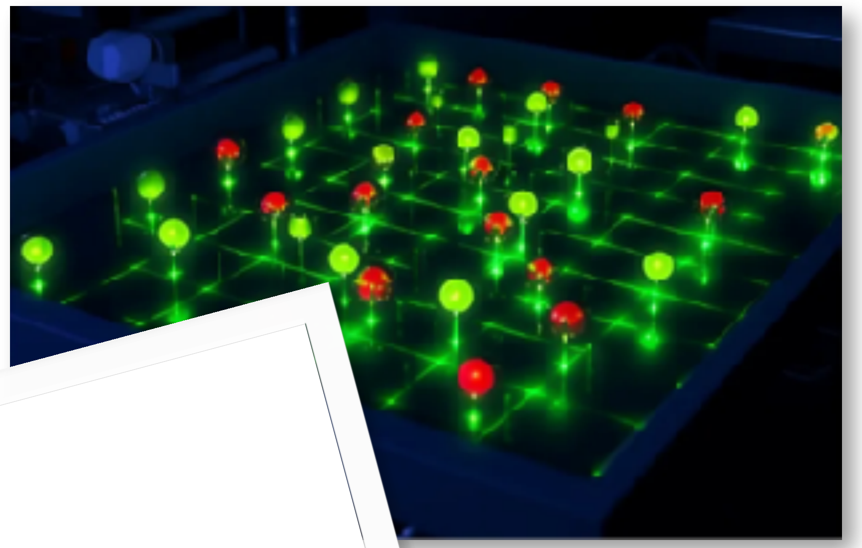
Estimate via **quantum** time evolution on simulator and *M* snapshots.

Evaluate **classically** using an efficient representation e.g. MPS $|S\rangle$, $R|S\rangle$ and MPO A

Efficient readout: Quantum-Classical Co-Processing

Hardware-efficient toolbox to compute spectral functions

$$D_A(\omega) = \sum \langle n | A | n \rangle \, \delta(\omega - \epsilon_n) =$$



$$\int \mathrm{Corr}_{\mathrm{classical}} \; \mathrm{Corr}_{\mathrm{quantum}}$$

evolution times

sta…

o…

time evo…

kno…

superposition state

Makes procedure sample efficient:

D_A can be found for **different A** with **same sample set**

D^A …

$\ldots(0)e^{-iE_s t}$ $\quad C_{O_s,R}(t) + O(1/\sqrt{M})$

Uniformly sample perturbe…

$\mathbb{E}_{R\sim\mathcal{R}}\left[R|S\rangle\langle S|R\right] = \ldots$

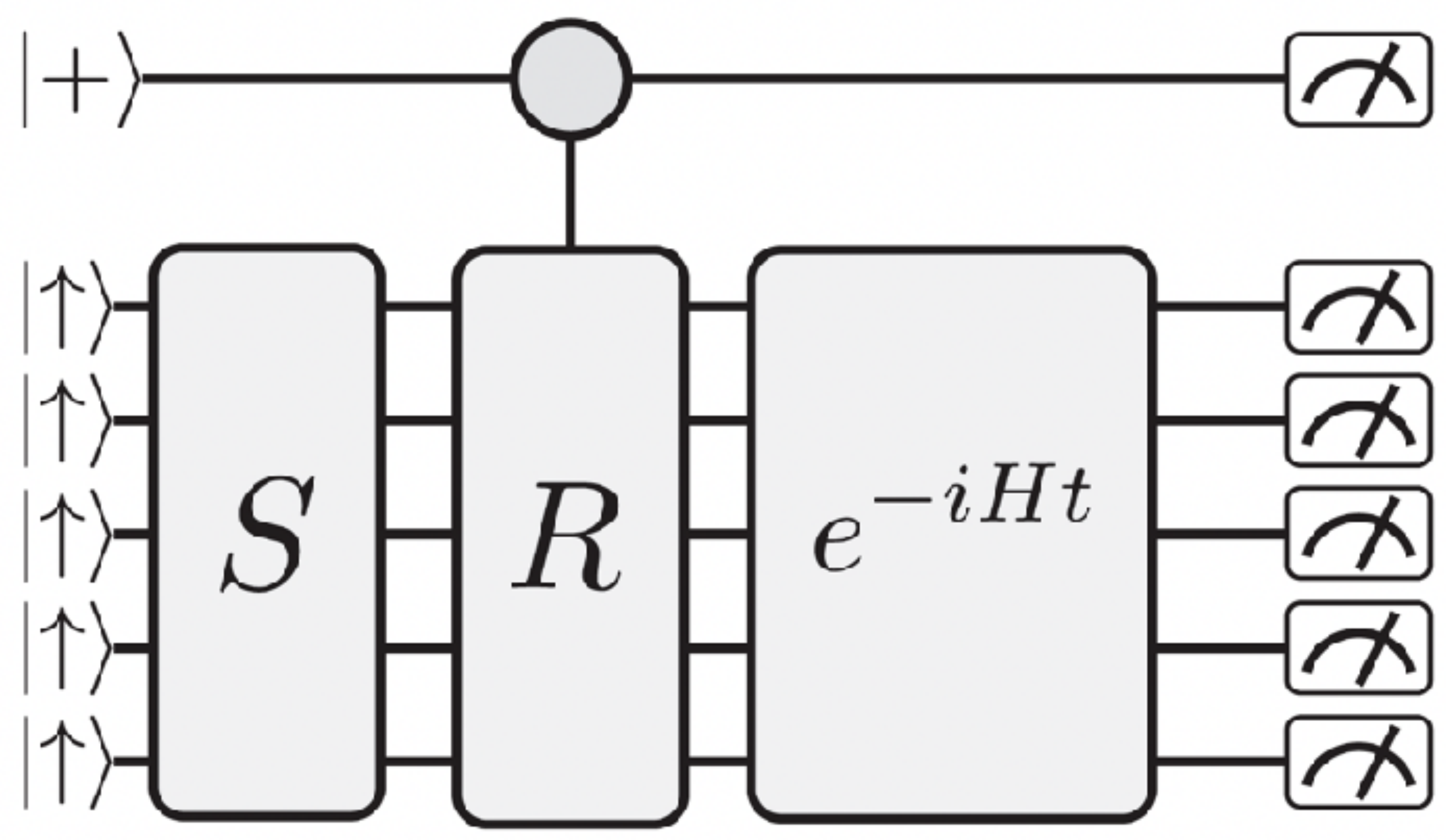
Sum over observables

$\sum [O_s|S\rangle\langle S|O_s] = I$

Estimate via **quantum** time evolution on simulator and *M* snapshots.

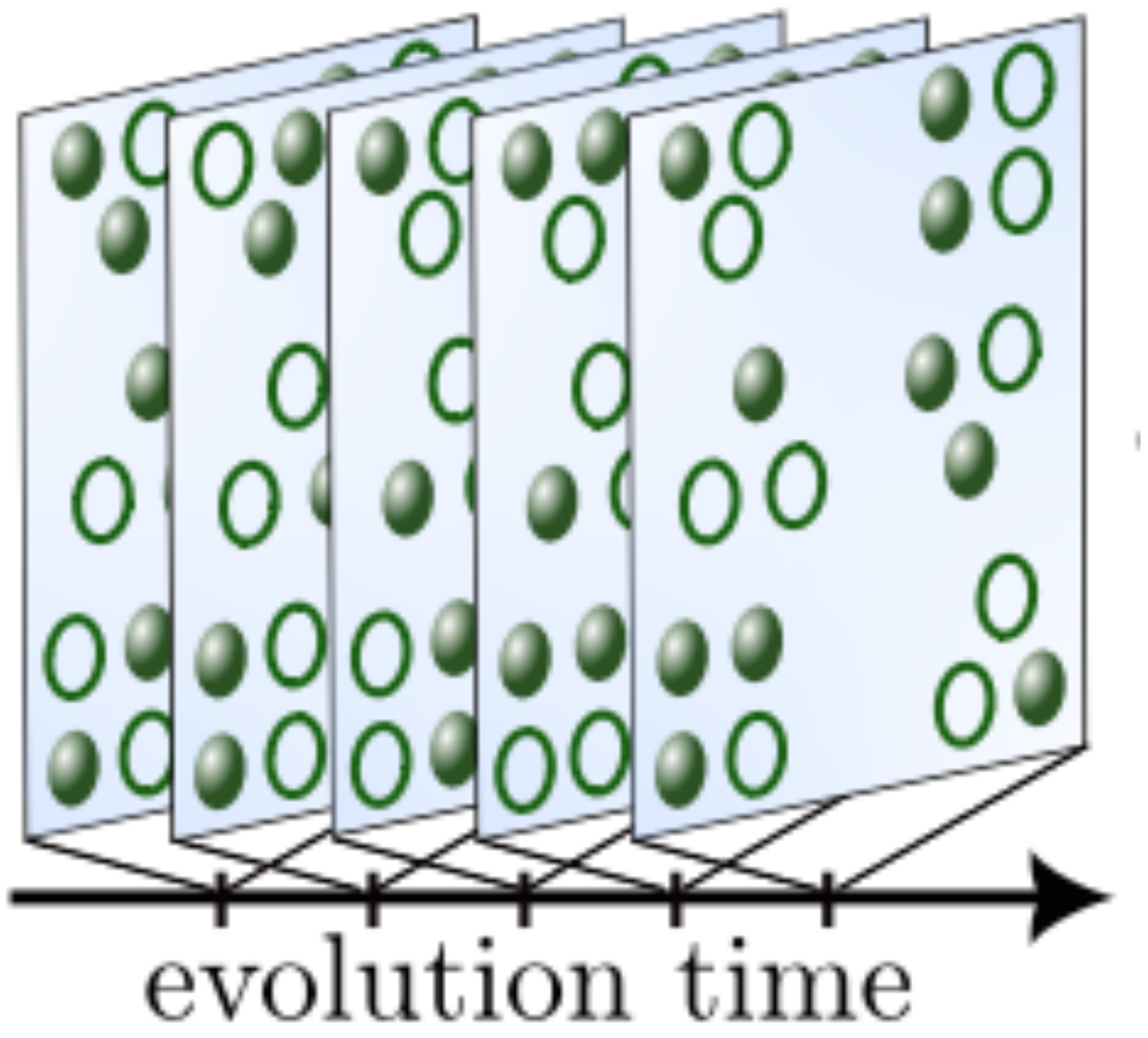
Evaluate **classically** using an efficient representation e.g. MPS $|S\rangle$, $R|S\rangle$ and MPO A

