Fullqubit alchemist:

Quantum algorithm for alchemical free energy calculations

Po-Wei Huang[1,2], *Gregory Boyd*[1], Gian-Luca R. Anselmetti[3], Matthias Degroote[3], Nikolaj Moll[3], *Raffaele Santagati*[3], Michael Streif[3], Benjamin Ries[3], Daniel Marti-Dafcik[1,2], Hamza Jnane[1,2], Sophia Simon[4], Nathan Wiebe[4,5,6], Thomas R. Bromley[2], *Bálint Koczor*[1,2]

[1]University of Oxford, [2]Quantum Motion, [3]Boehringer Ingelheim, [4]University of Toronto, [5]PNNL, [6]CIFAR

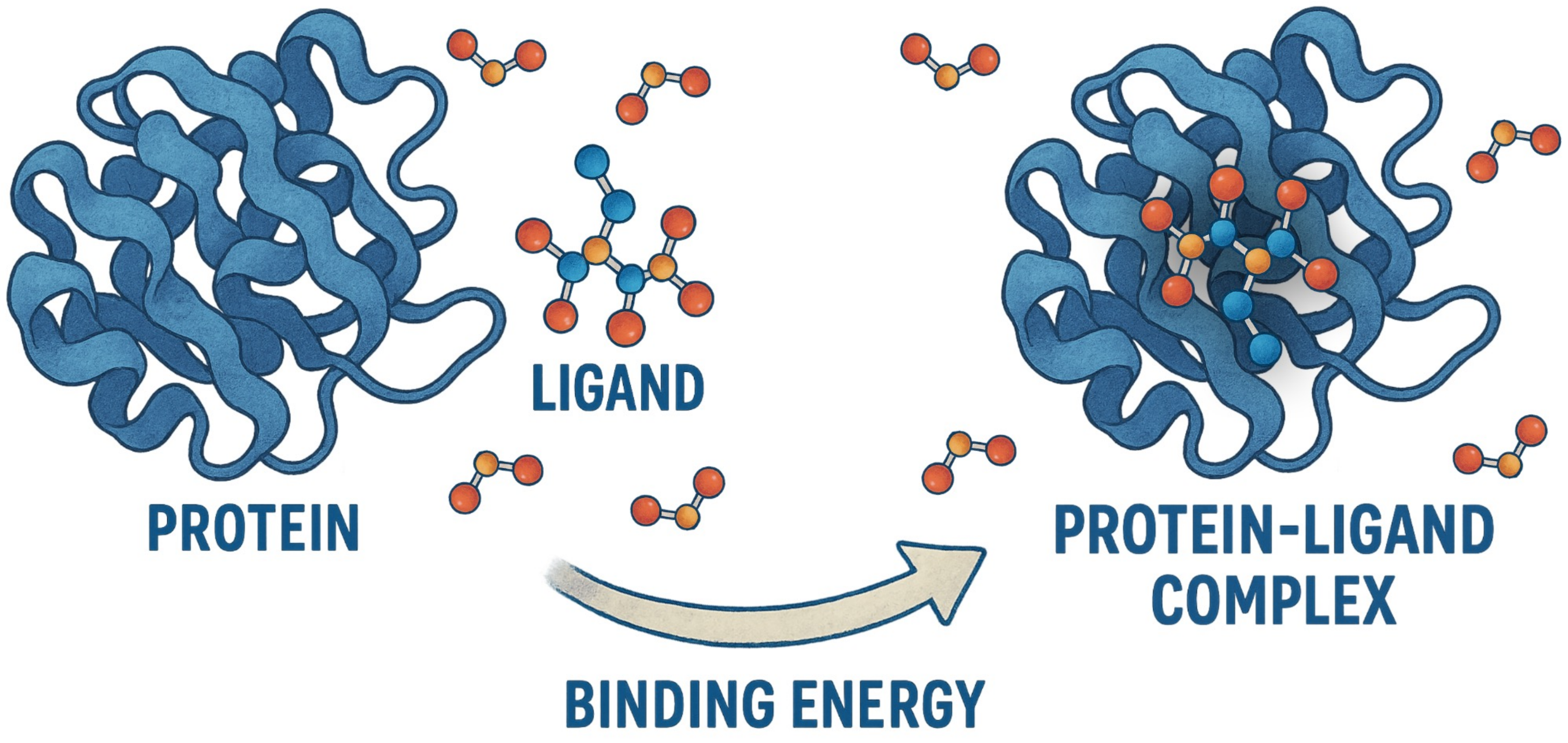
University of Oxford

Quantum Motion

Drug design

Motivation and applications

- Binding free energy is used to rank drug candidates.
- Free energy is computed via thermal averages over many many many different molecular configurations.



Talk questions and outline

PART I: Molecular dynamics on a quantum computer

How do we generate the various molecular configurations on the quantum computer?

PART II: Alchemical free energy calculations

How do we calculate the relative free energy between two different systems?

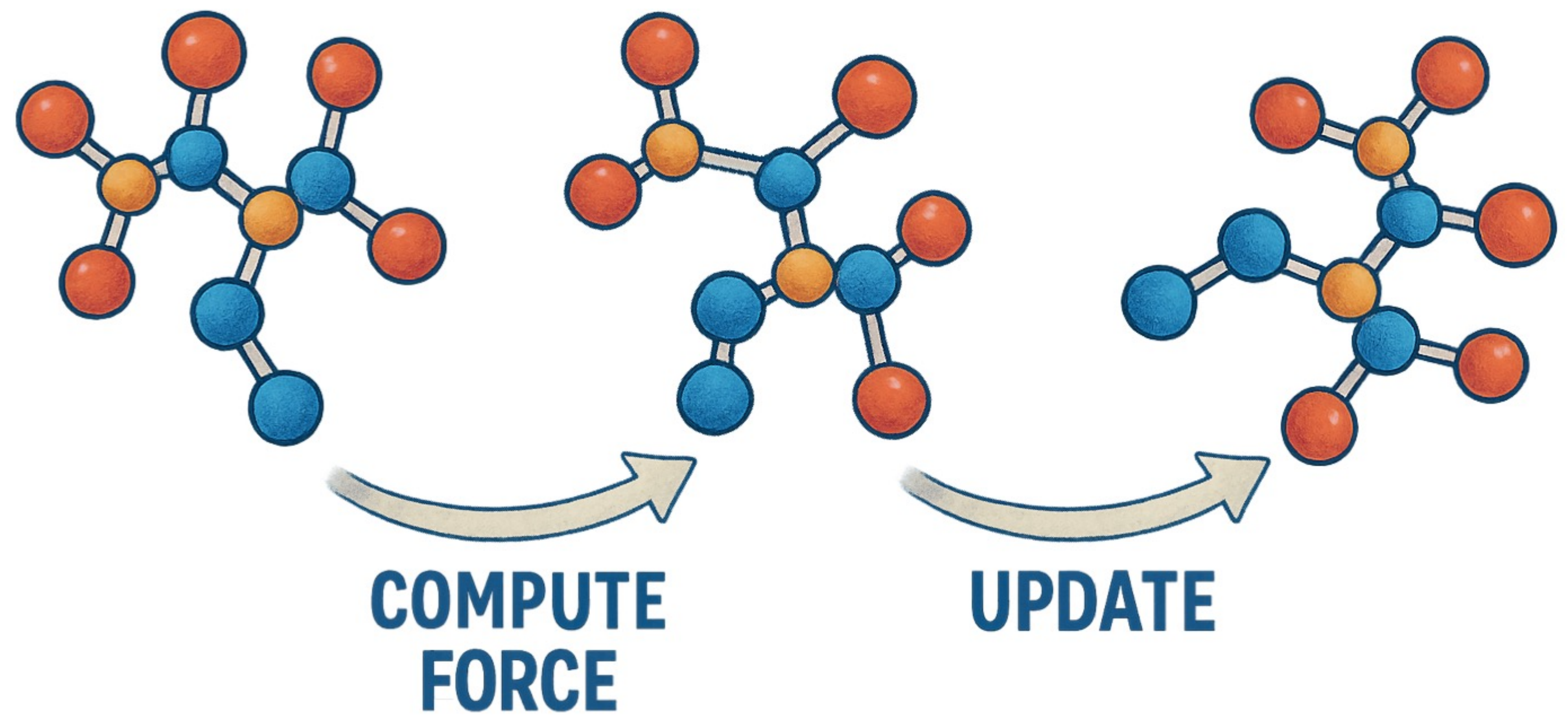
Part I:

