# Quantum Computing for Plasma Physics: State of the Art and Potential Opportunities

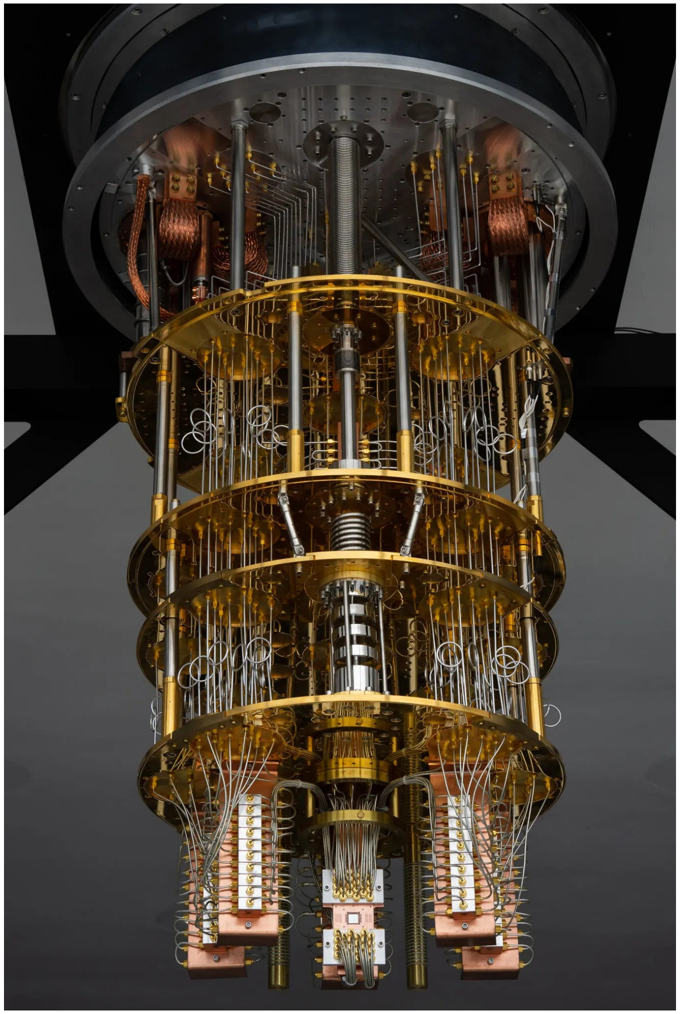
**Ilya Y. Dodin**

*Princeton Plasma Physics Laboratory*
*Princeton University*

**3rd International Fusion and Plasma Conference**
**Seoul Olympic Parktel, Seoul**

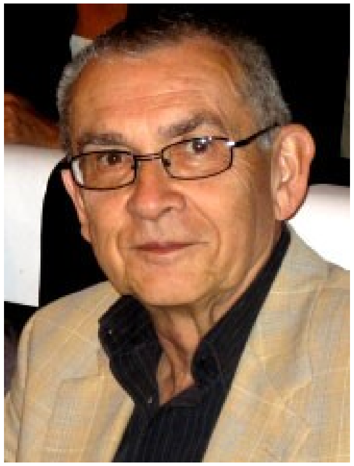
https://www.princeton.edu/∼idodin/

# Outline



- Introduction to quantum computing
  - History
  - Current specs
  - Basic concepts
- Quantum algorithms for plasma physics
  - Linear solvers
  - Nonlinear solvers
- Final thoughts & summary

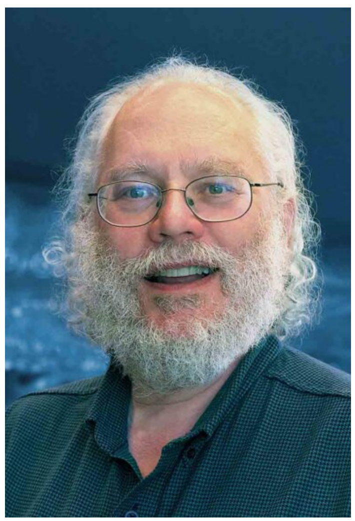
Image: a model of the interior of a quantum computer at IBM (nytimes.com)



Yury Manin

Richard Feynman

David Deutsch



Peter Shor

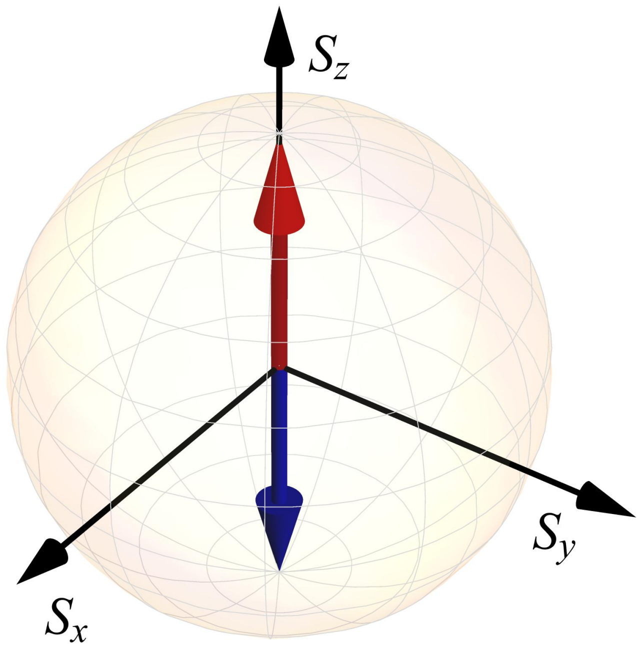
- Inception of the idea in the early 80s: Manin, Feynman
- Feynman, in "Simulating Physics with Computers" ('81):

  *"…nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem…"*

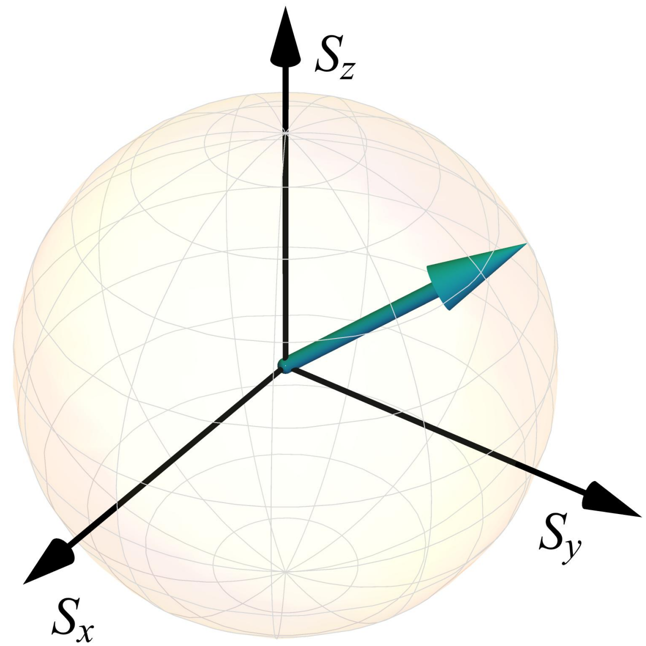
- Formalized notion of a quantum computer, question about applications beyond QM: Deutsch ('85)
- Applied algorithms: Shor's algorithm for factoring integers ('94). . .

Feynman (1982); Preskill (2021)

# Elementary memory cells: classical bit vs. qubit



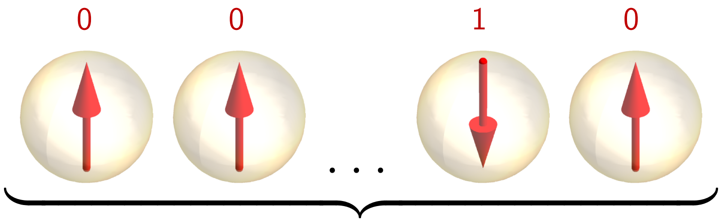

vs.




$$\begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix} \equiv |0\rangle \,, \qquad \begin{pmatrix} \mathbf{0} \\ \mathbf{1} \end{pmatrix} \equiv |1\rangle$$

- Memory cell = two-level system characterized by a state vector $\psi$ or $\boldsymbol{S_a} \sim \boldsymbol{\psi^\dagger \sigma_a \psi}$.
- A classical memory cell (bit) can be only in one of the pure states: $|0\rangle$ or $|1\rangle$. A quantum cell (qubit) can be in any state $\alpha\,|0\rangle + \beta\,|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$.
- Bits are flipped using $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. Qubits: can use any unitaries, e.g. $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$.