Efficient readout: Quantum Circuit + Spectrum



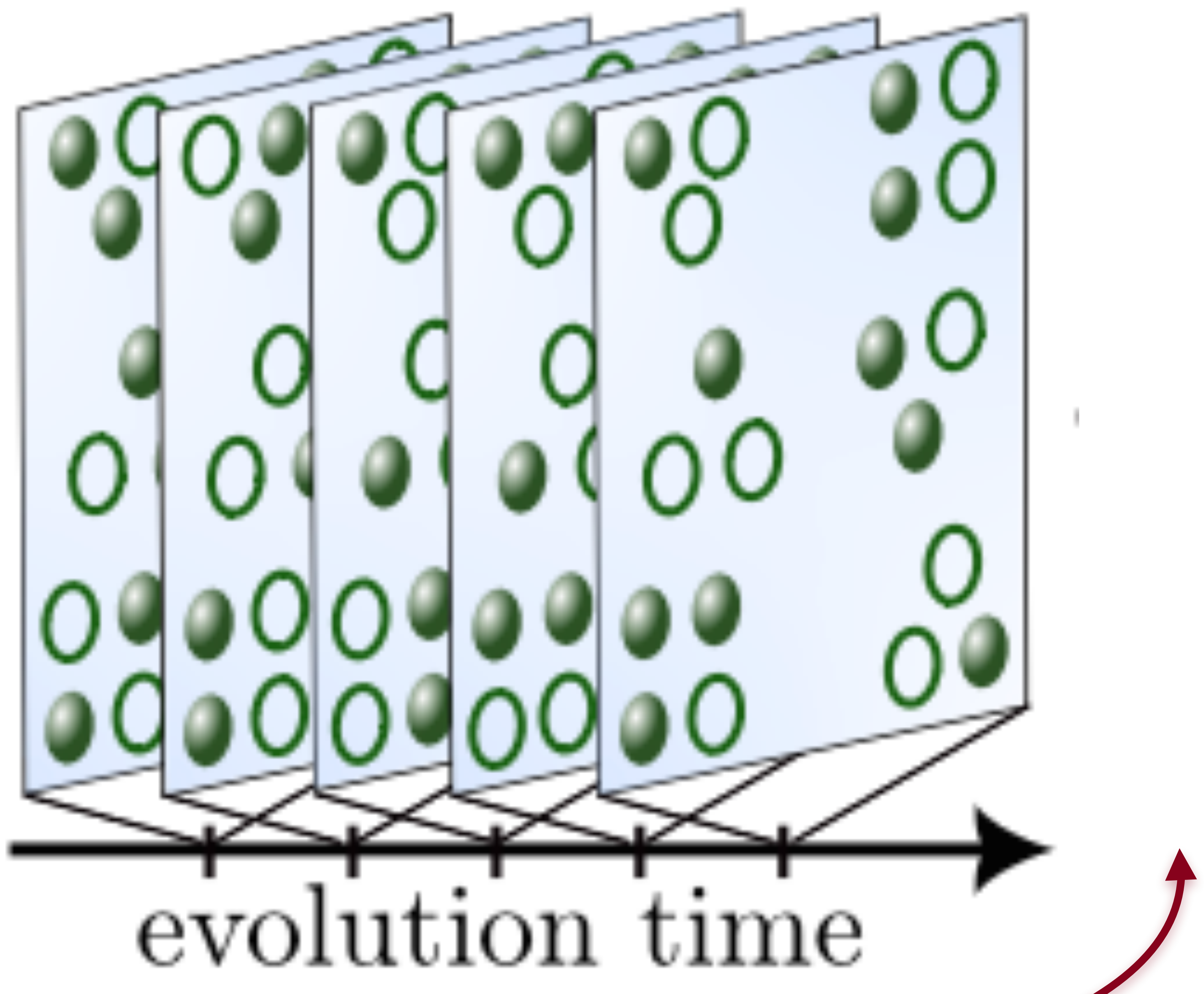
Parallel measurement of 2^n observables
(any operator diagonal in measurement basis)

Efficient readout: Quantum Circuit + Spectrum



Parallel measurement of 2^n observables
(any operator diagonal in measurement basis)

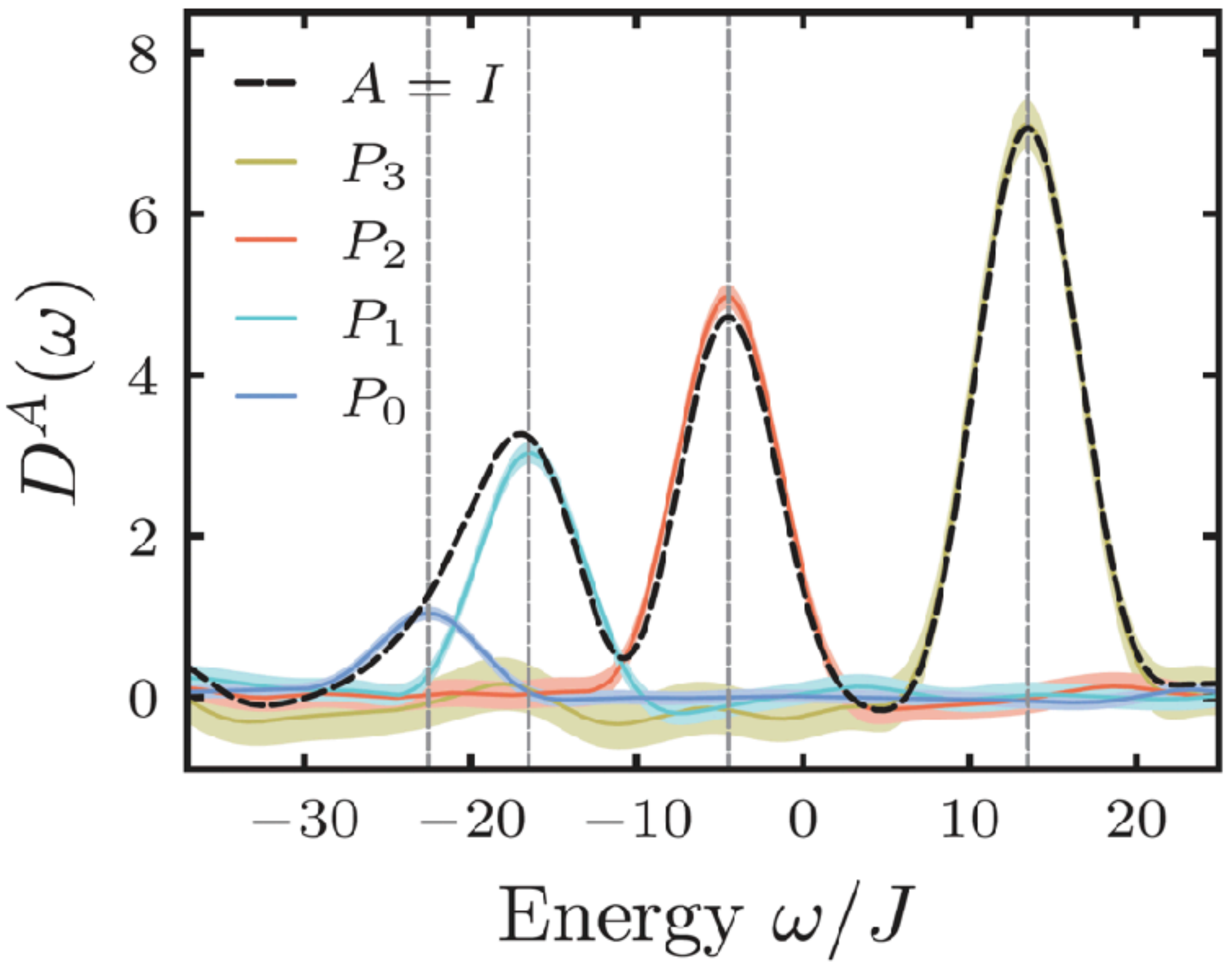
Efficient readout: Quantum Circuit + Spectrum



Parallel measurement of 2^n observables
(any operator diagonal in measurement basis)