Molecular Dynamics on a Quantum Computer

Molecular dynamics

In practice for classical computers

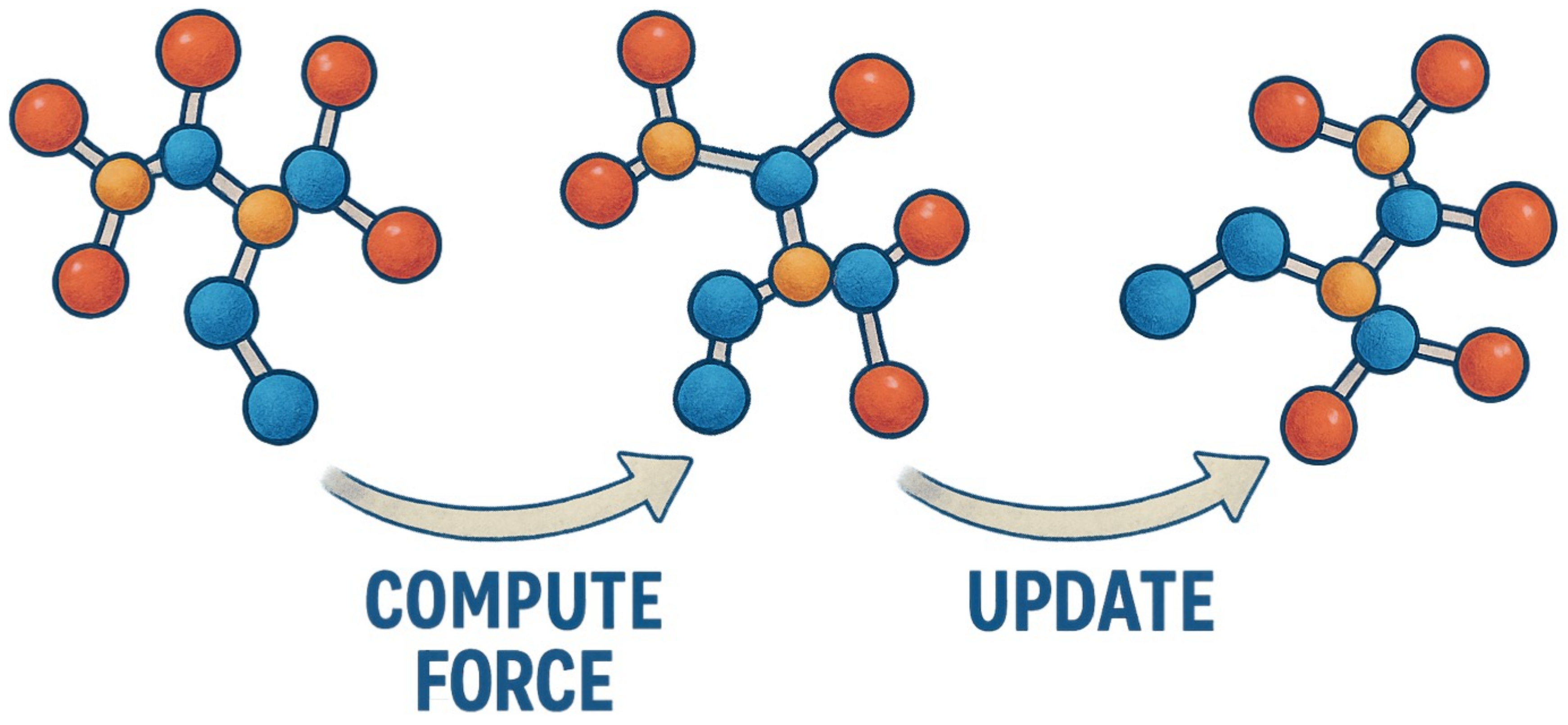
- Propagate particles with Hamiltonian dynamics
- Time average of evolution trajectory to replace ensemble average
  - Lots and lots of compute time!
- Used to compute thermal averages such as free energy

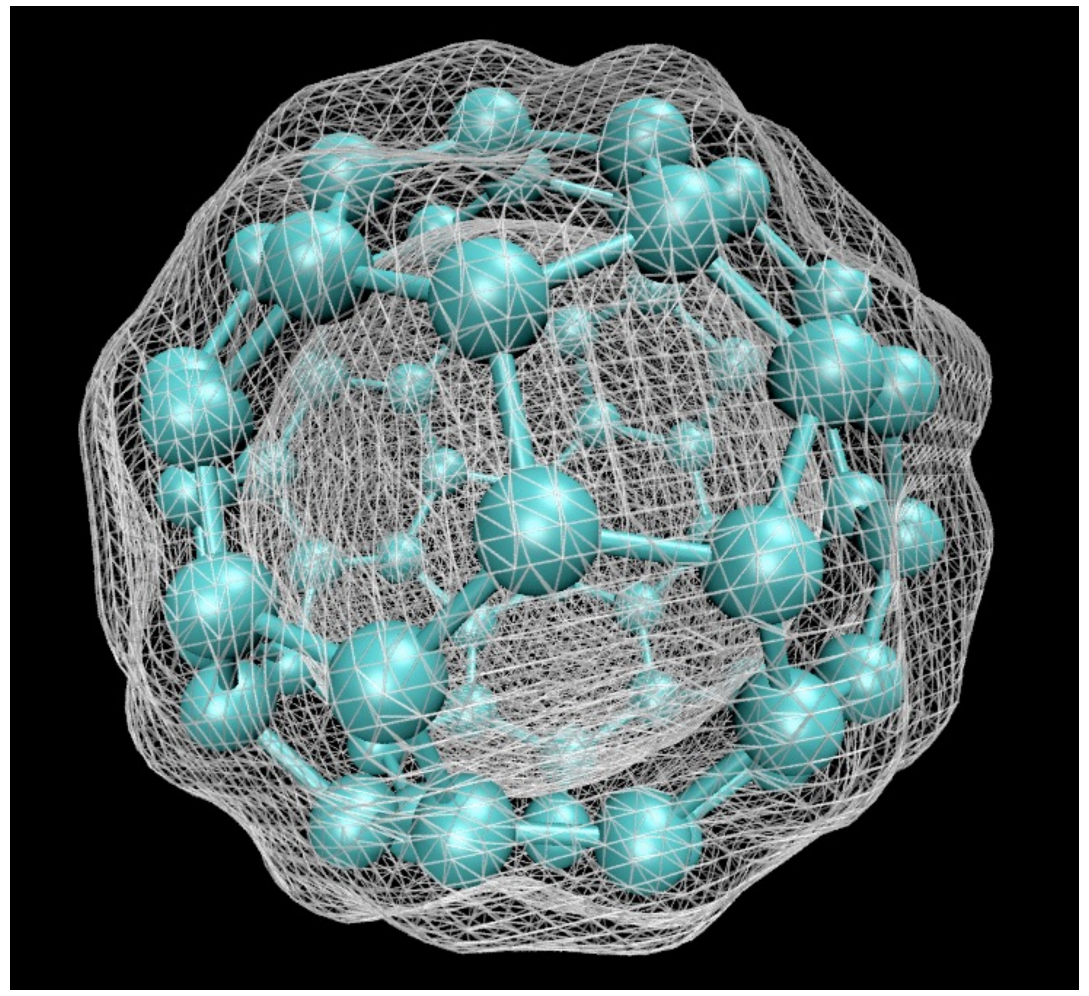


Hybrid quantum semi-classical modeling

Not all particles are classical…

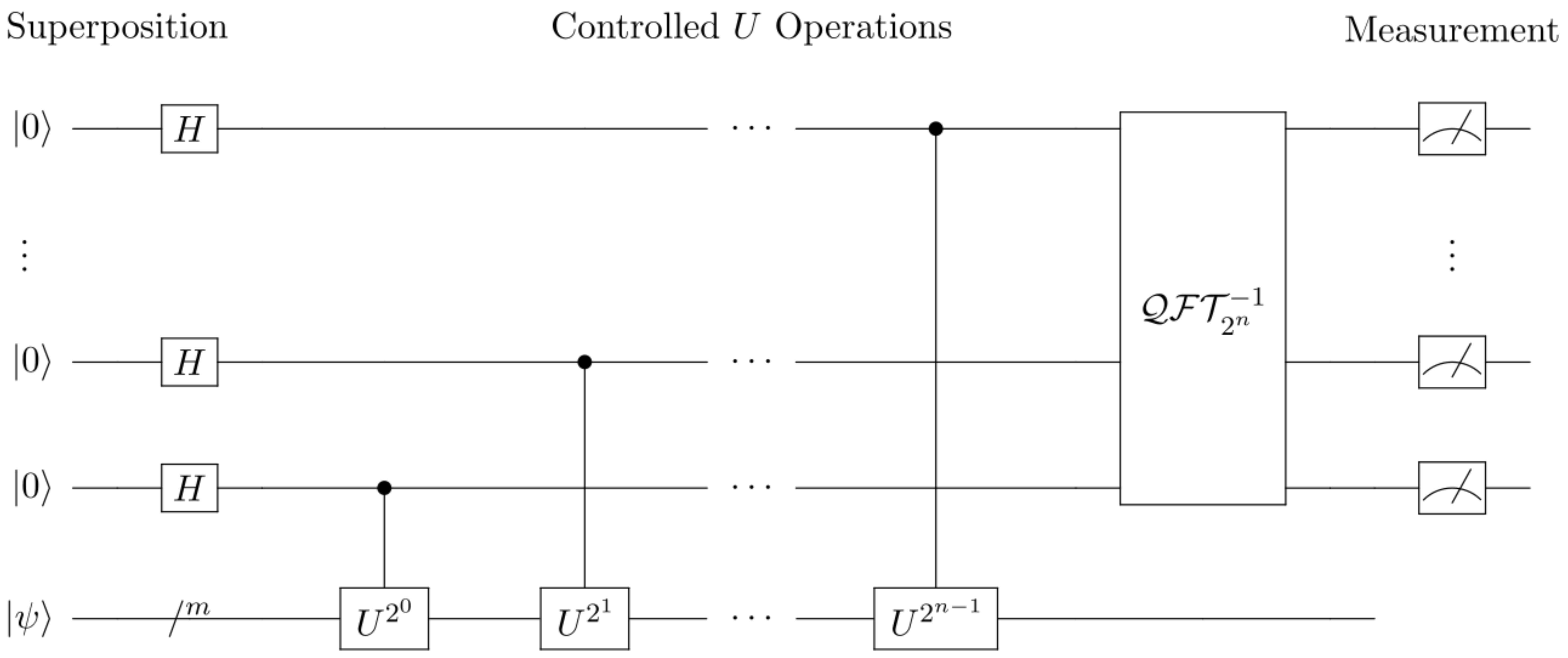
- While nuclei are heavy enough to be considered classical, electrons whizzing around definitely are not…
- Born-Oppenheimer Approximation:
    - The wave function of the *nuclei* and *electrons* are treated separately.
        - Treat nuclei *classically* → propagate by molecular dynamics
        - Treat electrons *quantumly* → electronic structure calculations (eg DFT, CCSD) of the ground state