# Quantum parallelism




$n$ cells, "bit string" 00...10

$$N = 2^n \begin{cases} |b_1\rangle = |0\ldots00\rangle \\ |b_2\rangle = |0\ldots01\rangle \\ |b_3\rangle = |0\ldots10\rangle \\ |b_4\rangle = |0\ldots11\rangle \\ \quad\ldots \\ |b_{N-1}\rangle = |1\ldots10\rangle \\ |b_N\rangle = |1\ldots11\rangle \end{cases}$$

- A classical register with $n$ cells allows one to encode **one integer** $a \in [1, N]$, $N = 2^n$:

$$|\psi\rangle = 0\,|b_1\rangle + 0\,|b_2\rangle + \ldots + \mathbf{1}\,|b_a\rangle + \ldots 0\,|b_N\rangle$$

operations = shifts of the unit coefficient

- A quantum register with $n$ cells allows one to encode $\boldsymbol{N = 2^n}$ **complex numbers** $\psi_k$:

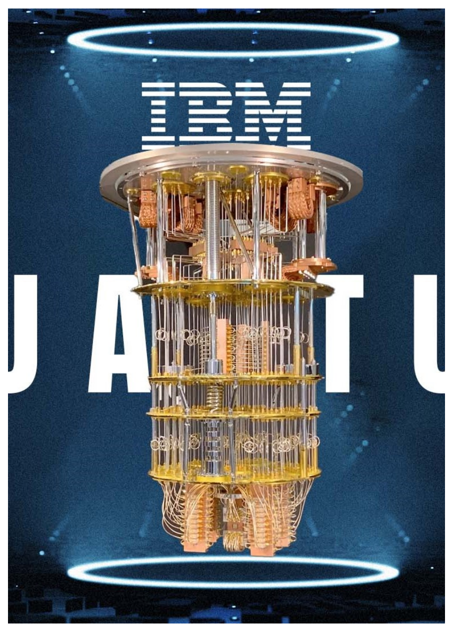
$$|\psi\rangle = \sum_{k=1}^{N} \psi_k\,|b_k\rangle\,, \quad \sum_{k=1}^{N} |\psi_k|^2 = 1$$

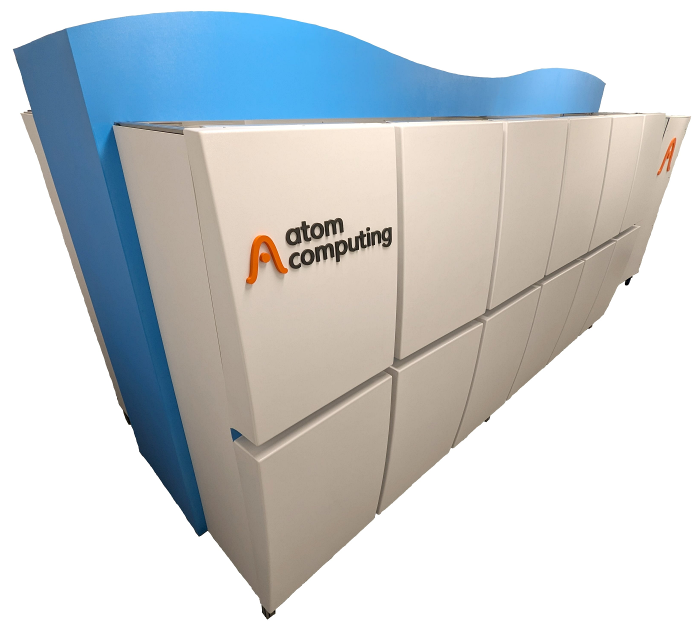
operations = any unitary transformations
(single-qubit and multi-qubit gates)

- A quantum computer can perform parallel processing of exponentially many complex numbers (e.g. field amplitudes on a grid).

# Existing hardware

- D-Wave Advantage: 5000+ qubits, but it does only quantum annealing.
- Atom's chip: 1125 qubits. IBM's Condor chip: 1121 qubits. But IBM's latest System Two uses Heron chips with only 133 qubits (5x smaller error rates).
- Typical are NISQ* computers: $< 10^3$ qubits, error rate $\sim 1\%$.


*NISQ = noisy intermediate-scale quantum

# Quantum supremacy


- In 2019, Google claimed "quantum supremacy" on a 53-qubit machine:

  circuit depth $\sim 20$
  number of gates $\sim 1.5$k
  fidelity $\sim 0.2\%$
  equiv classical simulation $\sim 10^4$ yrs

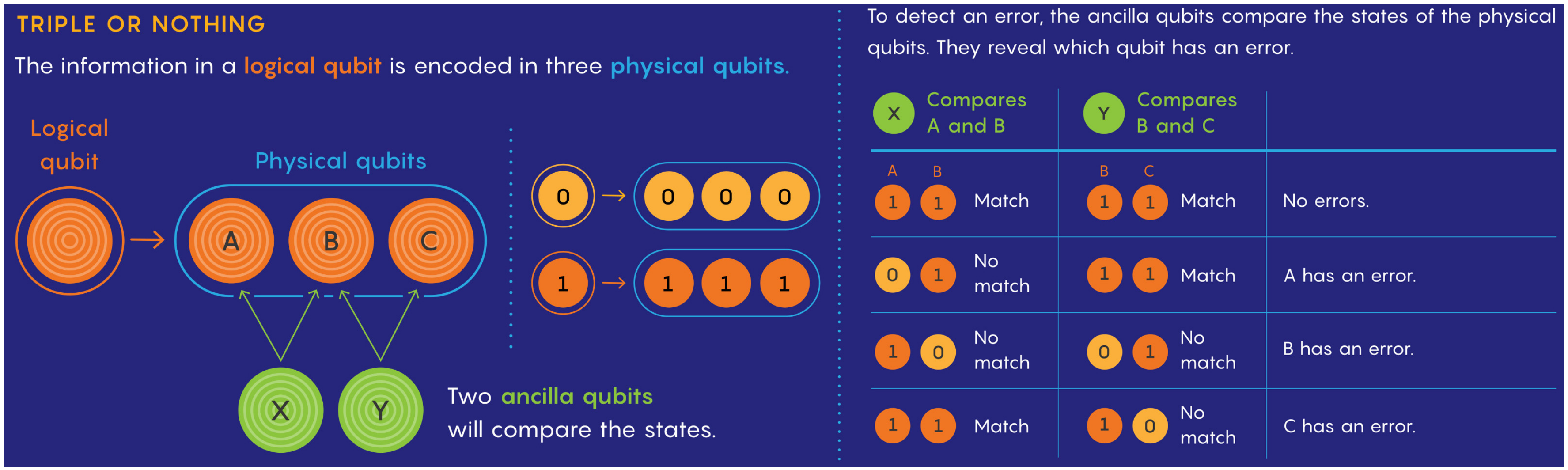
- But:
  - later debunked: classical computers can do the same within days and with much higher accuracy;
  - not a useful problem anyway.

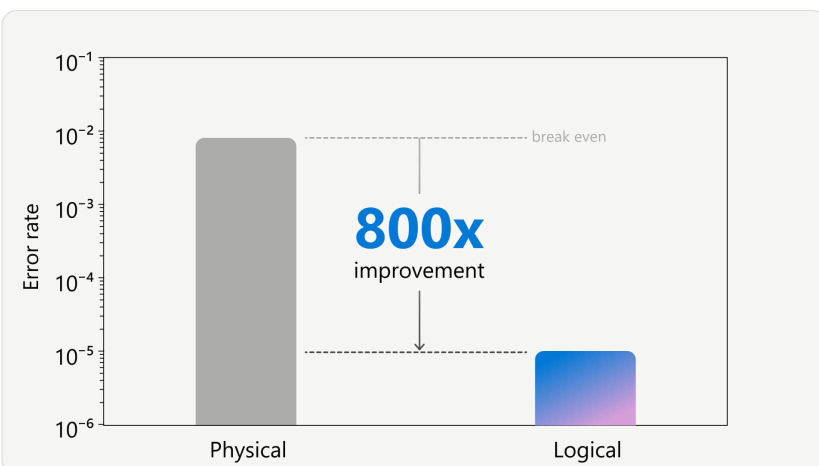
- As of now, classical computers outperform quantum computers for all real-world applications. We need more qubits, and we need qubits to be more reliable.