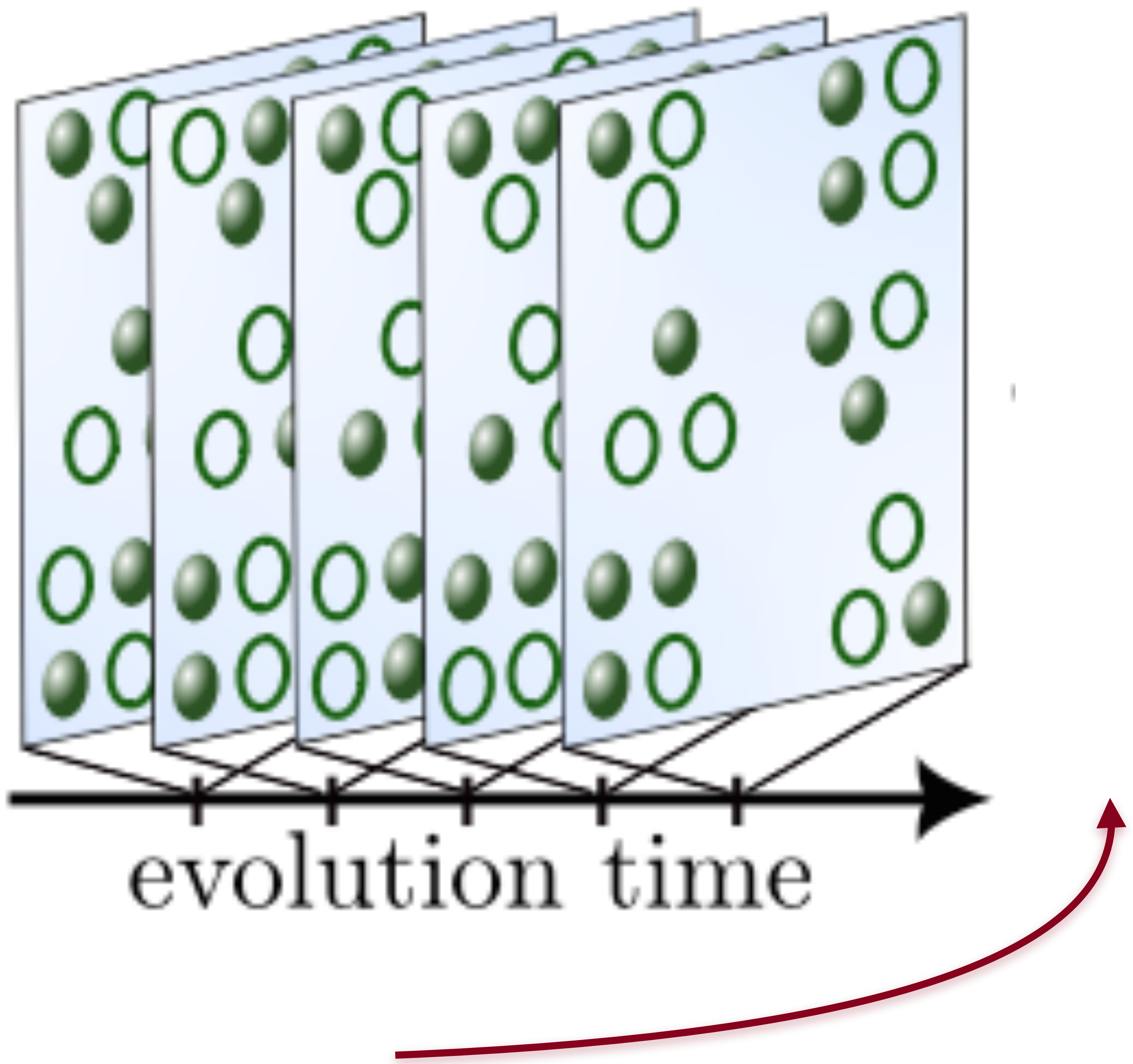
Example: two interacting spin-3/2s

$$H = J\,\mathbf{S}_1 \cdot \mathbf{S}_2$$



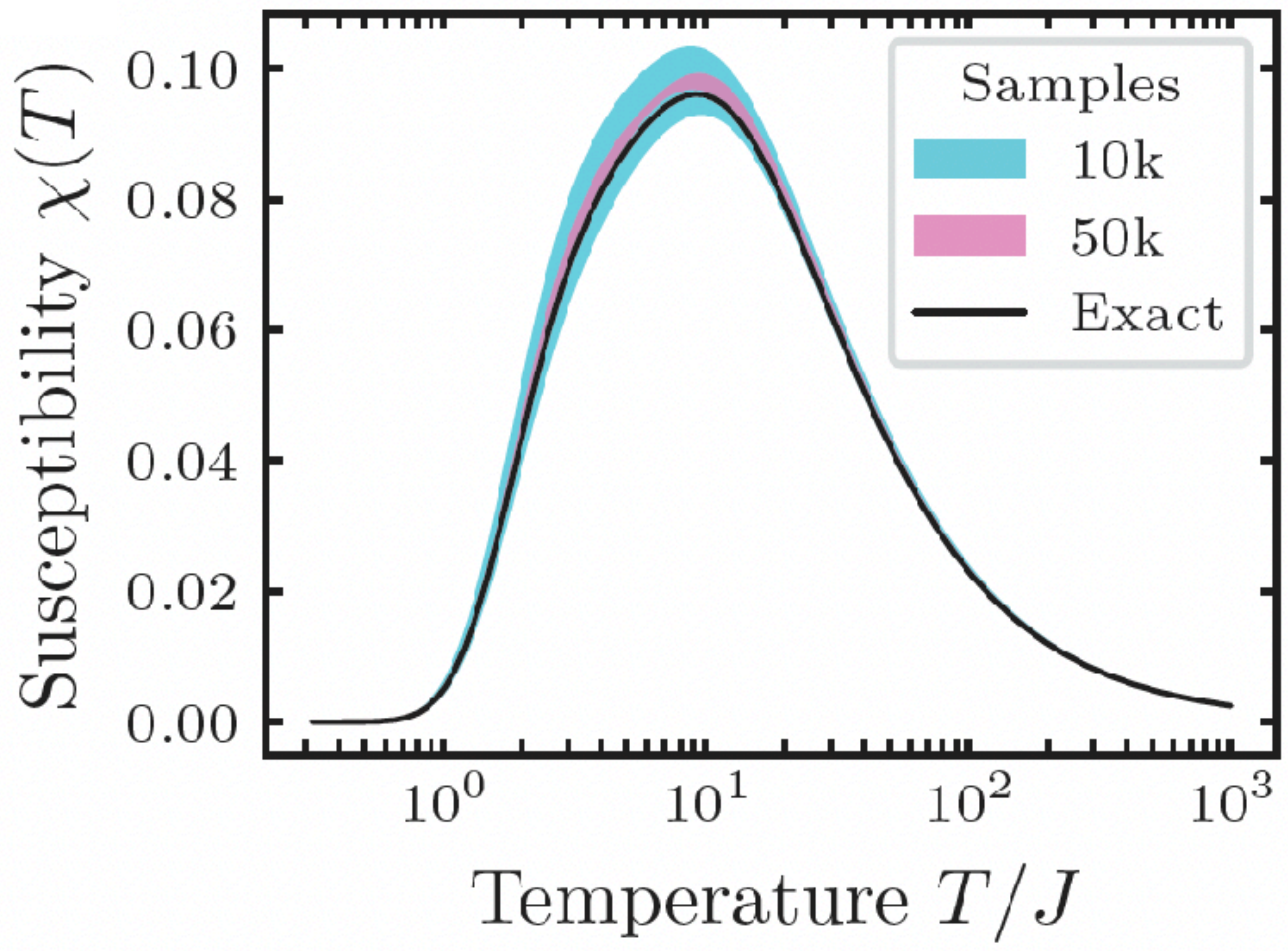
20,000 samples

Efficient readout: Quantum Circuit + Spectrum

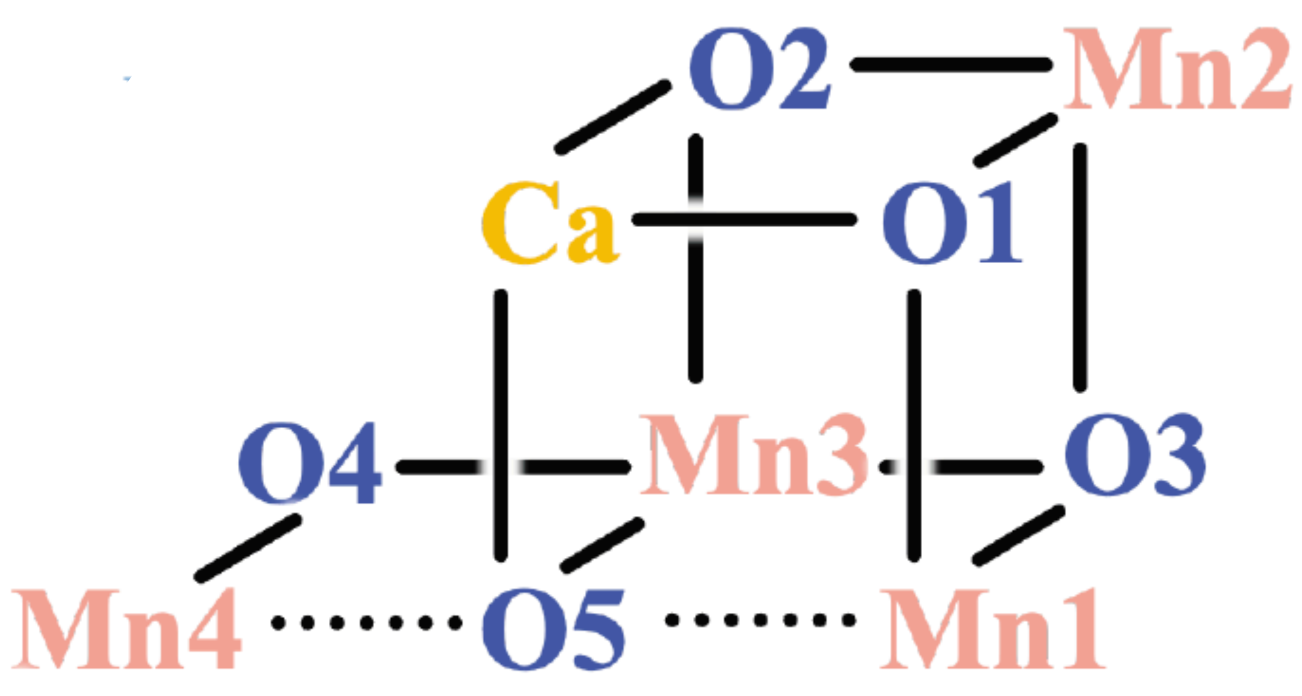


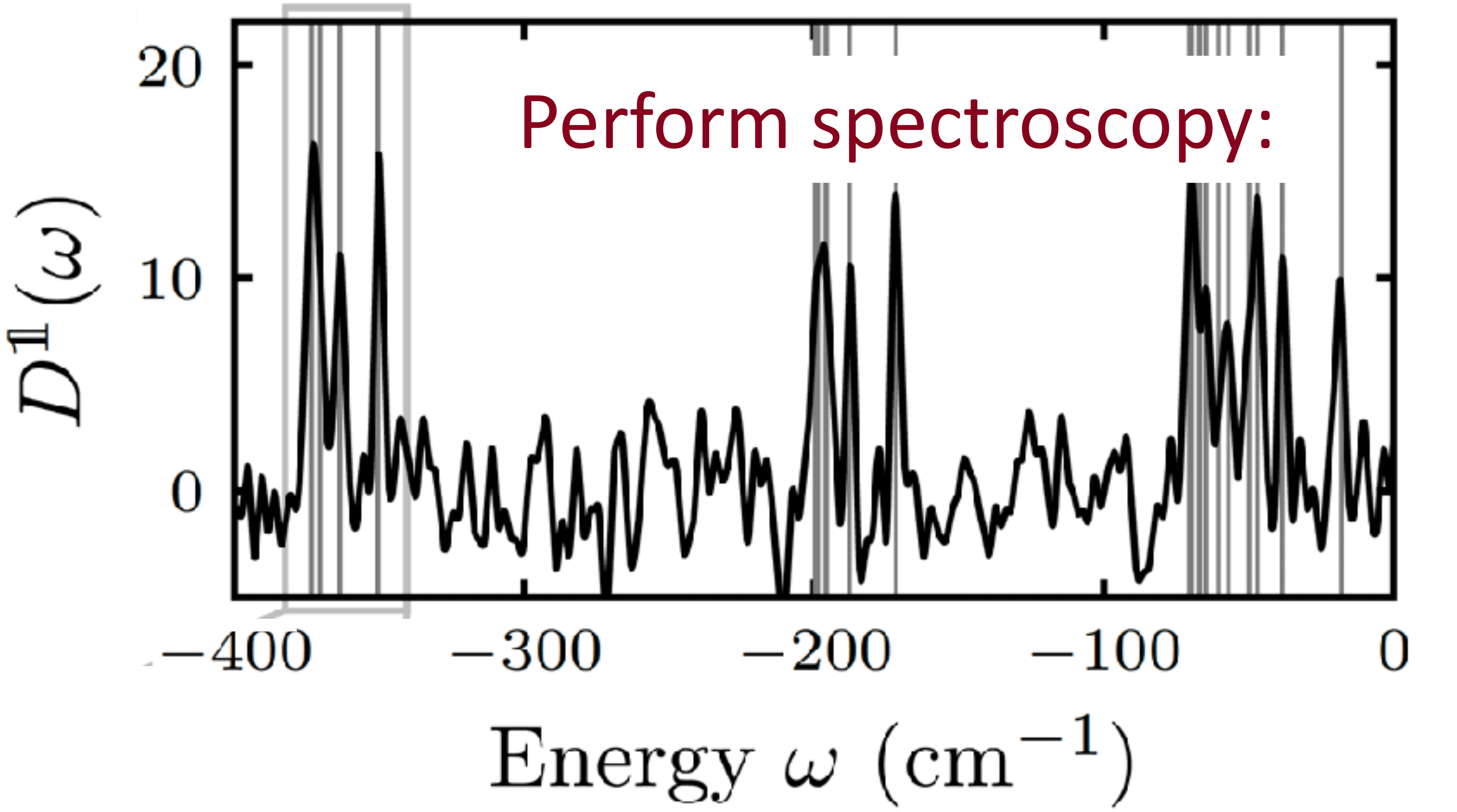
Parallel measurement of 2^n observables
(any operator diagonal in measurement basis)

Access to full spectral information
⇒ finite temperature properties!