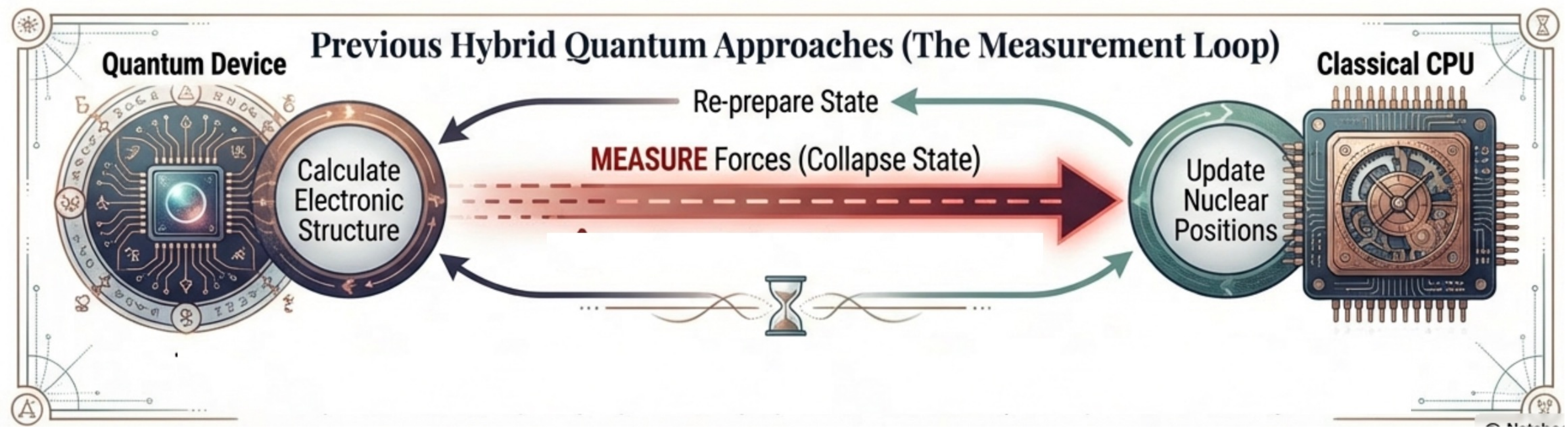
Calculating ground state properties…

Seem familiar?



- Offload ground state energy calculations to quantum computer
- Feed ground state energy calculations to MD step

Case closed?



Is there a coherent approach that doesn't require hybrid loops?

Can we implement molecular dynamics on a quantum computer?

PRX QUANTUM **5,** 010343 (2024)

Improved Precision Scaling for Simulating Coupled Quantum-Classical Dynamics

Sophia Simon,[1,*] Raffaele Santagati,[2,†] Matthias Degroote,[2] Nikolaj Moll,[2] Michael Streif,[2] and Nathan Wiebe[3,4,5,‡]

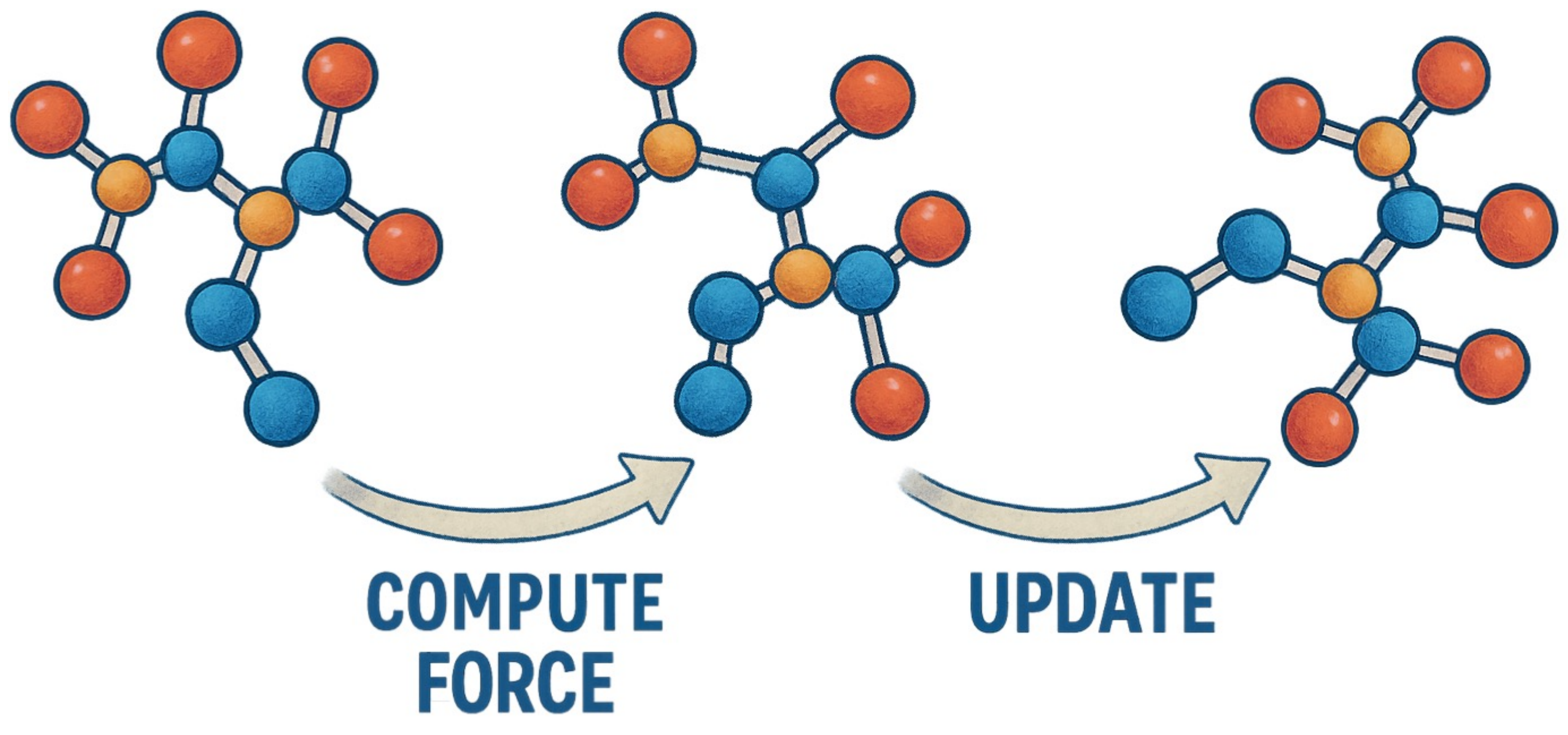
[1] *Department of Physics, University of Toronto, Toronto, Ontario M5S 1A7, Canada*
[2] *Quantum Lab, Boehringer Ingelheim, Ingelheim am Rhein 55218, Germany*
[3] *Department of Computer Science, University of Toronto, Toronto, Ontario M5S 2E4, Canada*
[4] *Pacific Northwest National Laboratory, Richland, Washington 99354, USA*
[5] *Canadian Institute for Advanced Studies, Toronto, Ontario M4G 4G8, Canada*

(Received 27 September 2023; accepted 12 February 2024; published 12 March 2024)