Arute *et al.* (2019); Pednault *et al.* (2019)

# Quantum error correction

- A set of physical qubits can operate as a fault-tolerant logical qubit:



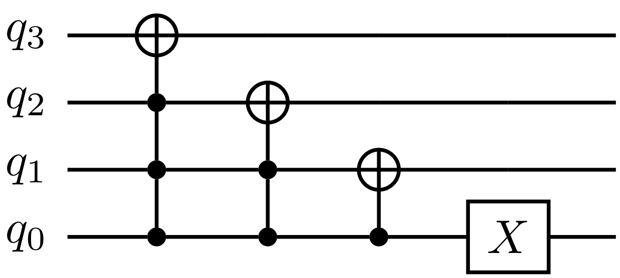



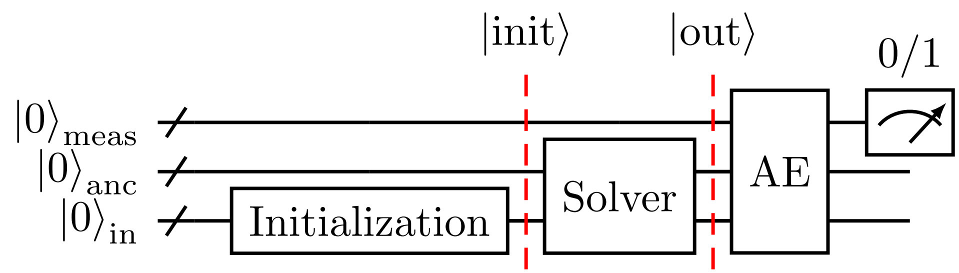

- Record by Microsoft (April 2024):
  - error rate reduced by 800x, to $10^{-5}$
  - 4 stable logical qubits made out of 30 physical (ion-trap) qubits
- Let's assume we have enough logical qubits. . .

Images taken from quantamagazine.com , blogs.microsoft.com ; see also Bluvstein *et al.* (2023)

# Quantum program is a circuit.

- Quantum computation = sequence of unitary operations on qubits. An immediate bottleneck (one of many): hardware implementation of multi-controlled gates.




- Generic circuit: initialization → solver → amplitude estimation (AE) → measurement.
  - **Initialization:** only certain states can be created efficiently – bottleneck.
  - **Solver:** involves ancilla qubits for intermediate calculations and returns

$$|\text{out}\rangle = |\mathbf{0}\rangle_{\text{anc}} \sum_k \psi_k^{\text{out}} |s_k\rangle_{\text{in}} + |\neq \mathbf{0}\rangle_{\text{anc}} |\ldots\rangle$$

  - **Measurement:** run $N_{\text{run}}$ times, $|\psi_k^{\text{out}}|^2 = N_k/N_{\text{run}} + \mathcal{O}(N_{\text{run}}^{-1/2})$. With AE, a measurement returns $|\psi_k^{\text{out}}|^2$ with probability close to one and error $\mathcal{O}(N_{\text{run}}^{-1})$.

# Part II: Quantum algorithms for plasma physics

| | initial-value problems | boundary-value problems* |
|---|---|---|
| **linear** | $\partial_t \psi = -i\hat{H}\psi$ | $A\psi = b$ |
| **nonlinear** | $\partial_t u = g(u)$ | $F(u) = 0$ |

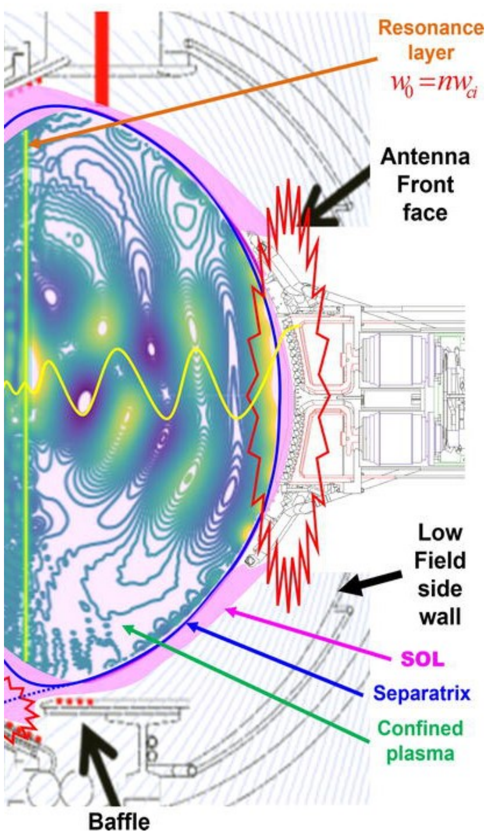
* These also include non-differential equations and optimization problems.

Not discussing eigenvalue problems here, but see, e.g. Parker & Joseph (2020).

# Quantum Hamiltonian simulations (QHS)

- Quantum computers are naturally fit for *quantum Hamiltonian simulations*. For a Hermitian $\hat{H}$, a quantum circuit can directly implement unitary $\hat{U} = \exp(-i\hat{H}t)$.

$$\boxed{\partial_t \psi = -i\hat{H}\psi, \qquad \psi(t) = \exp(-i\hat{H}t)\psi_0}$$




- Example: cold-plasma waves, $\psi \sim (g_1 \boldsymbol{v}_1, g_2 \boldsymbol{v}_2, \dots, \tilde{\boldsymbol{E}}, \tilde{\boldsymbol{B}})^\intercal$

$$\partial_t \tilde{\boldsymbol{v}}_s = e_s \tilde{\boldsymbol{E}}/m_s + \tilde{\boldsymbol{v}}_s \times \boldsymbol{\Omega}_s$$
$$\partial_t \tilde{\boldsymbol{E}} = c\nabla \times \tilde{\boldsymbol{B}} - 4\pi \textstyle\sum_s e_s n_{0s} \tilde{\boldsymbol{v}}_s$$
$$\partial_t \tilde{\boldsymbol{B}} = -c\nabla \times \tilde{\boldsymbol{E}}$$

$$\hat{H} = \begin{pmatrix} -\boldsymbol{\alpha}\cdot\boldsymbol{\Omega}_1(\boldsymbol{x}) & 0 & \dots & 0 & i\omega_{p1}(\boldsymbol{x}) & 0 \\ 0 & -\boldsymbol{\alpha}\cdot\boldsymbol{\Omega}_2(\boldsymbol{x}) & \dots & 0 & i\omega_{p2}(\boldsymbol{x}) & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & -\boldsymbol{\alpha}\cdot\boldsymbol{\Omega}_q(\boldsymbol{x}) & i\omega_{pq}(\boldsymbol{x}) & 0 \\ -i\omega_{p1}(\boldsymbol{x}) & -i\omega_{p2}(\boldsymbol{x}) & \dots & -i\omega_{pq}(\boldsymbol{x}) & 0 & c\boldsymbol{\alpha}\cdot\nabla \\ 0 & 0 & \dots & 0 & -c\boldsymbol{\alpha}\cdot\nabla & 0 \end{pmatrix}$$

$$\alpha_x = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -i \\ 0 & i & 0 \end{pmatrix}, \quad \alpha_y = \begin{pmatrix} 0 & 0 & i \\ 0 & 0 & 0 \\ -i & 0 & 0 \end{pmatrix}, \quad \alpha_z = \begin{pmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