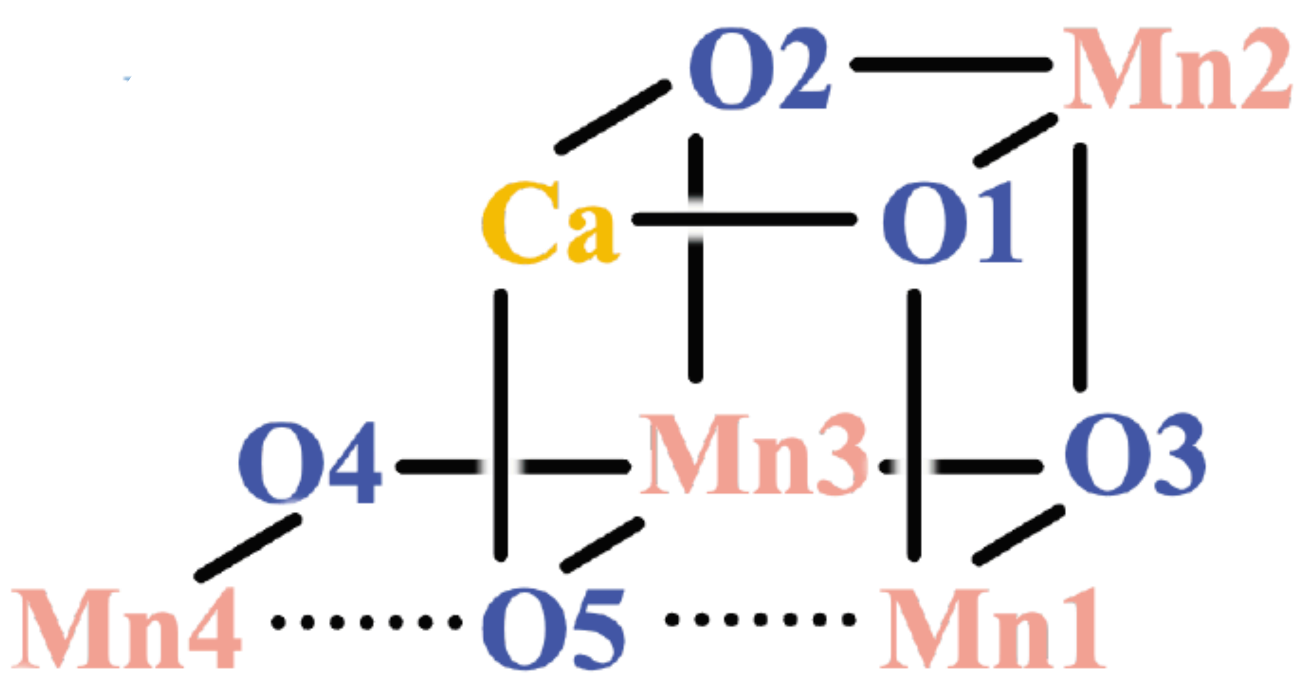


Benchmark on Larger Molecule (OEC Complex)

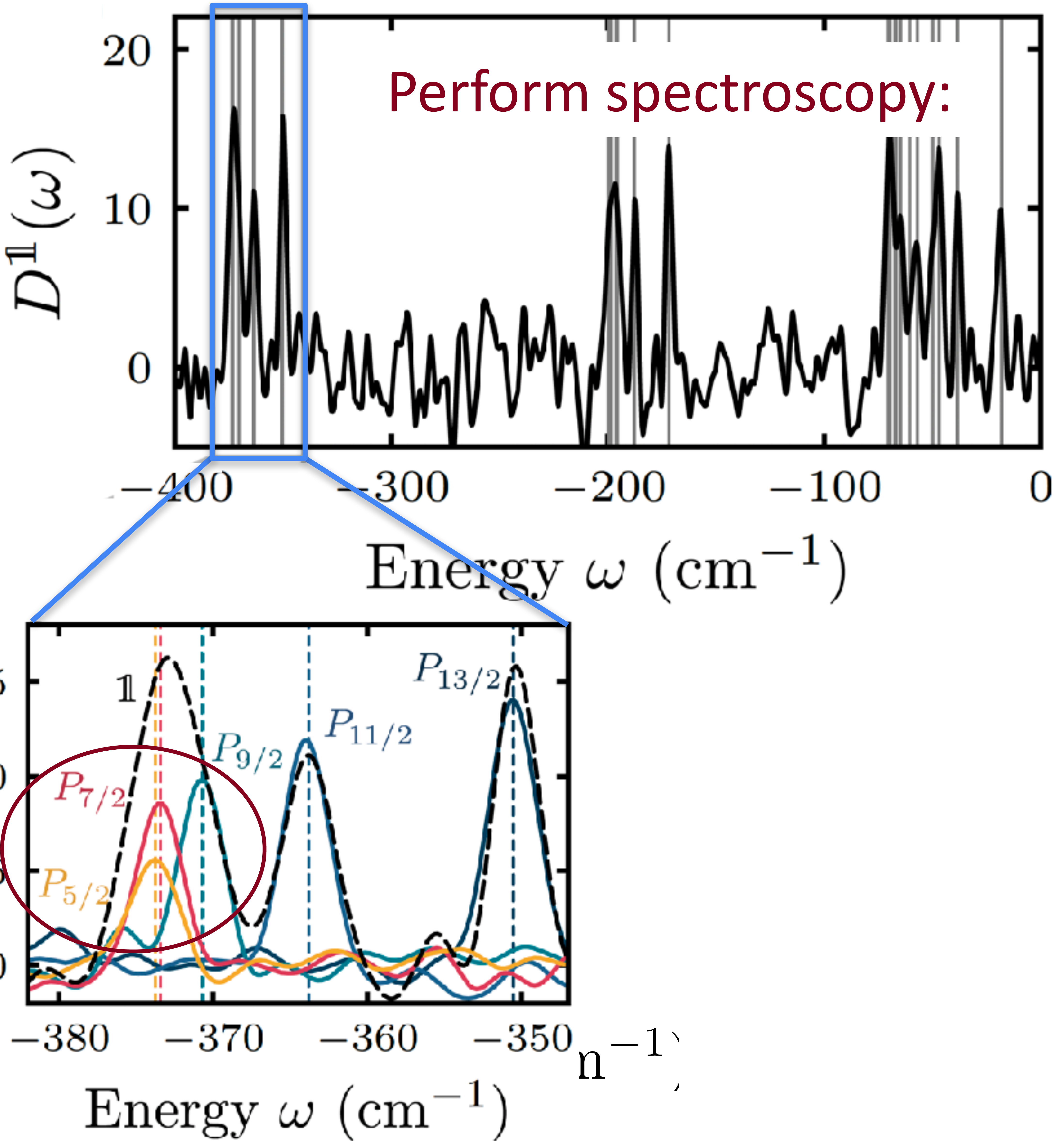




Benchmark on Larger Molecule (OEC Complex)

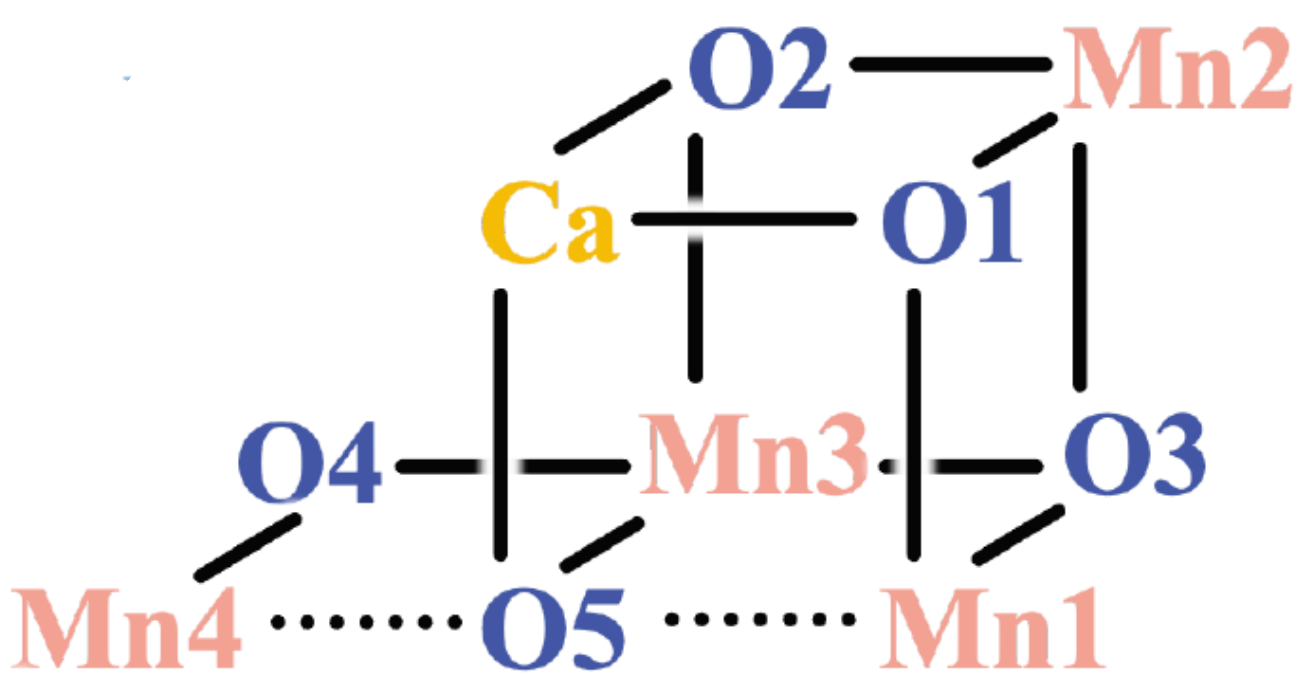


Perform spectroscopy:

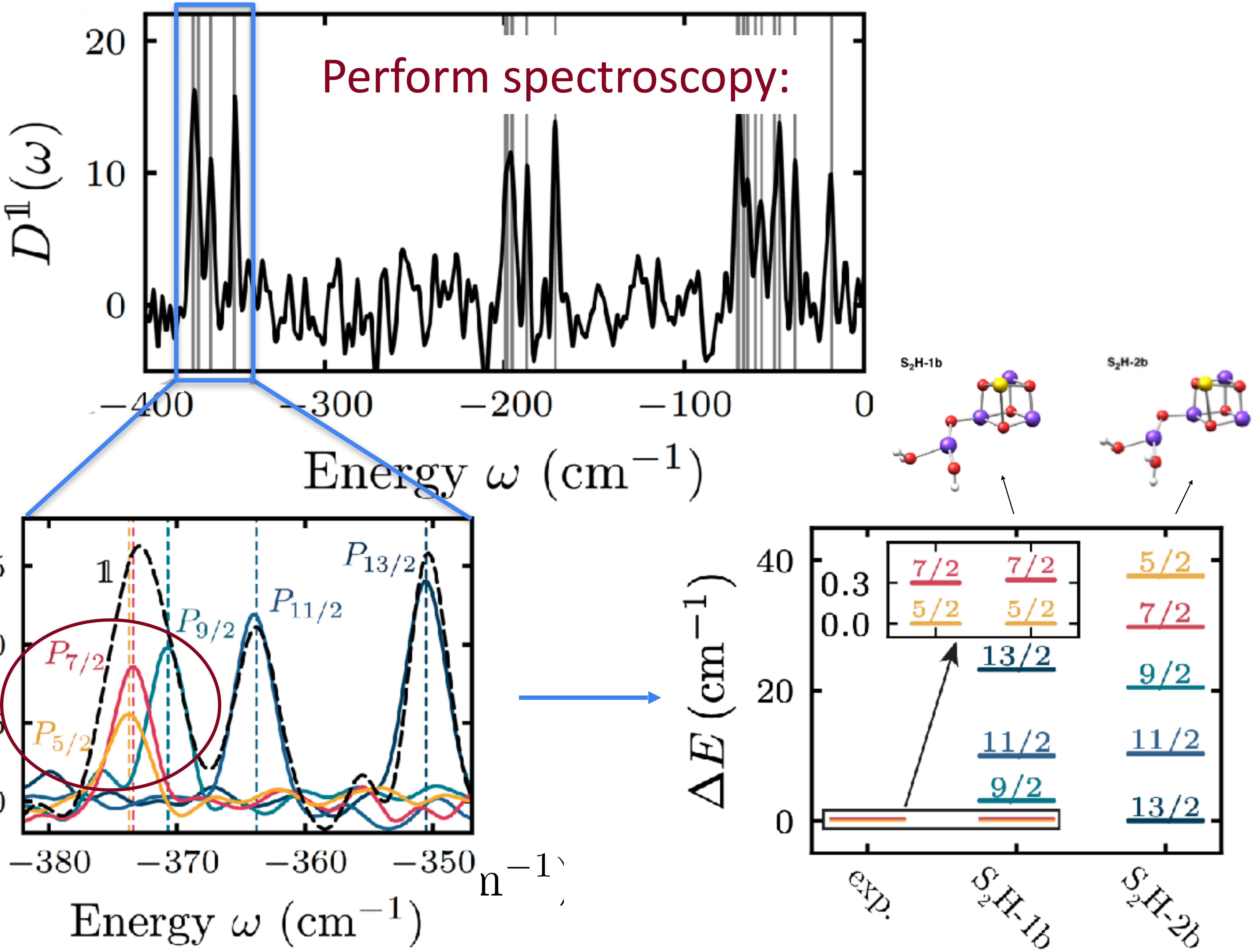


spin-projection allows detection of tiny energy differences (0.3cm^{-1})