Molecular dynamics on a quantum computer

What works and what doesn't

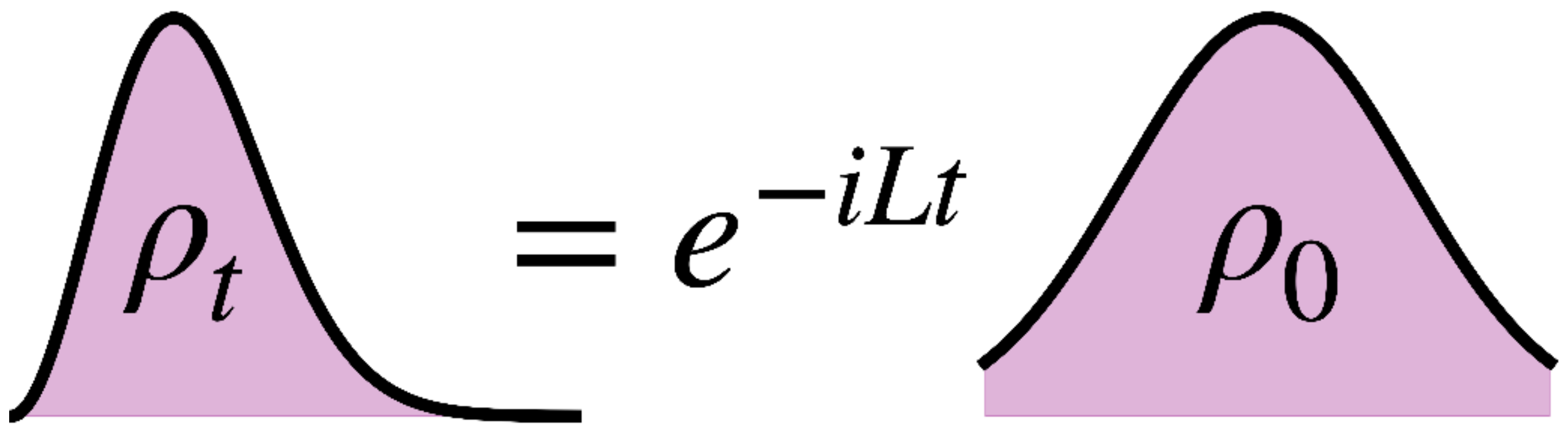
- Large Hilbert space with small number of qubits
- Can simulate/discretize the entire phase-space
- A quantum state can be used to encode an entire probability distribution
- Time-based averages may be hard to implement



Koopman von Neumann mechanics

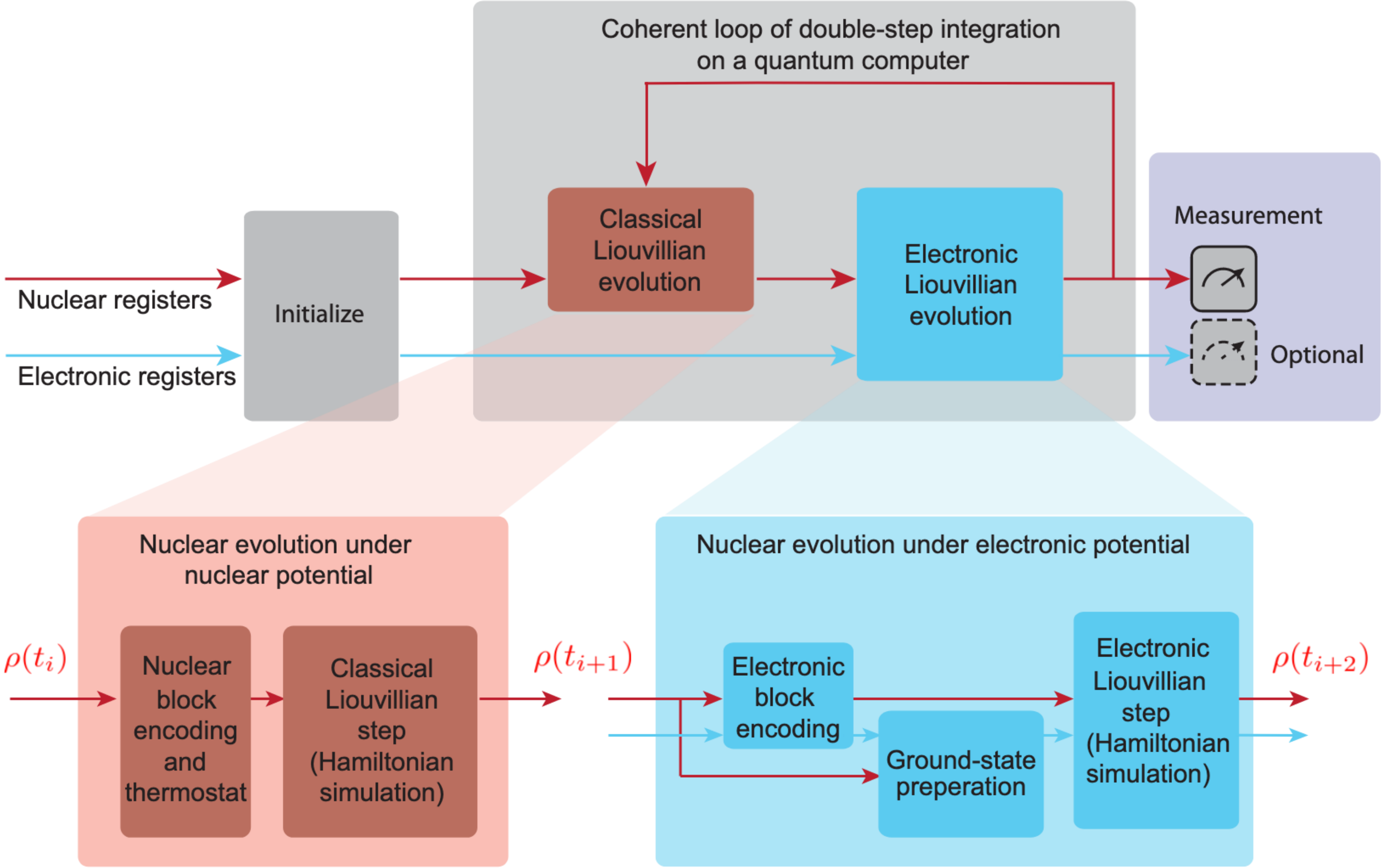
Under the Liouvillian picture

- Describes classical dynamics over a classical probability distribution of a statistical ensemble
- Classical wavefunction $\rho = \psi^*_{KvN}(p, q, t)\psi_{KvN}(p, q, t)$ that encodes distribution
- Evolution of wavefunction by Liouvillian $L = -i\left(\frac{\partial H}{\partial p}\frac{\partial}{\partial x} - \frac{\partial H}{\partial x}\frac{\partial}{\partial p}\right)$
- Evolution dynamics governed by $i\frac{\partial}{\partial t}\psi_{KvN} = L\psi_{KvN}$

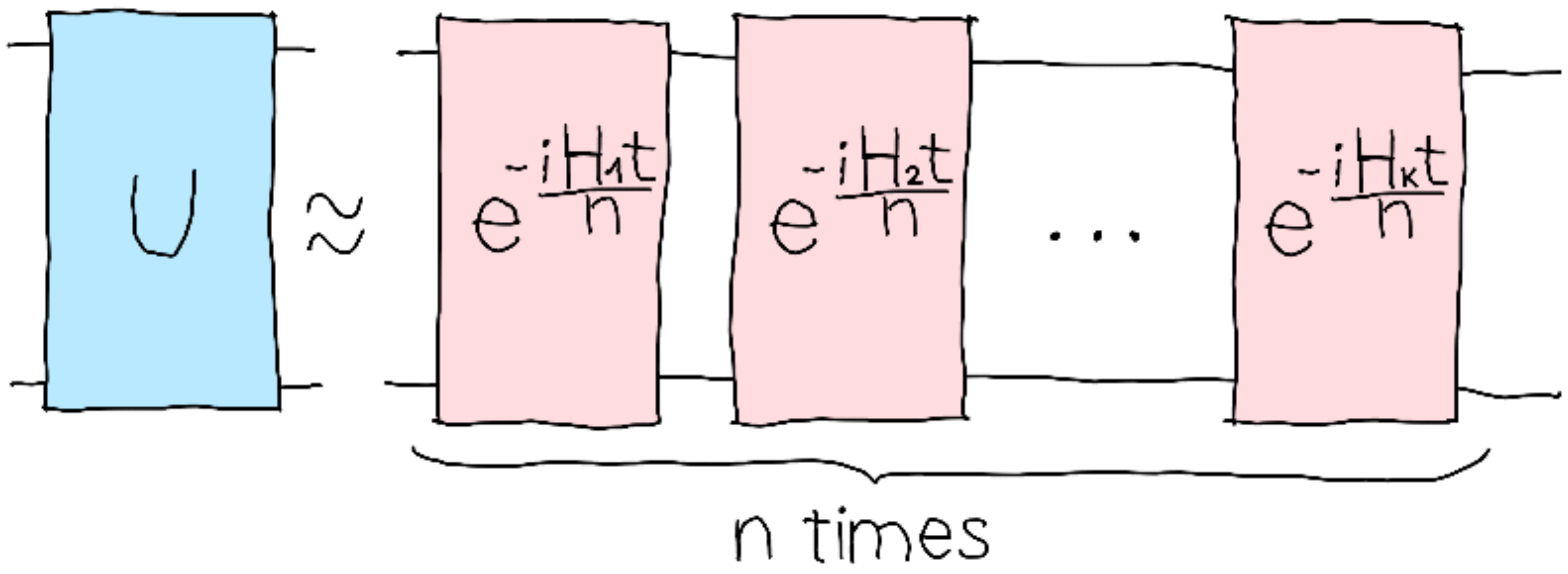


Simulating coupled quantum-classical dynamics

- Separate Liouvillian into classical and electronic parts, and evolve with small time steps individually.
- No more hybrid loop, but alternating time steps on a quantum computer



Case closed again?