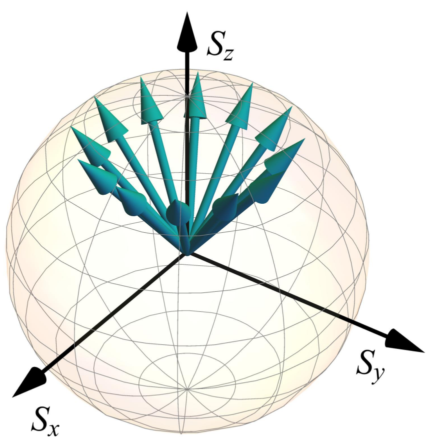
Dodin and Startsev (2021); image taken from intechopen.com

# There are various ways to construct $\hat{U} = \exp(-i\hat{H}t)$.

- **Suzuki–Trotter expansion**: (i) decompose $\hat{H} = \sum_j \hat{H}_j$, where $\hat{H}_j$ are rotation gates; (ii) divide $t$ into $m \gg 1$ intervals $\Delta t = t/m$, so $[\hat{H}_i\,\Delta t, \hat{H}_j\,\Delta t] = \mathcal{O}(m^{-2})$.

$$\hat{U} = \left[\exp\left(-\textstyle\sum_j \hat{H}_j\,\Delta t\right)\right]^m \approx \Big[\textstyle\prod_j \underbrace{\exp(-i\hat{H}_j\,\Delta t)}_{\text{implementable w/elementary gates}}\Big]^m$$

- Quantum Signal Processing (QSP)/Quantum Singular Value Transformation (QSVT):
  - Allow to calculate **polynomials of given matrices** for many applications.
  - Based on the idea of rotating unitaries. For example, by properly choosing $r$ angles $\phi_k$, one can construct an $r$th-order polynomial $P_r(a)$ of a scalar $a$:




$$U(a) = \begin{pmatrix} a & \sqrt{1-a^2} \\ \sqrt{1-a^2} & -a \end{pmatrix}, \quad R(\phi) = \begin{pmatrix} \mathrm{e}^{i\phi} & 0 \\ 0 & \mathrm{e}^{-i\phi} \end{pmatrix}$$

$$R(\phi_0)\,\underbrace{U(a)\,R(\phi_2)\ldots R(\phi_{r-1})\,U(a)}_{r\text{ powers of }U}\,R(\phi_r) = \begin{pmatrix} P_r(a) & * \\ * & * \end{pmatrix}$$

Suzuki (1990); Gilyén *et al.* (2018)

## Block encoding and application to Hamiltonian simulations

- To obtain a polynomial of a matrix, encode $A$ as a block of a unitary $U_A$, then rotate:

$$U_A = \begin{pmatrix} A & * \\ * & * \end{pmatrix}, \qquad R(\phi) = \begin{pmatrix} e^{i\phi}\,1 & 0 \\ 0 & e^{-i\phi}\,1 \end{pmatrix}, \qquad A \to \frac{A/||A||}{\text{sparsity}}$$

$$\underbrace{R(\phi_0)\,U_A\,R(\phi_2)\,U_A^\dagger \ldots U_A^\dagger R(\phi_{r-1})\,U_A\,R(\phi_r)}_{\text{same } \phi_k \text{ as for scalars; finding them for large } r \text{ is a bottleneck}} = \begin{pmatrix} P_r(A) & * \\ * & * \end{pmatrix}$$

- **QSP for Hamiltonian simulations**: express $e^{-i\hat{H}t}$ through Chebyshev polynomials $T_k(\hat{H})$ defined via $T_k(\cos\theta) = \cos(k\theta)$; calculate $T_k(\hat{H})$ using the appropriate $\phi_k$.

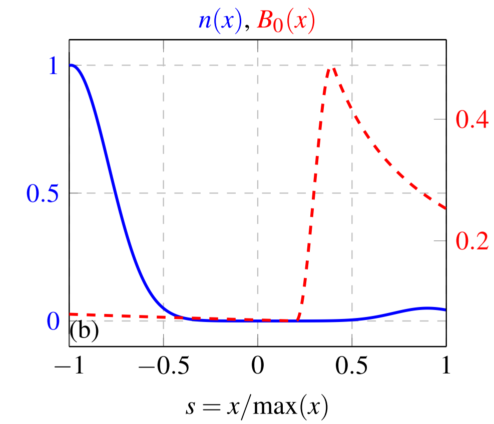
$$e^{-i\hat{H}t} = J_0(t) + 2\sum_{\text{even } k>0}^{\infty} (-1)^{k/2} J_k(t)\boldsymbol{T_k(\hat{H})} + 2i\sum_{\text{odd } k>0}^{\infty} (-1)^{(k-1)/2} J_k(t)\boldsymbol{T_k(\hat{H})}$$

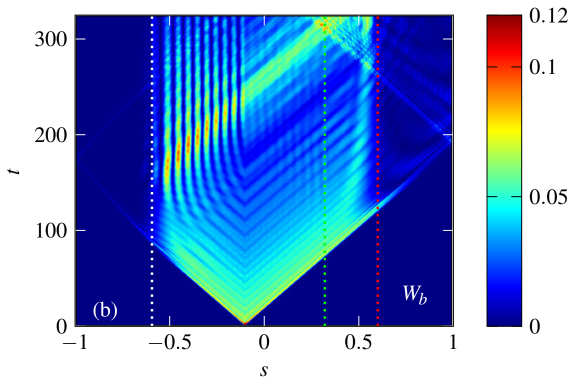
Near-optimal dependence of the number of calls to $\hat{H}$ on time and error:

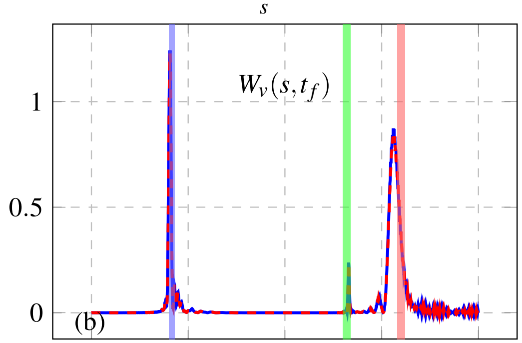
$$e^{-i\hat{H}t} = \textstyle\sum_{k=0}^{q}(\ldots) + \underbrace{\mathcal{O}((et/q)^q)}_{\epsilon} \quad \Rightarrow \quad q = \mathcal{O}(t + \log(1/\epsilon))$$

Childs *et al.* (2018); Low and Chuang (2019); Martyn *et al.* (2021); Dong *et al.* (2021)

# Example: exponential speedup of full-wave simulations for cold plasma



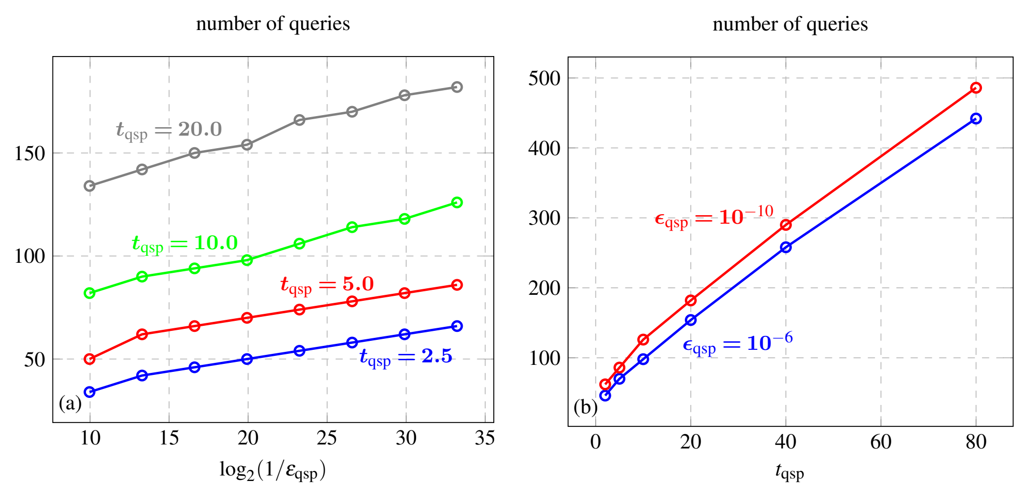




$$\mathcal{O}\Big( \underbrace{\text{poly}(\log_2 N_x)}_{\text{oracle}} \ \underbrace{(t + \log_2(1/\delta))}_{\text{QSP circuit}} \ \underbrace{\delta^{-1}}_{\text{measurements}} \Big)$$