Benchmark on Larger Molecule (OEC Complex)

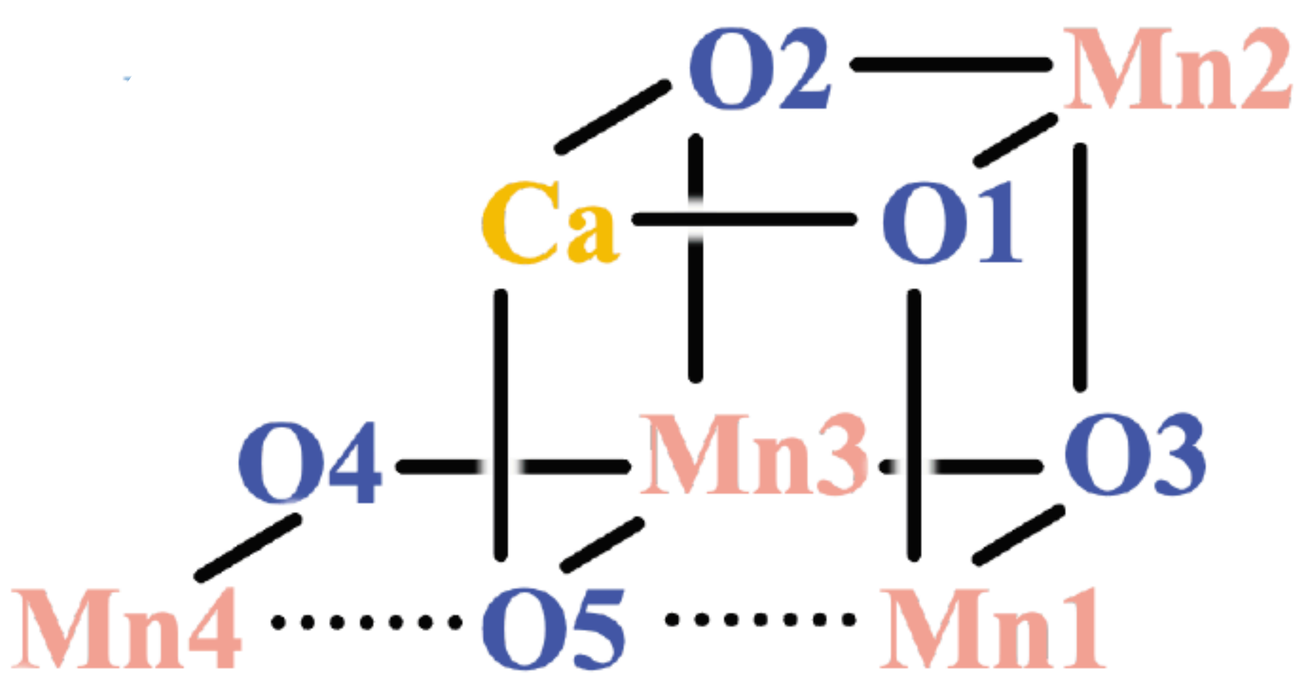


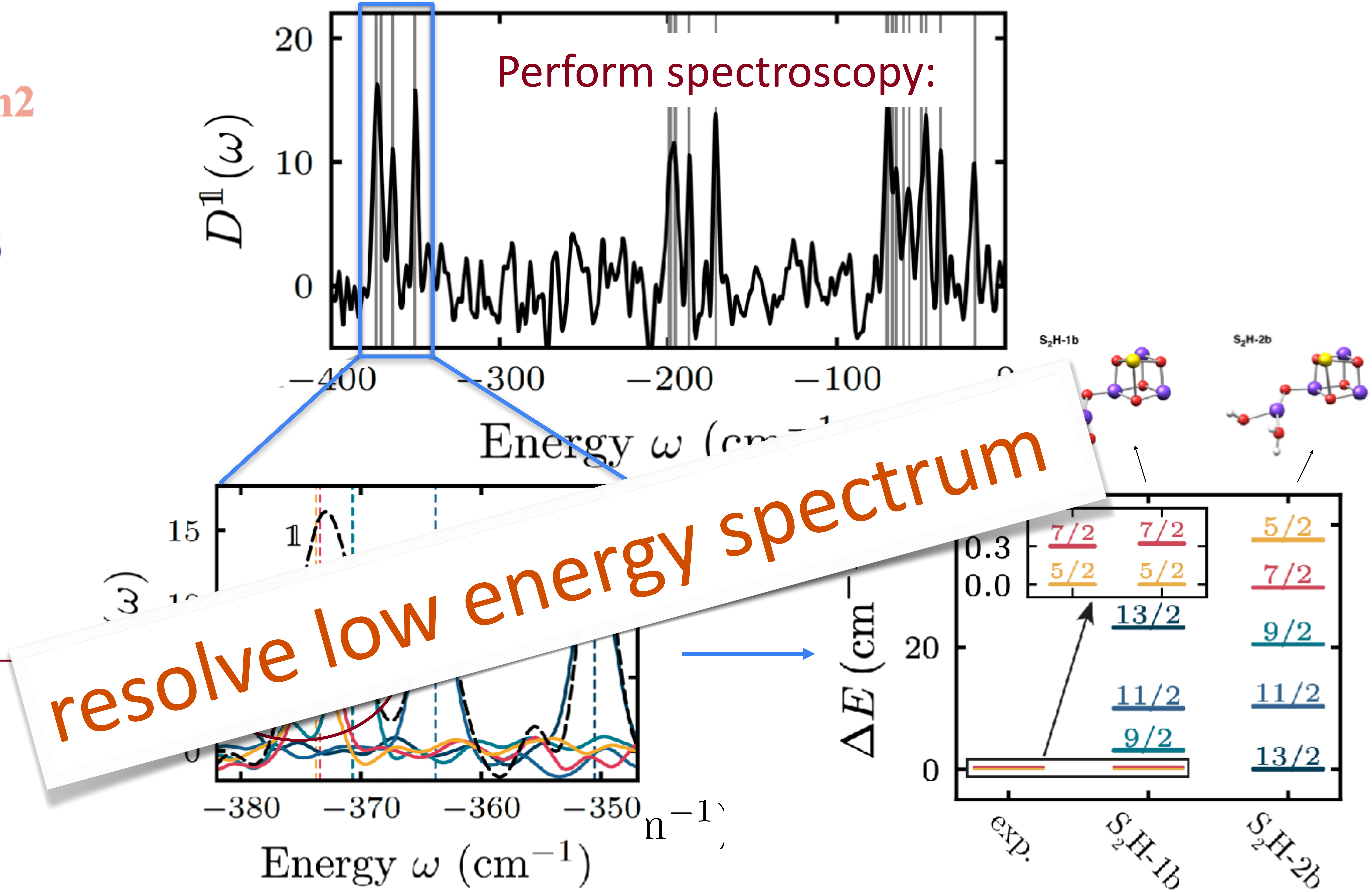
Perform spectroscopy:



spin-projection allows detection of tiny energy differences (0.3cm^{-1})

Benchmark on Larger Molecule (OEC Complex)



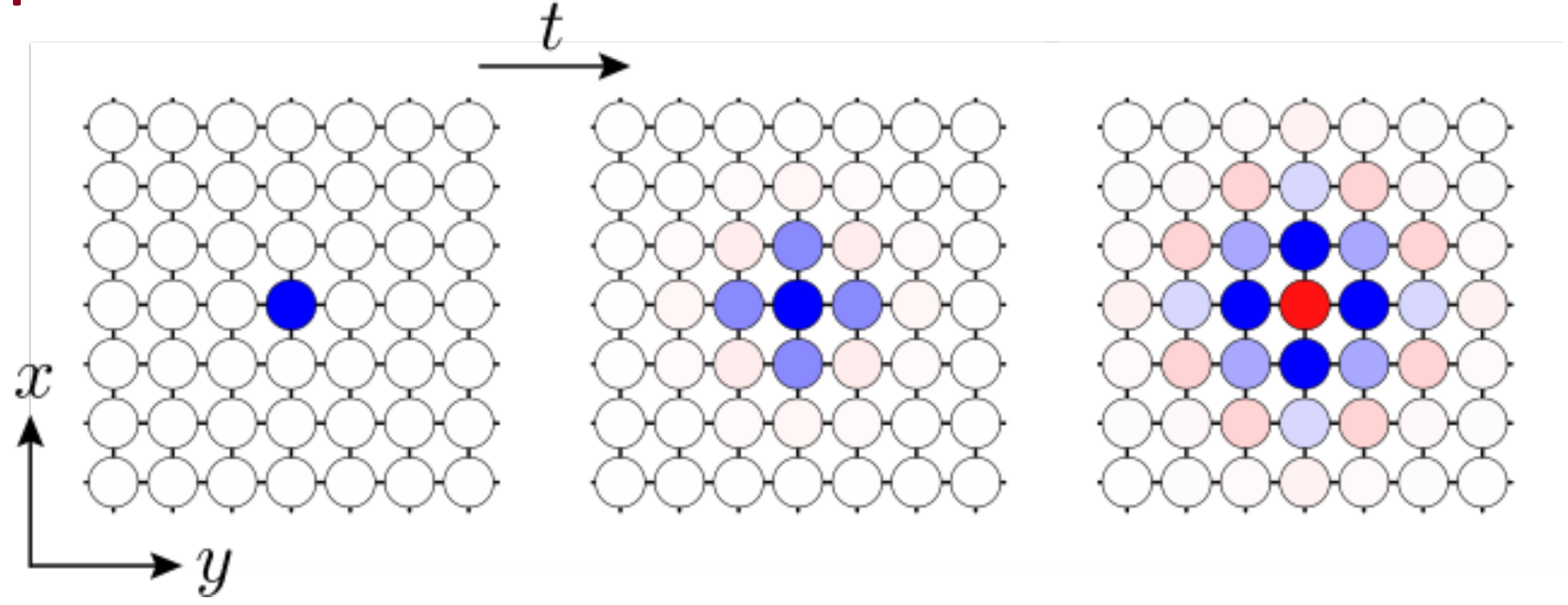


spin-projection allows detection of tiny energy differences (0.3cm^{-1})