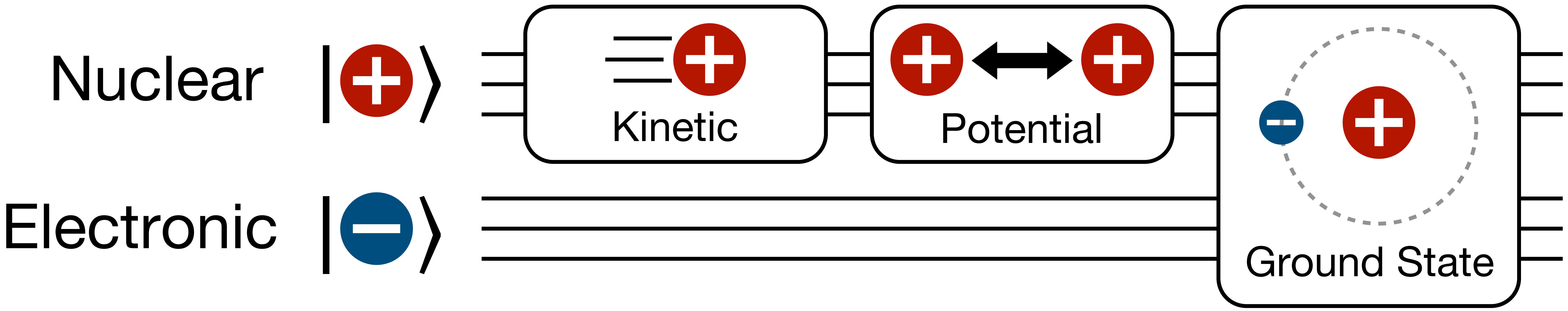


Can we do better than Trotterizing the Liouvillian?

Directly implementing the Liouvillian

Or, how to block-encode the Liouvillian

- Two register sets on quantum computer to implement the Liouvillian:
    - "Classical" nuclear register – Corresponds to classical MD in Liouvillian formalism on exponential sized grid
    - "Quantum" electronic register – Corresponds to ground state energy/forces calculations



$$L = -i\left(\frac{\partial H}{\partial p}\frac{\partial}{\partial x} - \frac{\partial H}{\partial x}\frac{\partial}{\partial p}\right)$$

Writing down the terms

Aka the slide with a lot of math

$$H_{\text{ext}}^{(\text{NVT})} = \sum_{n=1}^{N}\sum_{j=1}^{3} \frac{p_{n,j}'^2}{2m_n s^2} + \sum_{k=1}^{N-1}\sum_{n=k+1}^{N} \frac{Z_n Z_k}{\|x_n - x_k\|} + \frac{p_s^2}{2Q} + N_f k_B T \ln(s) + E_{\text{el}}(\boldsymbol{x}),$$

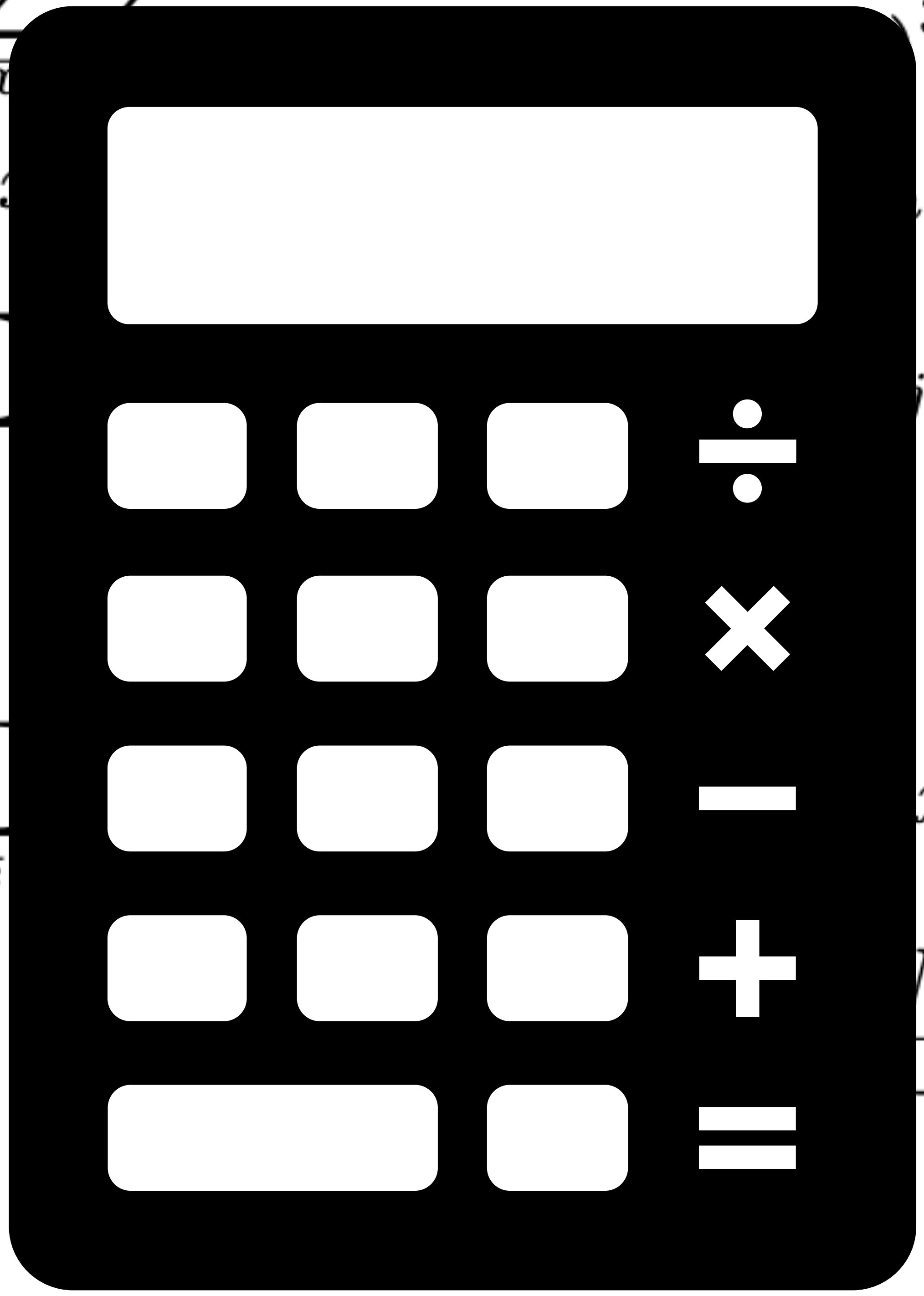
$$L_{\text{cl,disc}} := -i \sum_{n=1}^{N}\sum_{j=1}^{3} \left(D_{x_{n,j}} \otimes \frac{\partial H_{\text{cl}}}{\partial p'_{n,j}} - \frac{\partial H_{\text{cl}}}{\partial x_{n,j}} \otimes D_{p'_{n,j}} \right) - i \left(D_s \otimes \frac{\partial H_{\text{cl}}}{\partial_{p_s}} - \frac{\partial H_{\text{cl}}}{\partial_s} \otimes D_{p_s} \right), \qquad \text{(A.11)}$$

$$L_{\text{el}} = L - L_{\text{cl}} := i \sum_{n=1}^{N}\sum_{j=1}^{3} \frac{\partial E_{\text{el}}(\boldsymbol{x})}{\partial x_{n,j}} \partial_{p'_{n,j}}$$

$$\frac{\partial H_{\text{cl}}}{\partial x_{n,j}} = \sum_{n' \neq n} \sum_{\bar{x}} \sum \frac{-Z_n Z_{n'}}{\cdots^{3/2}} \times (\cdots \rangle\langle \bar{x}_{n'}| \qquad \text{(A.12)}$$

$$\frac{\partial H_{\text{cl}}}{\partial p'_{n,j}} = \sum_{\bar{p}'_{n,j}} \sum_{\bar{s}} \cdots | \otimes |\bar{s}\rangle\langle \bar{s}| \qquad \text{(A.13)}$$

$$\frac{\partial H_{\text{cl}}}{\partial s} = \sum_{\bar{p}_{n,j}} \sum_{\bar{s}} \cdots | \otimes |\bar{s}\rangle\langle \bar{s}| \cdots \frac{\cdots N_f k_B T}{\cdots + s_{\min}} |\bar{s}\rangle\langle \bar{s}| \qquad \text{(A.14)}$$



$$\frac{\partial H_{\text{cl}}}{\partial p_s} = \sum_{\bar{p}_s} \frac{p_s}{Q} |\bar{p}_s\rangle\langle \bar{p}_s| \qquad \text{(A.15)}$$