- X wave propagation in 1-D electron plasma; antenna = oscillator. Efficient initialization and measurements.

$$|\psi|^2 = n_0 m_e \tilde{v}_e^2/2 + (\tilde{E}^2 + \tilde{B}^2)/8\pi$$

$$\langle\psi|\text{window operator}|\psi\rangle = \text{energy}$$



# Quantum Hamiltonian simulations beyond the cold-plasma approximation

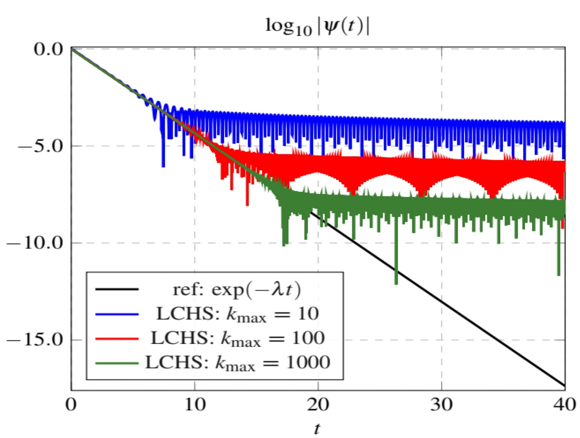
- QHS are applicable to linear kinetic waves in *homogeneous isotropic plasma*. The general linearized Vlasov–Maxwell dynamics is not always Hermitian (instabilities).

$$
\begin{aligned}
i\partial_t \boldsymbol{E} &= c(\boldsymbol{\alpha}\cdot\nabla)\boldsymbol{B} + \textstyle\sum_{s,\boldsymbol{p}} R_s \boldsymbol{v}_s g_s \\
i\partial_t \boldsymbol{B} &= -c(\boldsymbol{\alpha}\cdot\nabla)\boldsymbol{E} \\
i\partial_t g_s &= \hat{h}_s g_s + R_s \boldsymbol{E}\cdot\boldsymbol{v}_s, \quad g_s \propto \tilde{f}_s
\end{aligned}
$$

$$
i\partial_t \underbrace{\begin{pmatrix} \tilde{\boldsymbol{E}} \\ \tilde{\boldsymbol{B}} \\ \boldsymbol{g} \end{pmatrix}}_{\psi} = \underbrace{\begin{pmatrix} 0 & c(\boldsymbol{\alpha}\cdot\nabla) & R\boldsymbol{v} \\ -c(\boldsymbol{\alpha}\cdot\nabla) & 0 & 0 \\ R\boldsymbol{v} & 0 & \hat{h} \end{pmatrix}}_{\hat{H}=\hat{H}^\dagger \text{ when } R_s \propto (f'_{0s})^{1/2} \text{ is real}} \begin{pmatrix} \tilde{\boldsymbol{E}} \\ \tilde{\boldsymbol{B}} \\ \boldsymbol{g} \end{pmatrix}
$$

- QHS w/non-Hermitian $\hat{H}$: Linear Combination of Hamiltonian Simulations (LCHS)

$$
\mathrm{e}^{-i\hat{H}t} = \frac{1}{\pi}\int \frac{1}{1+k^2}\underbrace{\exp\left(-i\hat{H}_{\mathrm{H}}t + ik\hat{H}_{\mathrm{A}}t\right)}_{\text{unitary}} \mathrm{d}k \approx \sum_k (\ldots)_k \,\Delta k
$$




However, truncation and discretization cause errors, so one needs *many* terms in the sum.

- Also, in practice, linear-wave problems are usually boundary-value, so let's consider those. . .

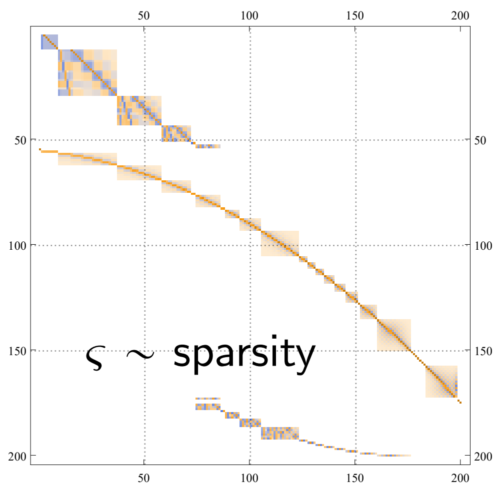
Engel *et al.* (2019); Dodin and Startsev (2021); An *et al.* (2023); Novikau (unpublished)

# Linear boundary-value problems using QSVT

- A typical RF problem: $\partial_t = -i\omega$, so $\hat{\varepsilon}$ is a *spatial* operator with $\omega$ as a parameter.

$$c^2\nabla\times\nabla\times\tilde{\boldsymbol{E}} - \omega^2\hat{\varepsilon}(\omega)\tilde{\boldsymbol{E}} = \underbrace{4\pi i\omega J_{\text{ext}}}_{\text{antenna}} \quad\Rightarrow\quad A\psi = b, \quad \psi = A^{-1}b$$

- **QHS-based:** Harrow–Hassidim–Lloyd (HHL) algorithm and its later variations
  - Exponential speedup for sparse and well-conditioned $A$, $\kappa \equiv \lambda_{\max}/\lambda_{\min} \sim 1$.
  - Pre-exponential factors can be prohibitively large: quantum advantage would take $N \sim 10^8$, 340 qubits, depth $\sim 10^{25-29}$, runtime $10^{8-12}$ yrs (2-D scattering).

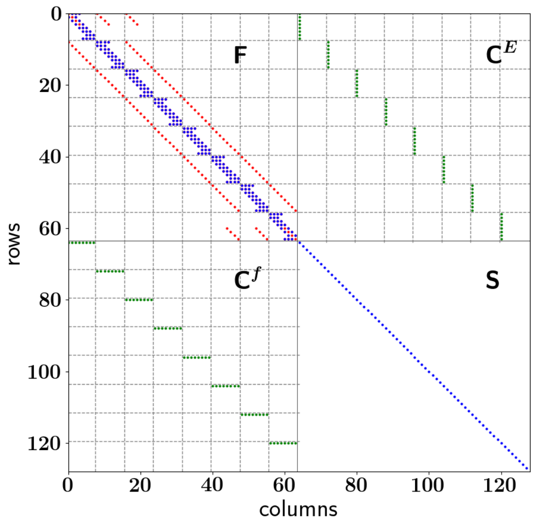



- **QSVT-based:** polynomial approximation to $A^{-1}$
  - FEM matrices, $N = N_x^D$: $\kappa = \mathcal{O}(N^{2/D}) = \mathcal{O}(N_x^2)$

    $$\mathcal{O}\left(\kappa^2\varsigma\ln(N)\ln(\kappa/\epsilon)\right) \to \mathcal{O}\left(N_x^4\varsigma\ln(N_x)\ln(N_x^2/\epsilon)\right)$$
  - best classical methods have $\mathcal{O}\left(\varsigma N_x^{D+2}\ln(1/\epsilon)\right)$, so quantum advantage is possible at $D \geqslant 3$

Harrow *et al.* (2009); Clader *et al.* (2013); Scherer *et al.* (2017); Novikau *et al.* (2023)