Additional Applications and Outlook

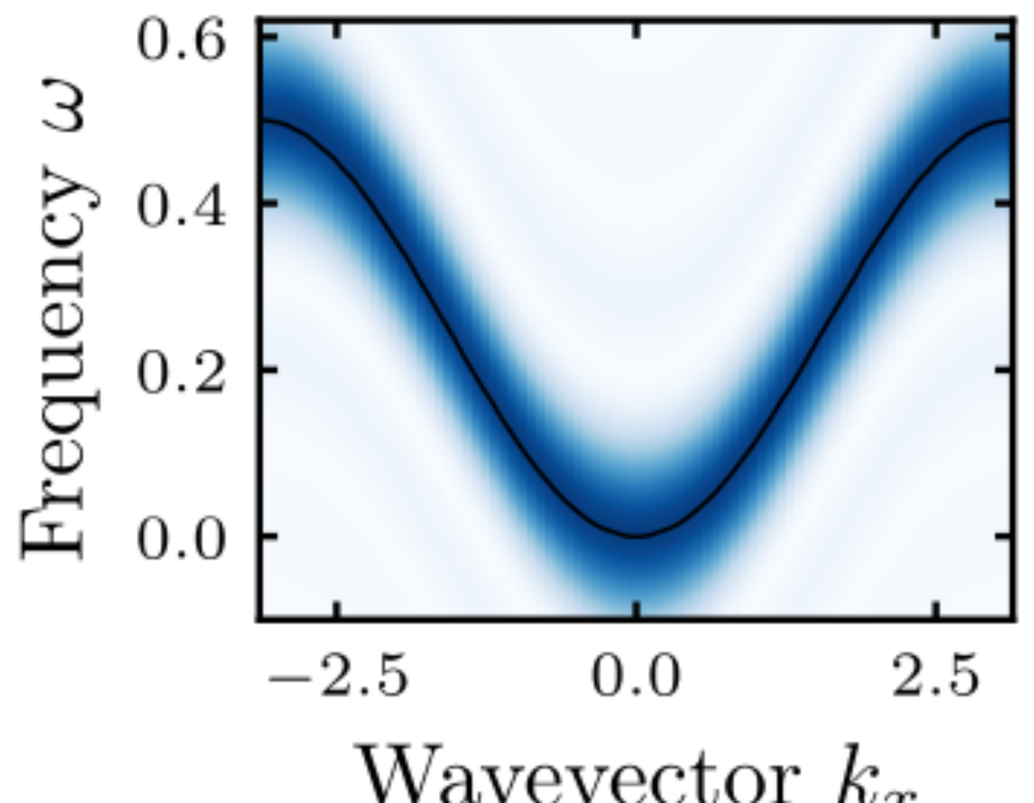
Application to 2D magnetic materials

Single-particle Green's function of FM Heisenberg:

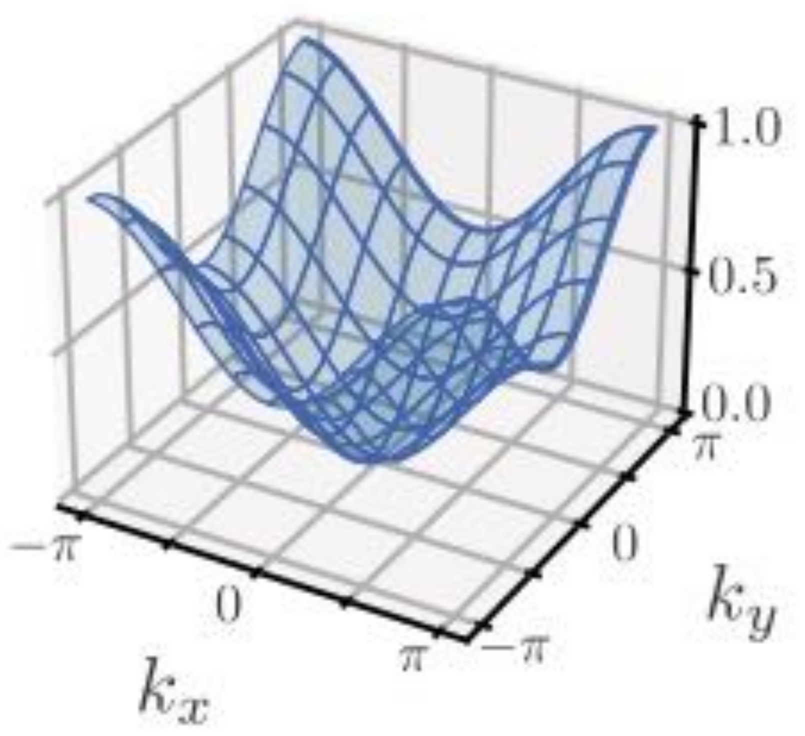


Quasi-particle properties encoded in spectral function

$$S(k,\omega) = |G(k,\omega)|^2$$



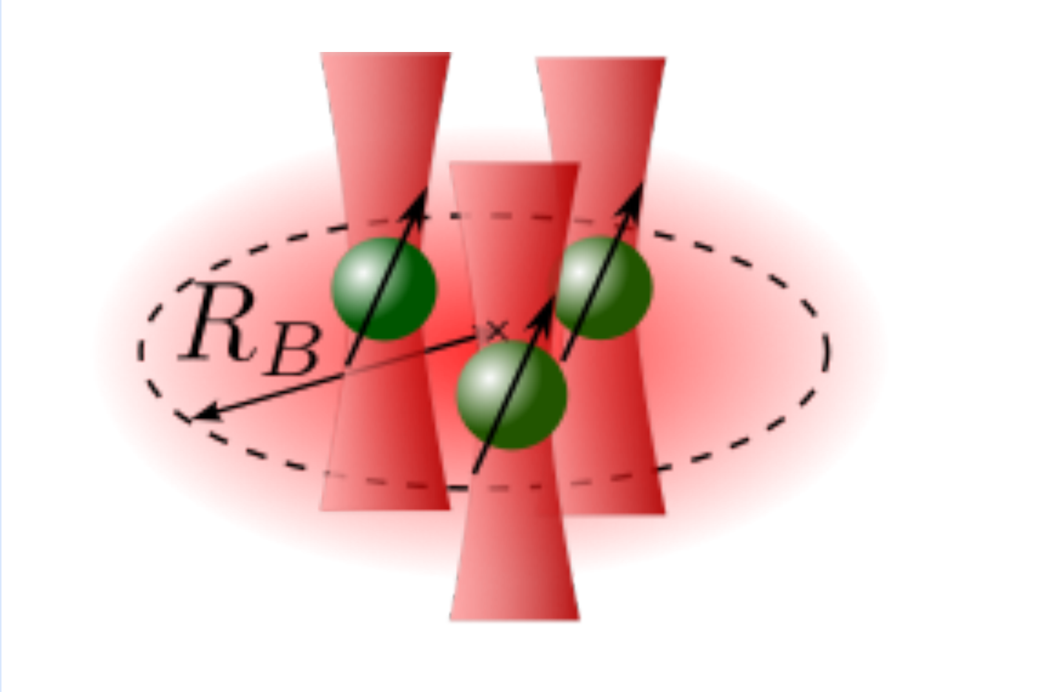
Dispersion $\epsilon(k)$



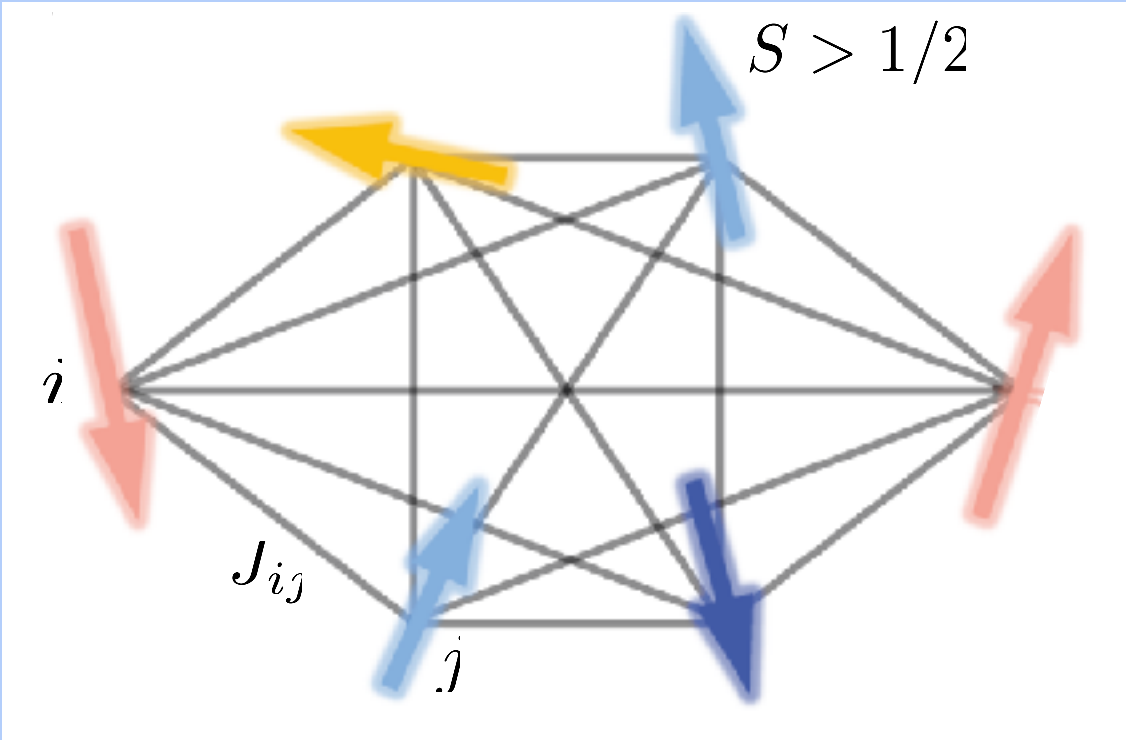
Next steps:

- include error correction
- dynamics of chemical reactions
- simulate fermions (e.g., Coulomb Hamiltonian)

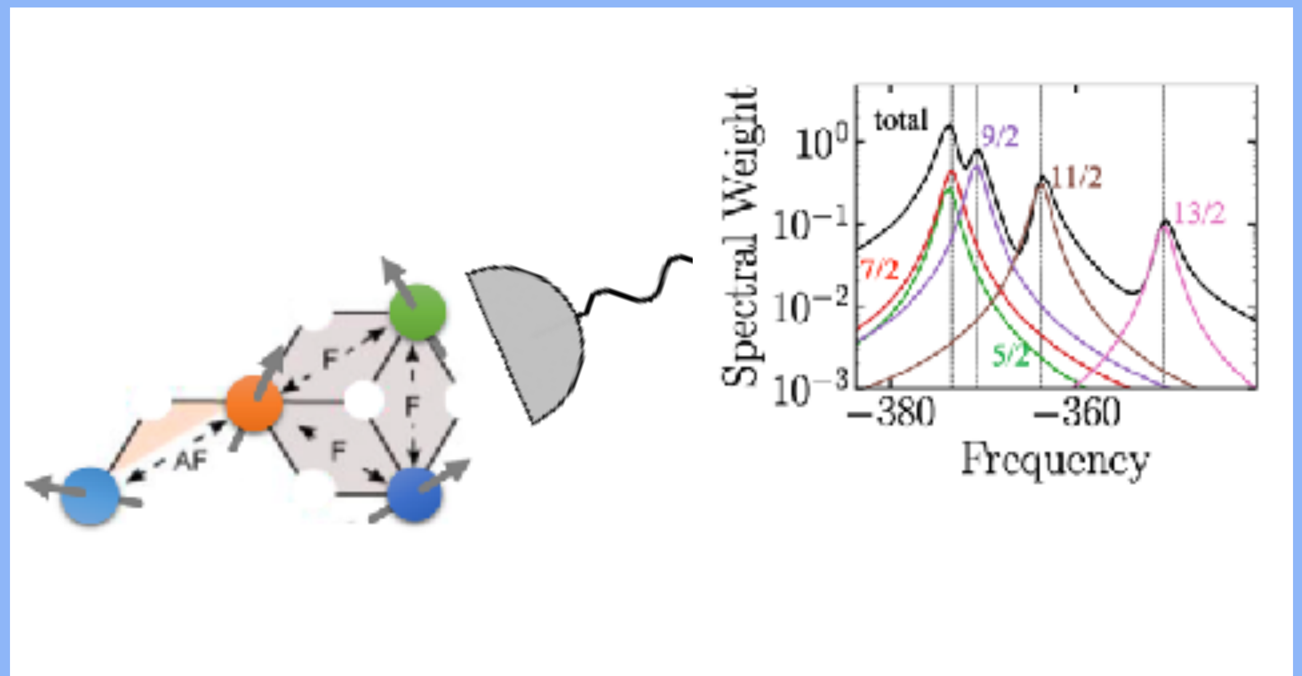
Control high spin (S>1/2)