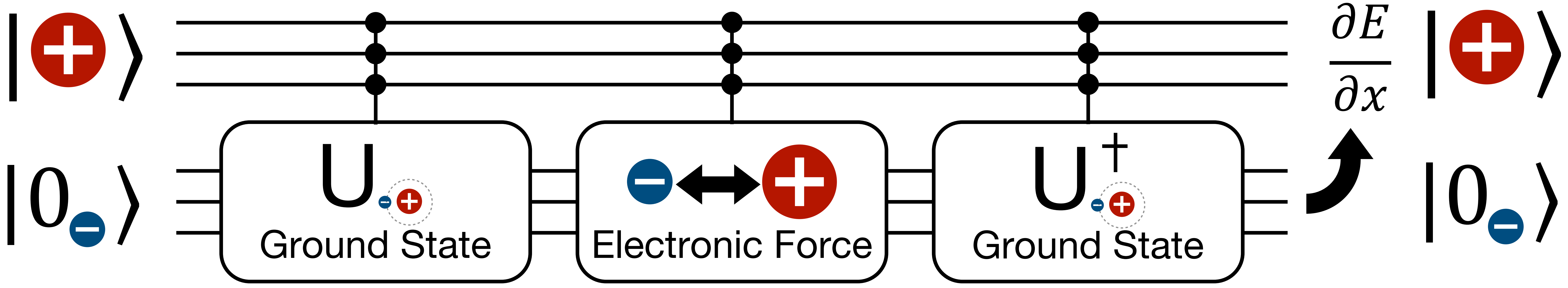
Block-encoding energies and forces

Obtaining these values without QPE

- Obtain quantum mechanical electronic forces via kicking back results from electronic register to nuclear register
- Compute ground state energy by preparing ground state, applying the Hamiltonian, then unpreparing the ground state
- Compute electronic forces with Hellmann-Feynman theorem and analytical derivative of first-quantized Hamiltonian

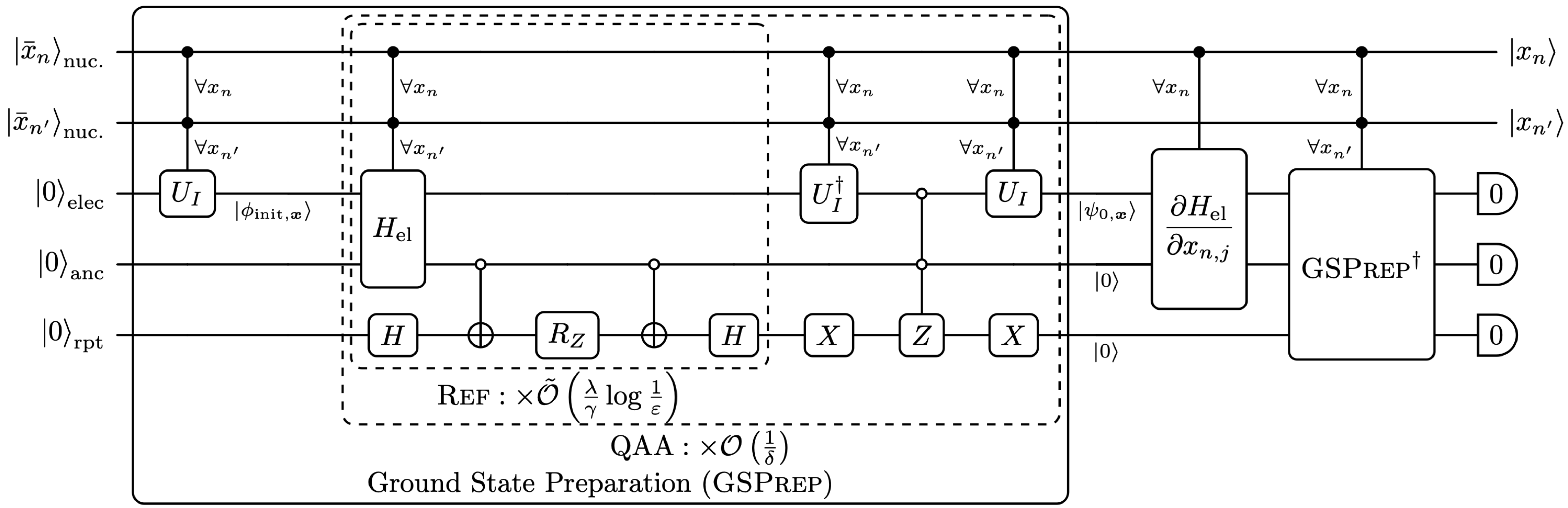
$$E = \langle \psi_g | H | \psi_g \rangle$$

$$\frac{\partial E}{\partial x} = \left\langle \psi_g \middle| \frac{\partial H}{\partial x} \middle| \psi_g \right\rangle$$



Implementation of energy derivatives

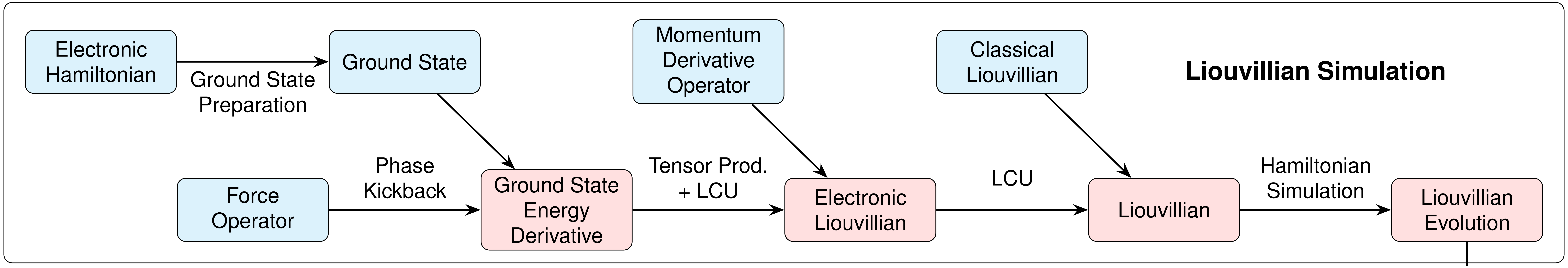
- Block-encoding of first-quantized Hamiltonian derivative found in [OSR+22]
- Ground state preparation algorithm by [LT20]: Eigenstate filtering+AA