# Example: boundary-value problem for kinetic waves

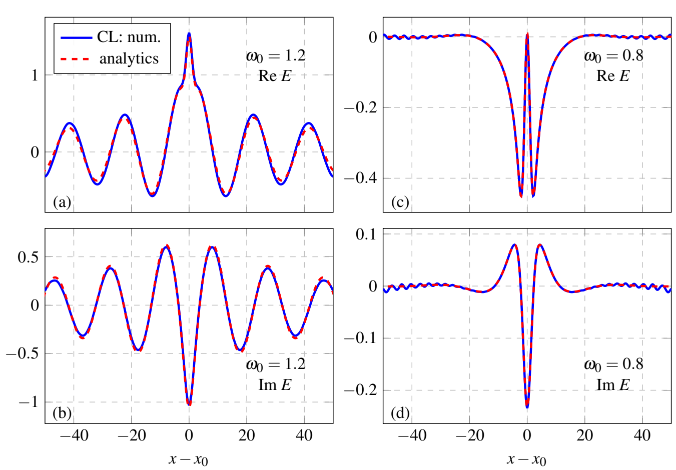


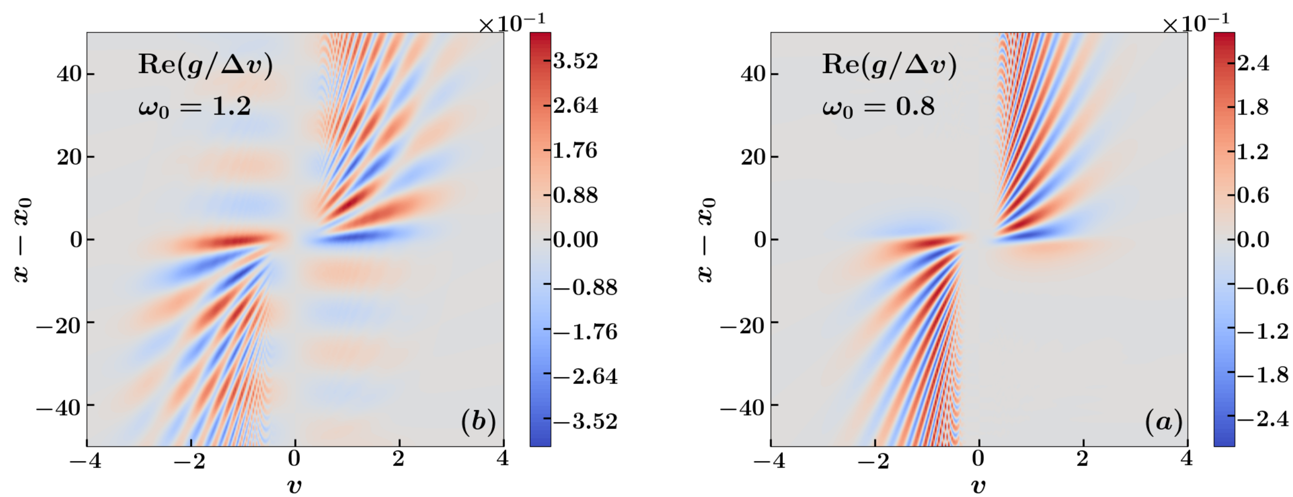

- Linearized Vlasov–Ampere system: 1-D electron plasma, $f(t,x,v) = F(x,v) + g(t,x,v)$, $\partial_t = -i\omega$:

$$-i\omega g + v\partial_x g - E\partial_v F = 0$$

$$-i\omega E - \int vg\,\mathrm{d}v = -j_{\mathrm{ext}}$$

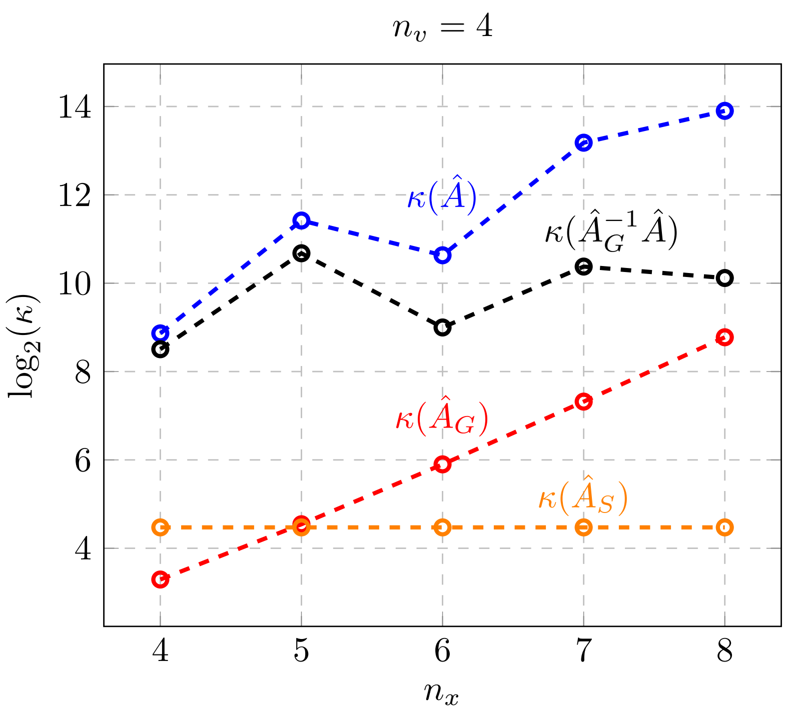
- Solve $A\psi = b$ for $\psi = (g, E)^\intercal$ and $b \sim (0, j_{\mathrm{ext}})^\intercal$



# Preconditioning

- Since the QSVT scaling, $\mathcal{O}\left(\kappa^2 \varsigma \ln(N_x) \ln(\kappa/\epsilon)\right)$, involves strong dependence on the condition number $\kappa \sim N_x^2$, preconditioning is likely necessary at large $N_x$.




  - A matrix $P$ can serve as a preconditioner for $A$ if $PA$ has a condition number $\kappa(PA) \ll \kappa(A)$.

$$A\psi = b \quad \Rightarrow \quad PA\psi = Pb$$

$$\boxed{\psi = (PA)^{-1} Pb}$$

  - Since $\kappa(PA) \ll \kappa(A)$, the matrix $(PA)^{-1}$ is easier to calculate than $A^{-1}$.

- For example, an approximation to $A_G^{-1}$ can serve as a preconditioner for our $A$.

$$A = A_G + A_S, \qquad A_G = \begin{pmatrix} F & 0 \\ 0 & S \end{pmatrix} - \alpha I, \qquad A_S = \begin{pmatrix} 0 & C^E \\ C^f & 0 \end{pmatrix} + \alpha I$$

## The challenge of nonlinear simulations

- **No-cloning theorem:** it is impossible to create a copy of an unknown quantum state. So one can't, for example, calculate $\psi^2$ like this:

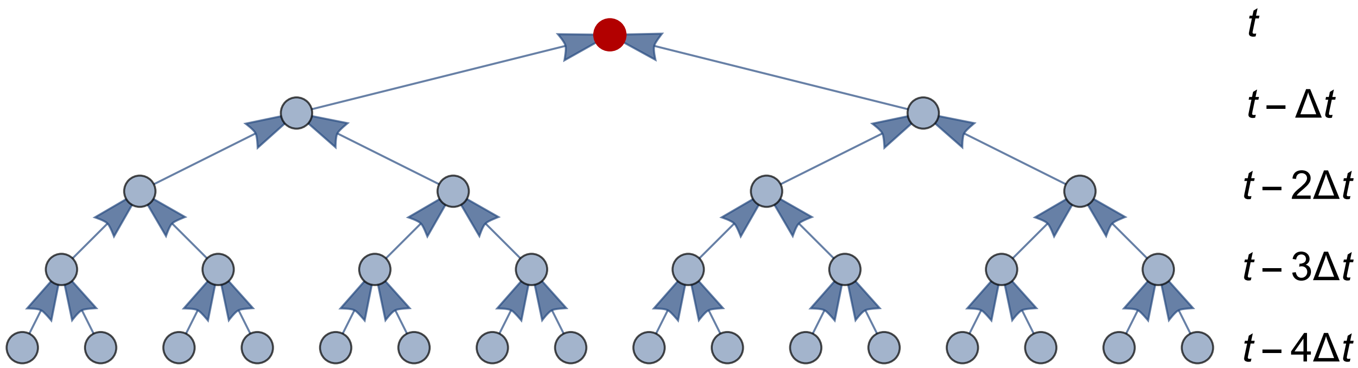
$$|\psi\rangle \not\mapsto \sum_{jk} A_{ijk}\psi_j\psi_k \, |i\rangle$$

- One *can* make $\psi^2$ out of two copies of $\psi$ prepared independently (w/post-selection):

$$|\psi\rangle\,|\psi\rangle\,|0\rangle_a \longmapsto |\psi\rangle\,|\psi\rangle\,|0\rangle_a + \sum_{jk} A_{ijk}\psi_j\psi_k \, |i\rangle\,|0\rangle\,|1\rangle_a$$

$$\langle 1|_a\,(\ldots) = \sum_{jk} A_{ijk}\psi_j\psi_k \, |i\rangle\,|0\rangle$$

- But then, at integrating, say, $\partial_t\psi_i = \sum_{jk} A_{ijk}\psi_j\psi_k$, $r$ steps require $\mathbf{2^r}$ copies of $\psi$:




Leyton and Osborne (2008)

# Linear embedding

- **Theorem** (kind of): any nonlinear system can be made *exactly* linear using a sufficiently large phase-space extension.