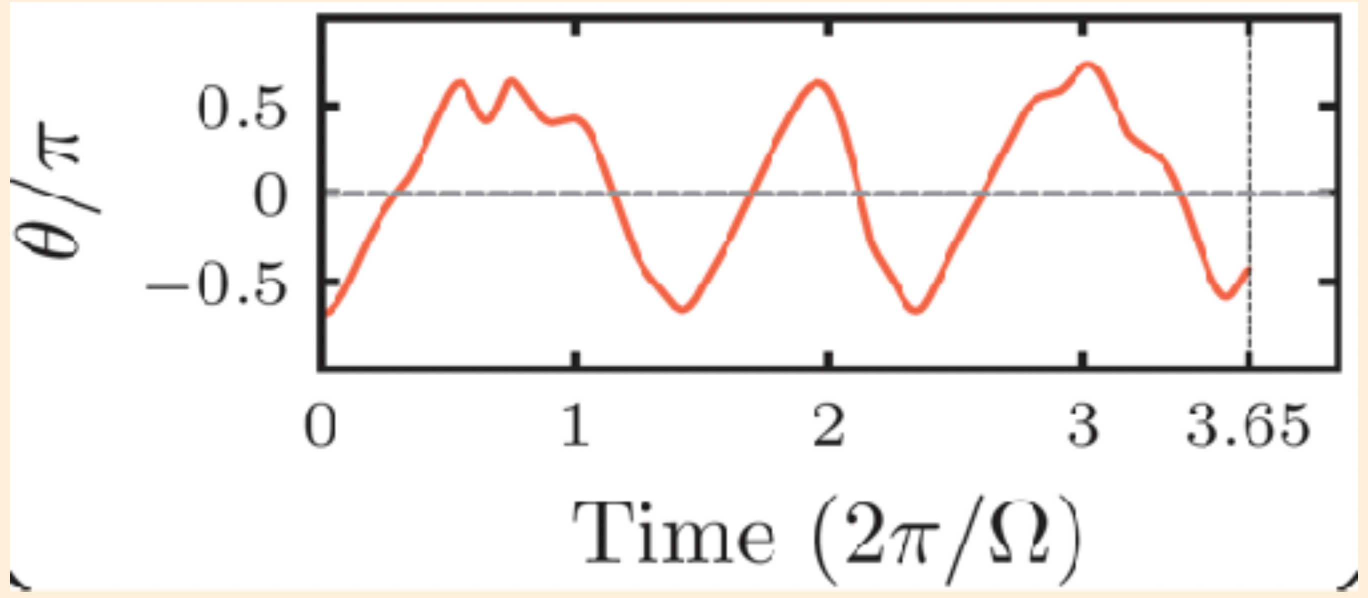
Non-local connectivity



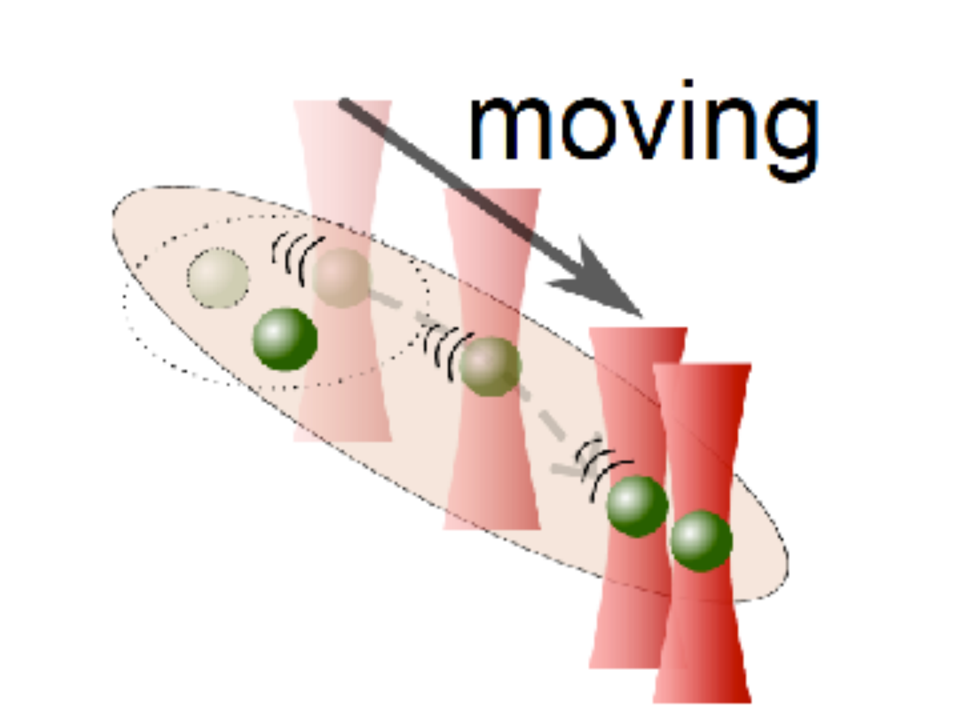
Solution read-out



Native multi-qubit gates

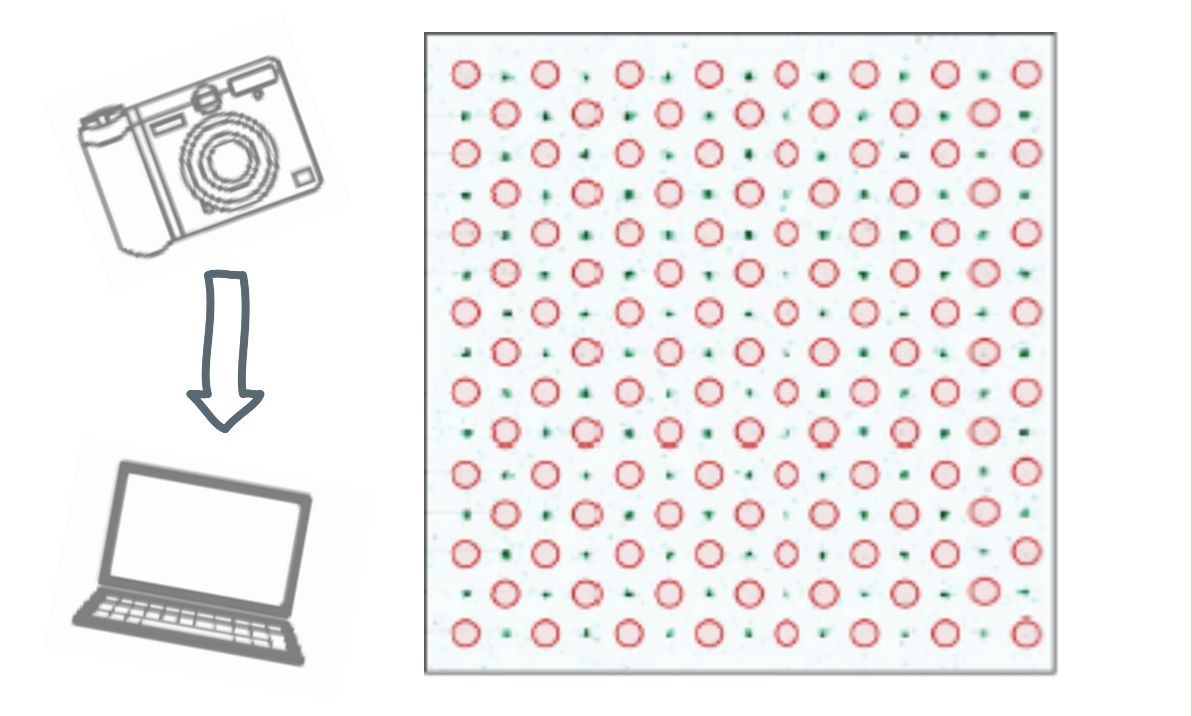


Atom moving



D. Bluvstein et al., Nature **604**, 451–456 (2022)

Co-processing



H.-Y. Huang et al., Nat. Phys. 16, 1050–1057 (2020)

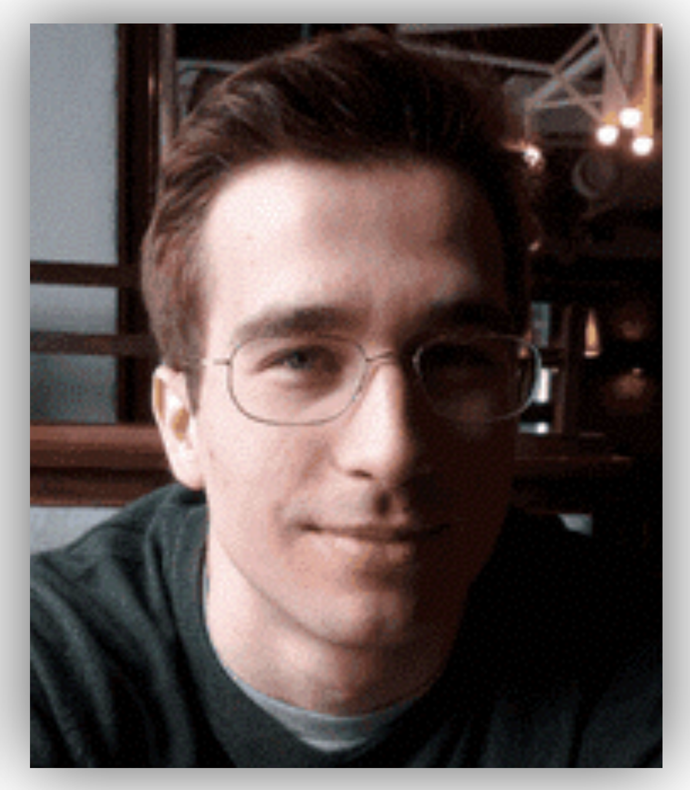
Maskara, Ostermann, Shee, Kalinowski, McClain Gomez, Araiza Bravo, Wang, Krylov, Yao, Head-Gordon, Lukin, Yelin, arXiv:2312.02265

Collaborators


Nishad Maskara
(Harvard)


Stefan Ostermann
(Harvard)


James Shee
(UC Berkeley
→ Rice)


Marcin Kalinowski
(Harvard)


Abigail McClain
Gomez
(Harvard)


Rodrigo Araiza Bravo
(Harvard)


Derek Wang
(Harvard
→ IBM Quantum)


Anna Krylov
(USC, Los Angeles)