Tying everything together…

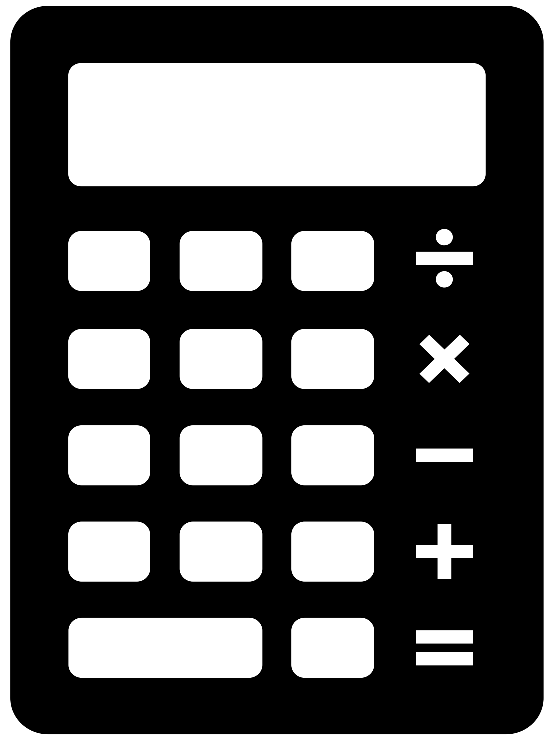
In Part I…

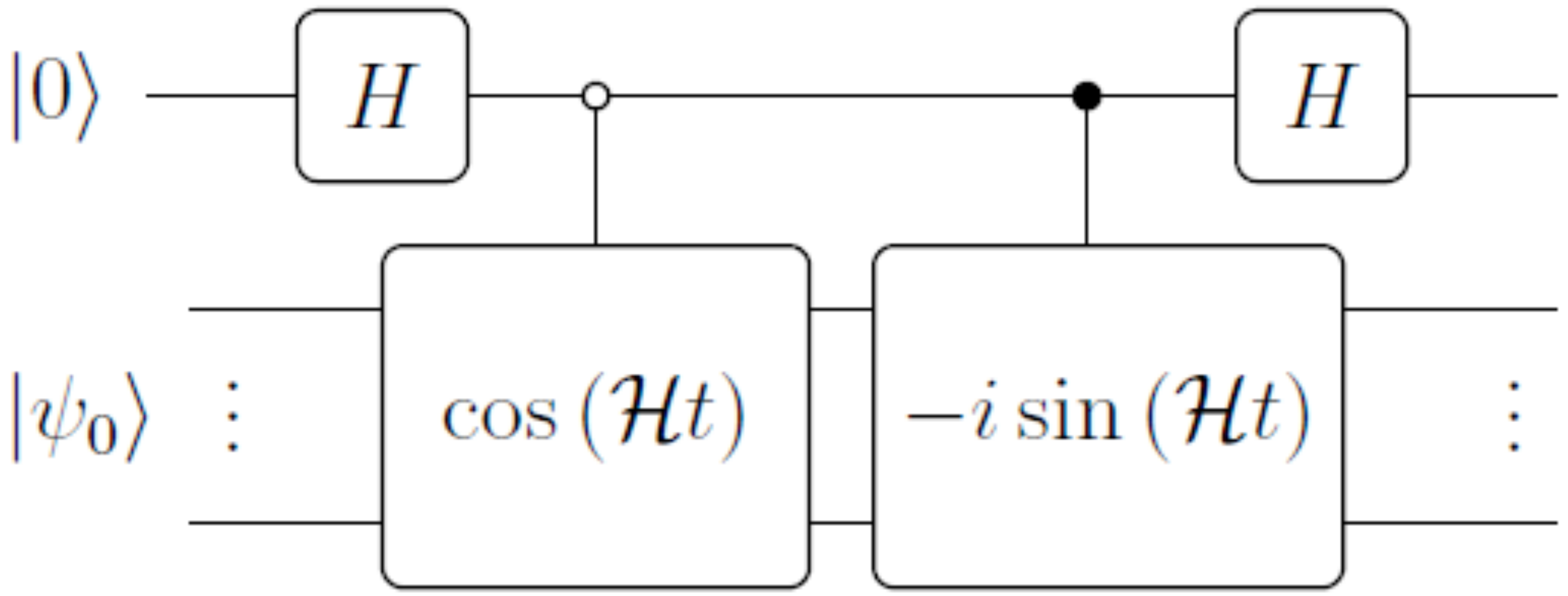
- Liouvillian simulation via QSVT-based Hamiltonian simulation



$$\frac{\partial E}{\partial x} = \left\langle \psi_g \middle| \frac{\partial H}{\partial x} \middle| \psi_g \right\rangle$$

$$L_{\mathrm{el}} = L - L_{\mathrm{cl}} := i \sum_{n=1}^{N} \sum_{j=1}^{3} \frac{\partial E_{\mathrm{el}}(\boldsymbol{x})}{\partial x_{n,j}} \partial_{p'_{n,j}}$$





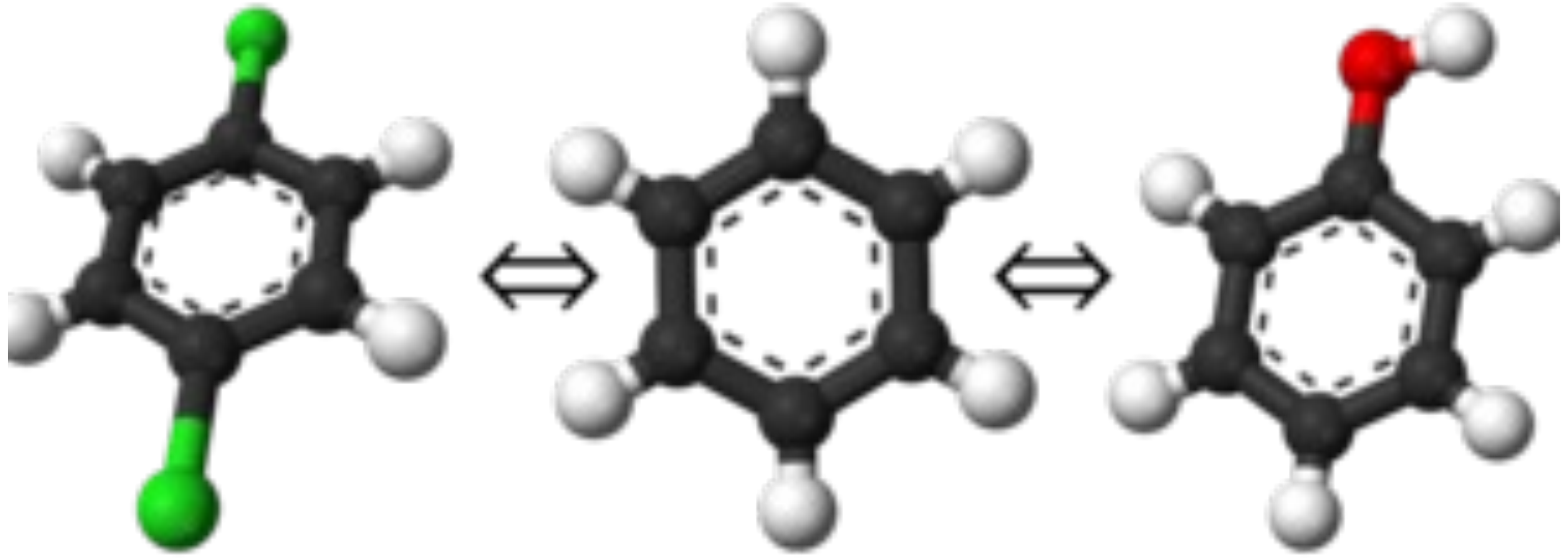
Part II:
Alchemical Free Energy Calculation

Thermodynamic integration

The alchemical way to calculate free energy

$$F = -k_B T \ln Z$$

- Absolute free energy difficult to compute
  - Either compute the entropy or partition function



$$\begin{aligned}
\Delta F_{A\to B} &= \int_0^1 \frac{\partial F(\Lambda)}{\partial \Lambda} d\Lambda \\
&= -k_B T \int_0^1 \frac{1}{Z}\frac{\partial Z}{\partial \Lambda} d\Lambda \\
&= \int_0^1 \frac{1}{Z}\sum_{\mathcal{S}} \frac{\partial E_{\mathcal{S}}(\Lambda)}{\partial \Lambda} e^{-E_{\mathcal{S}}(\Lambda)/k_B T} d\Lambda \\
&= \int_0^1 \left\langle \frac{\partial E(\Lambda)}{\partial \Lambda} \right\rangle_\Lambda d\Lambda\,, \\
&= \int_0^1 \langle E(B) - E(A) \rangle_\Lambda \, d\Lambda.
\end{aligned}$$