- For example, Hamiltonian systems can be *quantized*.

  - **Example 1:** single-particle motion in prescribed fields

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \frac{p}{m}, \quad \frac{\mathrm{d}p}{\mathrm{d}t} = -\frac{\partial V}{\partial x} \quad \Rightarrow \quad i\frac{\partial \psi}{\partial t} = \left(-\frac{\hbar^2\nabla^2}{2m} + V\right)\psi$$

  - **Example 2:** three-wave interaction. Use $A_i \to \hat{A}_i$ and $A_i^* \to \hat{A}_i^\dagger$. Due to $n_1 + n_2 =$ const and $n_1 + n_3 =$ const, the accessible Fock space is finite-D.

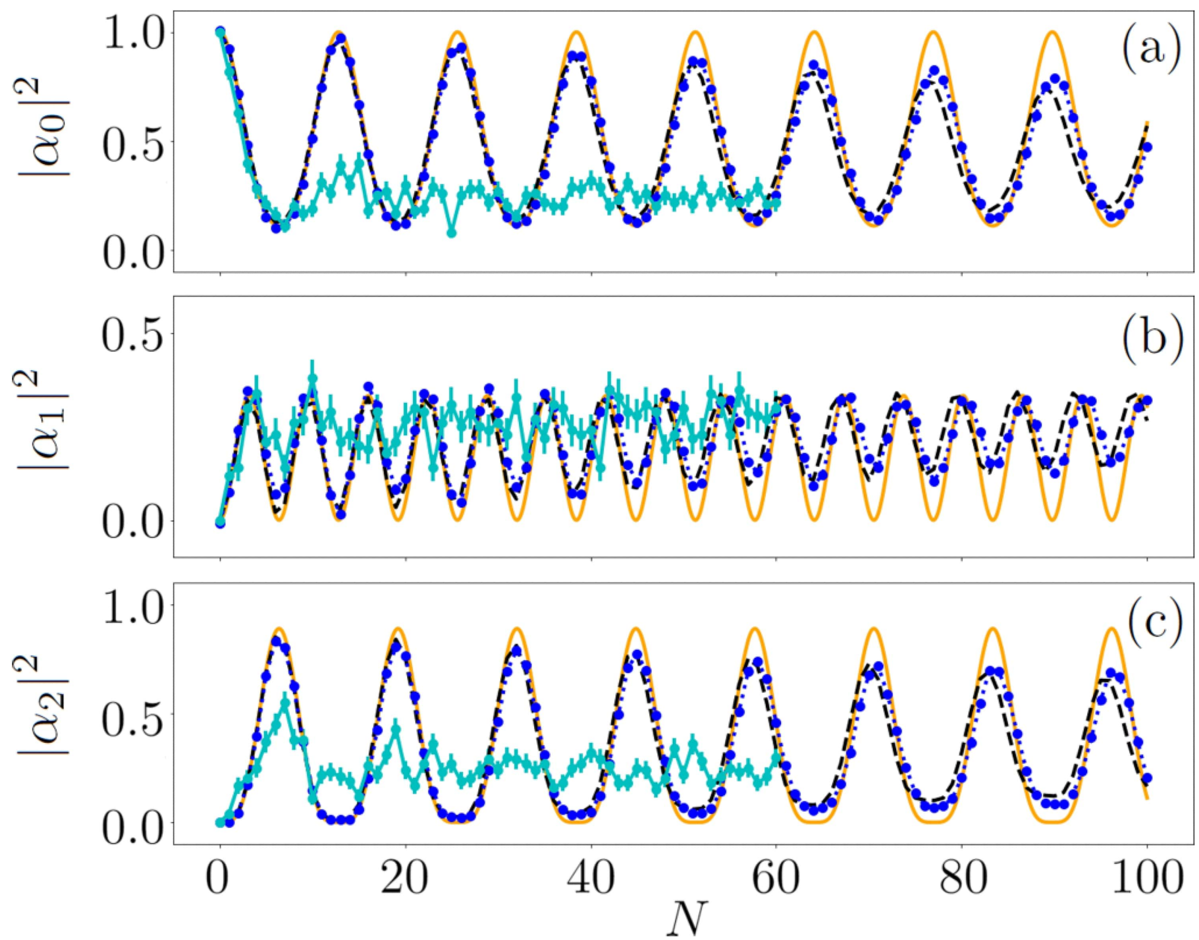
$$\dot{A}_1 = gA_2A_3, \qquad \dot{A}_2 = -g^*A_1A_3^*, \qquad \dot{A}_3 = -g^*A_1A_2^*$$

$$h = \begin{pmatrix} 0 & g\sqrt{2(s-1)} & 0 \\ g^*\sqrt{2(s-1)} & 0 & g\sqrt{2s} \\ 0 & g^*\sqrt{2s} & 0 \end{pmatrix}$$

Dodin and Startsev (2021); Engel *et al.* (2021); Shi *et al.* (2021)

# Quantum simulations of three-wave interaction: an experiment

- LLNL experiment on Aspen-4-2Q-A of Rigetti Computing, 2 qubits. $\hat{U} = \mathrm{e}^{-i\hat{H}t}$ is converted to $\sim 20$ native gates. $N$ is the number of times $\hat{U}$ is applied.




Shi *et al.* (2021)

# Ad hoc linear embeddings are generally unstable.

- **Example:** Carleman embedding (1932)

$$\dot{u} = g(u) = \underbrace{g(u_0)}_{a_0} + \sum_{k=1}^{\infty} \underbrace{\frac{g^{(k)}(u_0)}{k!}}_{a_k} \underbrace{u^k}_{y_k}$$

$$\dot{y}_1 = a_0 + a_1 y_1 + a_2 y_2 + \dots$$

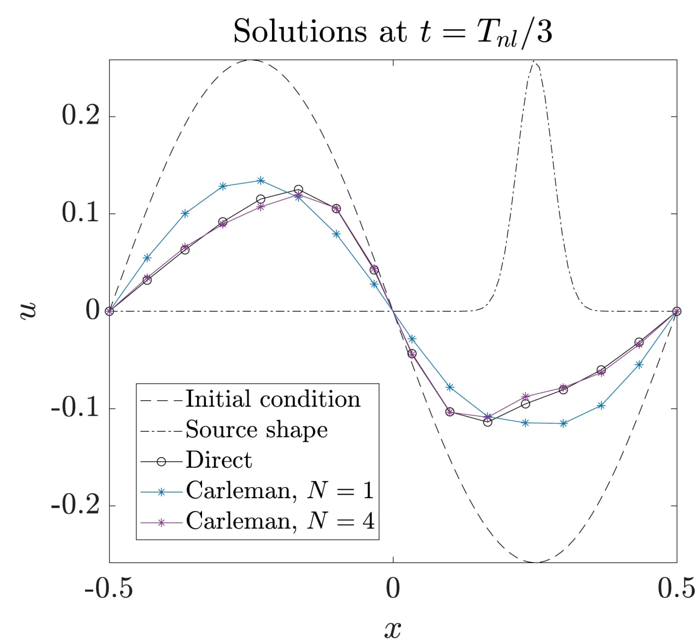
$$\begin{aligned}\dot{y}_2 &= 2u\dot{u} \\ &= 2u(a_0 + a_1 u + a_2 u^2 + \dots) \\ &= 2a_0 y_1 + 2a_1 y_2 + 2a_2 y_3 + \dots\end{aligned}$$

$$\dot{y}_3 = 3a_0 y_2 + 3a_1 y_3 + 3a_2 y_4 + \dots$$

$$\boxed{\dot{\boldsymbol{y}} = \boldsymbol{A}\boldsymbol{y} + \boldsymbol{b}, \quad \boldsymbol{y} = (y_1, y_2, \dots)^{\mathsf{T}}}$$

  - Works *only* for stable dynamics. Figure: 1-D driven Burgers' eqn at small enough Re.