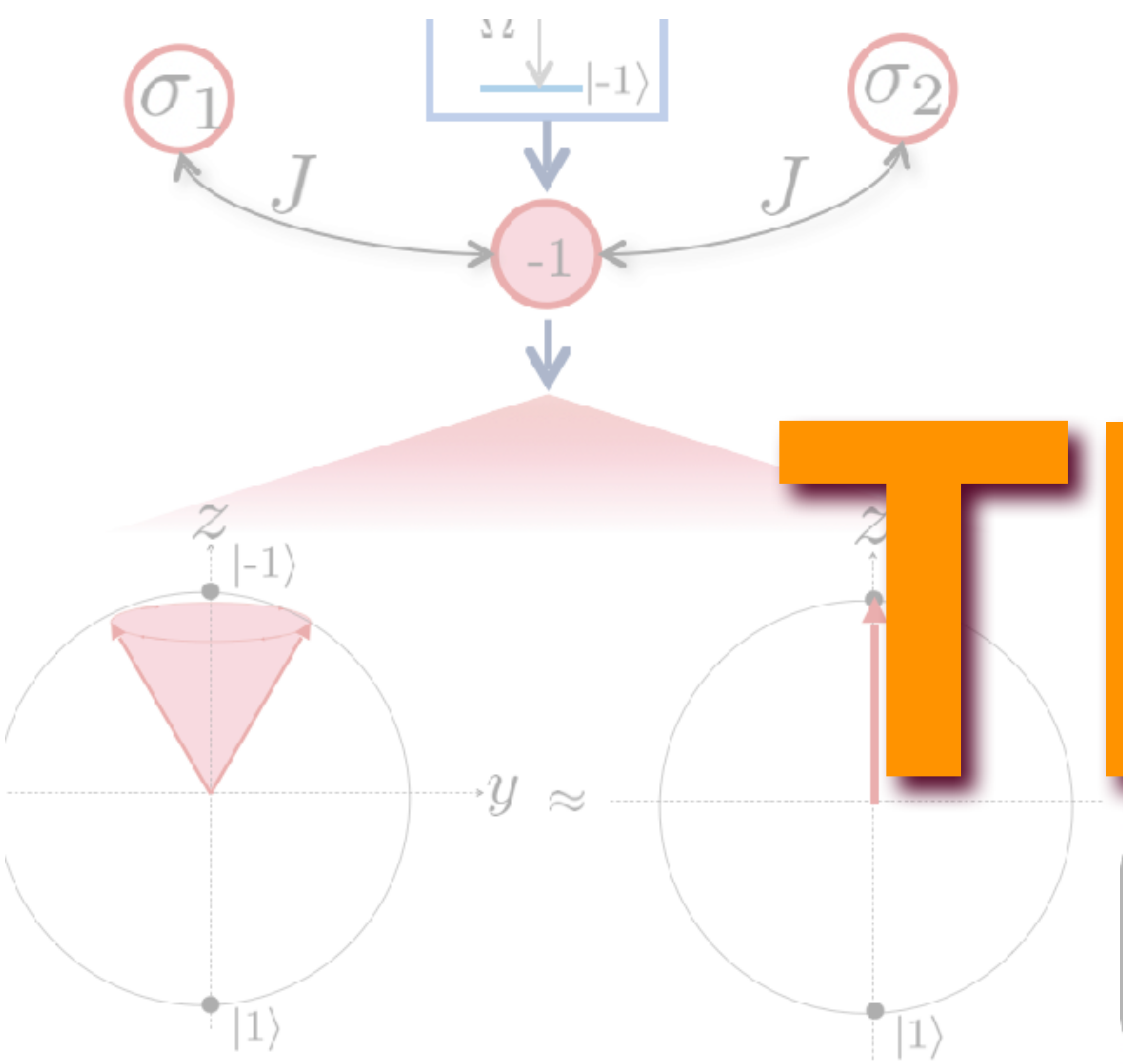

Norman Yao
(Harvard)

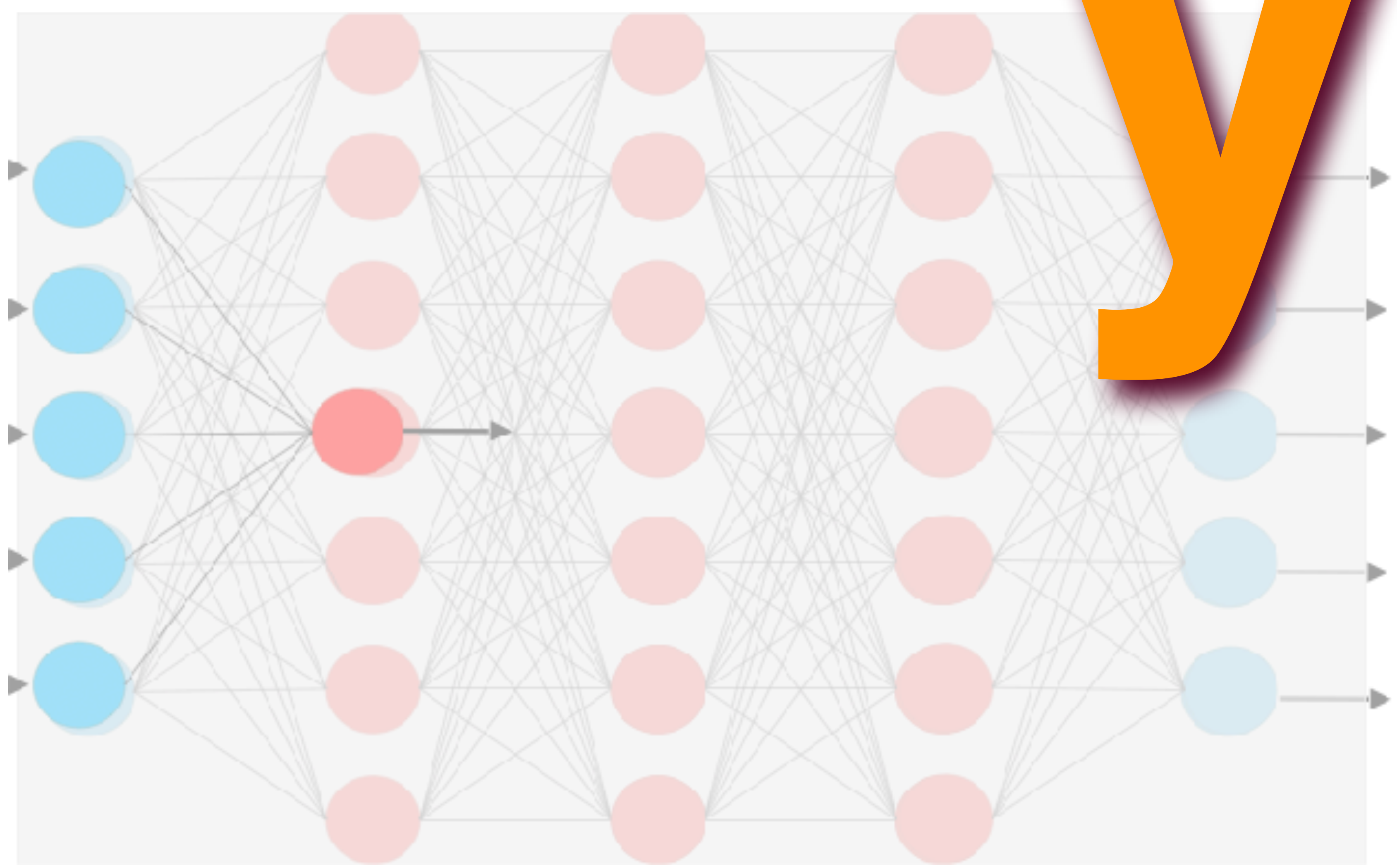

Martin Head-Gordon
(UC Berkeley)

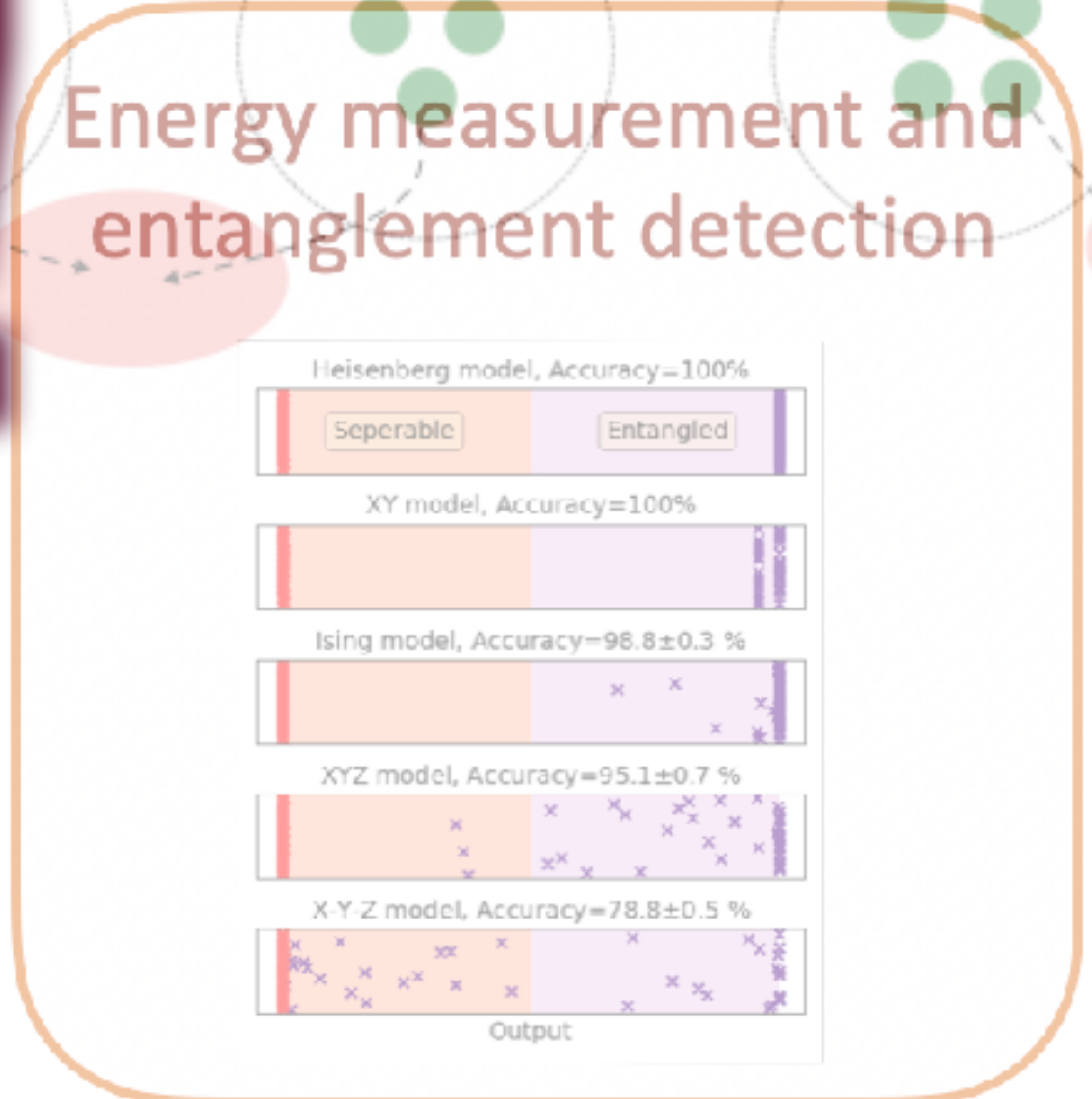

Mikhail Lukin
(Harvard)

And many thanks to: Dolev Bluvstein, Madelyn Cain, Joonho Lee, Nathan Leitao,, Kushal Seethram, and many more.

Thank you!






Energy measurement and entanglement detection
Heisenberg model, Accuracy=100%
Seperable
Entangled
XY model, Accuracy=100%
Ising model, Accuracy=98.8±0.3 %
XYZ model, Accuracy=95.1±0.7 %
X-Y-Z model, Accuracy=78.8±0.5 %
Output