Thermodynamic integration: A sketch

System A

System B

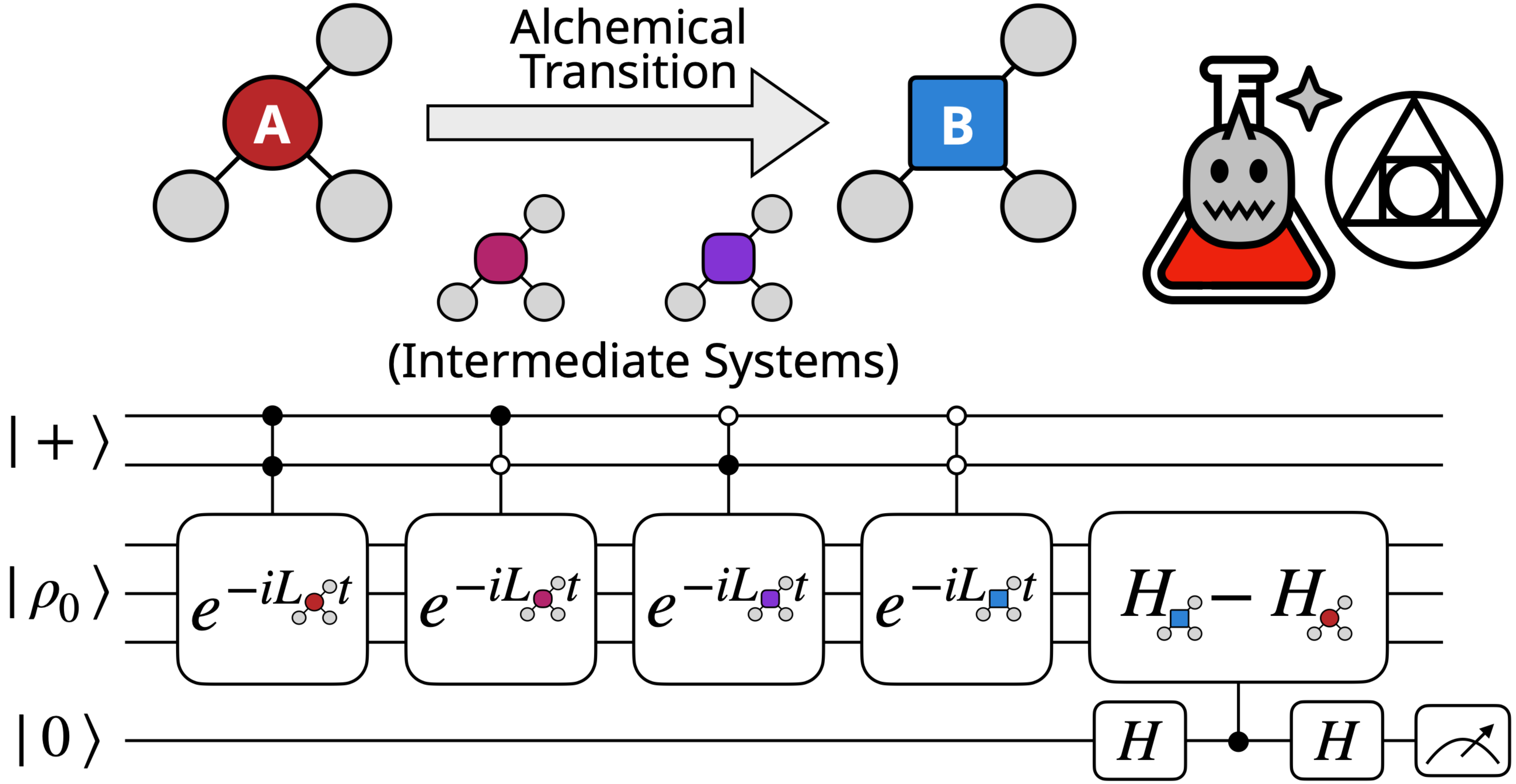
(Non-physical) intermediate state

Generate the thermal distributions under each (intermediate) system

Calculate internal energy difference for each (intermediate) system

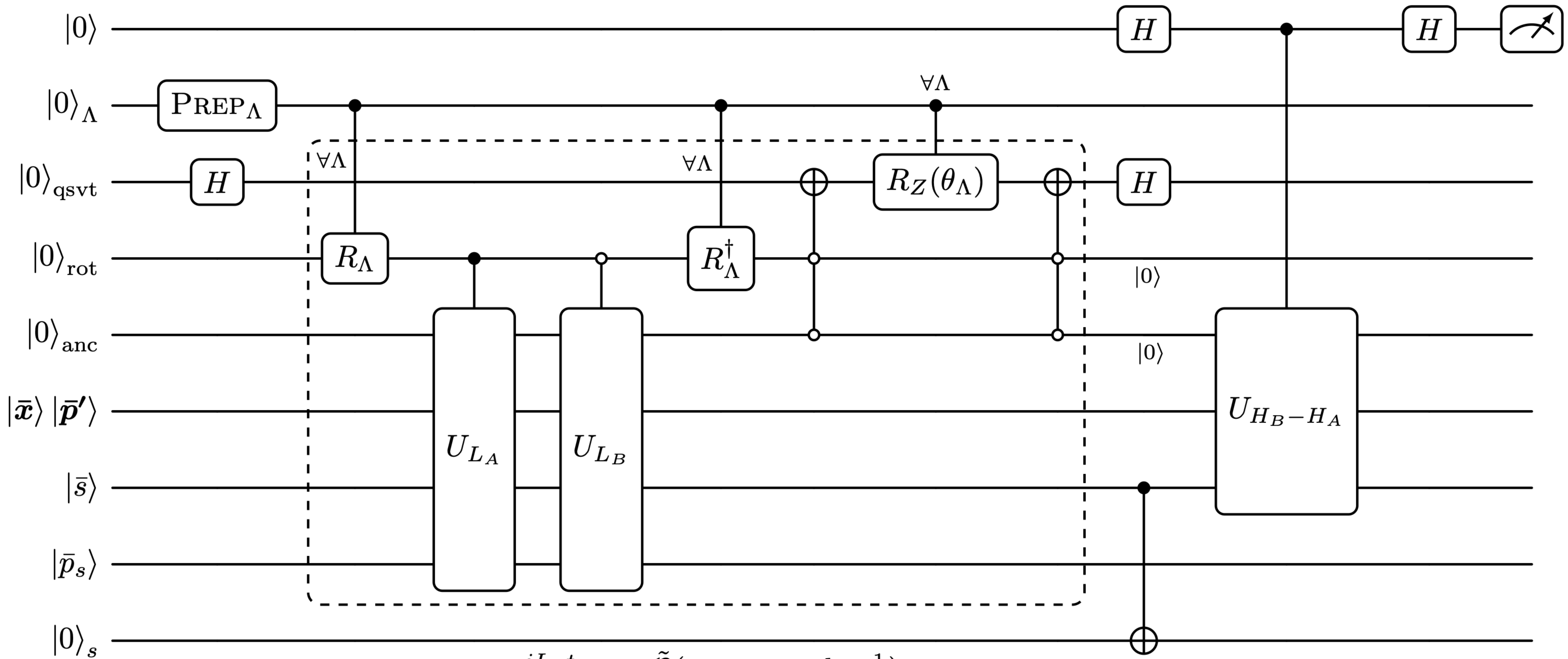
Calculating free-energy differences

Thermodynamic integration on a quantum computer



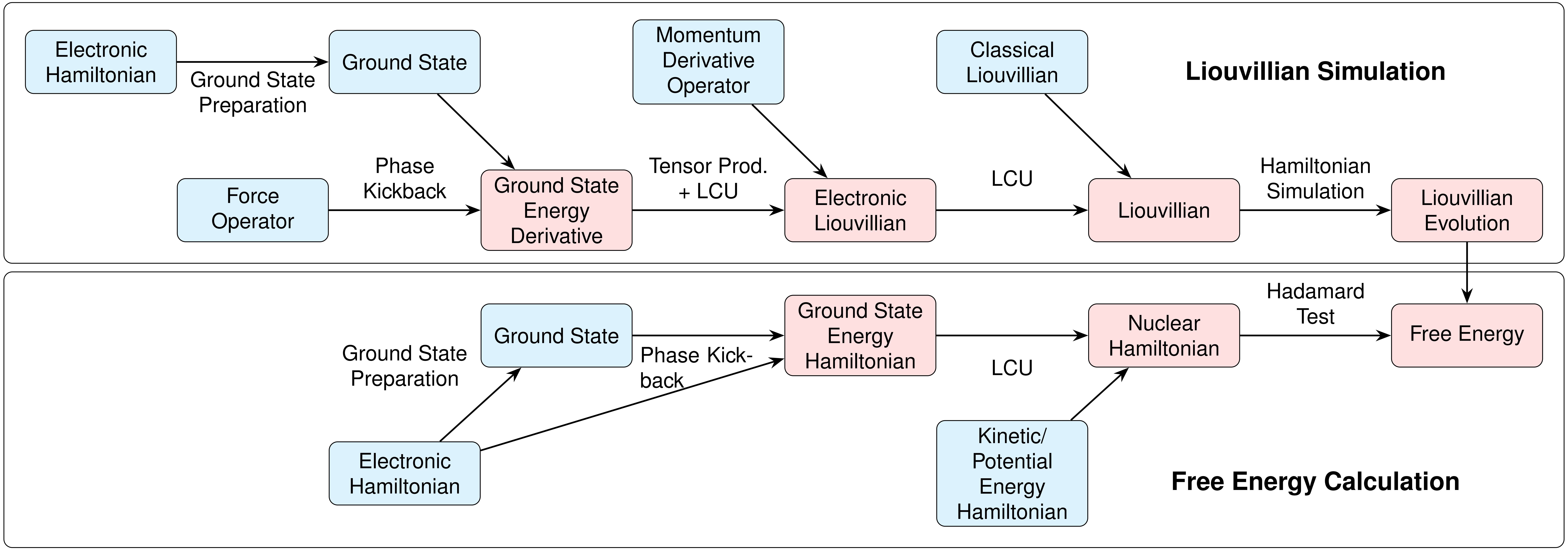
Controlled Liouvillian simulations

How the algorithm doesn't scale linear to the interpolation points



$e^{-iL_\Lambda t_{\mathrm{eq}}} : \times\tilde{\mathcal{O}}(\alpha_{\max} t_{\mathrm{eq}} + \log \frac{1}{\varepsilon})$

Algorithm breakdown



Performance comparison

With other quantum algorithms and advantages over classical

- Super-polynomial improvements in precision over iterating between MD time step and electronic structure calculations (with both classical and quantum computing methods)
- Super-polynomial improvements over previous Trotterized Liouvillian approach
- Bypass entropy estimation via relative free energy calculations (no sampling over exponential phase-space)

	MD simulation	Free energy calculation
QPE + Euler Int. [1]	$\widetilde{O}(\varepsilon^{-1})$	—
Liouville + Trott. [2]	$\widetilde{O}(\varepsilon^{-o(1)})$	$\tilde{O}(\eta^{1+o(1)}\varepsilon^{-1.5}+\varepsilon^{-2})$
Our work	$O(\log^3 \varepsilon^{-1})$	$\widetilde{O}(\varepsilon^{-1})$