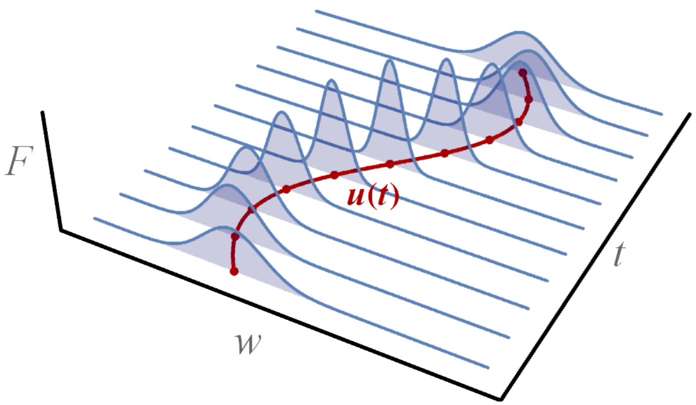
Torsten Carleman



# Koopman–von Neumann embedding: physics-based and stable

- Instead of $u$, consider the probability distribution $F(t,w) = \delta(w - u(t))$:

$$\dot{u} = g(u) \quad \Rightarrow \quad \underbrace{\partial_t F + \partial_w[g(t,w)\,F] = 0}_{\text{Liouville equation}}$$




- $\psi = \sqrt{F}$ satisfies $i\partial_t \psi = \hat{H}\psi$ with $\hat{H} = \hat{H}^\dagger$:

$$\hat{H} = {}^1\!/\!_2(g\hat{\rho} + \hat{\rho} g), \qquad \hat{\rho} = -i\partial_w$$

- Variational representation $\delta\,\langle \psi | i\partial_t - \hat{H} | \psi \rangle = 0$ → **structure-preserving truncations possible**.

- A finite-dimensional representation is obtained by substituting a truncated expansion $|\psi\rangle = \sum_{i=1}^{r} \psi_i \,|e_i\rangle$ in a suitable basis $|e_i\rangle$ into the variational principle:

$$i\partial_t \begin{pmatrix} \psi_1 \\ \dots \\ \psi_r \end{pmatrix} = \hat{\mathsf{H}} \begin{pmatrix} \psi_1 \\ \dots \\ \psi_r \end{pmatrix}, \qquad \hat{\mathsf{H}}_{jk} = \langle e_j | \hat{H} | e_k \rangle = \hat{\mathsf{H}}^*_{kj}$$

- Maybe $\dim \psi$ does not have to be very large if one chooses the right basis?..

# Koopman–von Neumann embedding for the Vlasov–Poisson system

- Vlasov equation does not feature individual particles, but one can emulate Vlasov dynamics using $N$ macroparticles. The $N$-particle distribution equation is **linear**:

$$\boxed{\frac{\partial \Psi}{\partial t} = -\sum_{i=1}^{N} v_i \frac{\partial \Psi}{\partial x_i} + \sum_{i=1}^{N} \left( \frac{\partial V_{\text{ext}}}{\partial x_i} + \sum_{j=1, j\neq i}^{N} \frac{\partial V_{ij}}{\partial x_i} \right) \frac{\partial \Psi}{\partial p_i} \equiv -i\hat{\mathcal{H}}\Psi}$$

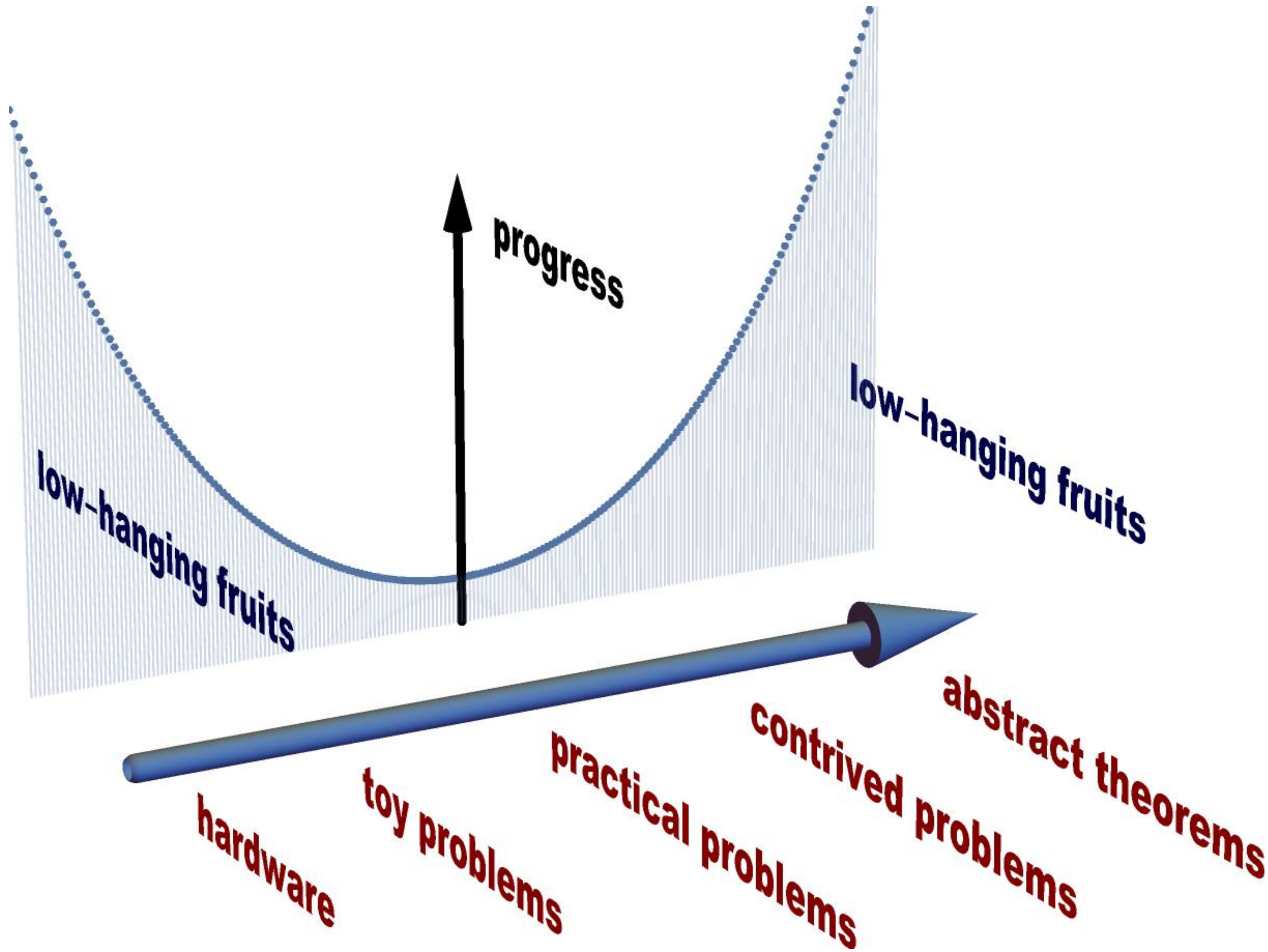
Here, $V_{\text{ext}}$ is the external potential and $V_{ij} = V(|x_i - x_j|)$ is the interaction (Coulomb) potential, which is a fixed known function.

- No need to simulate macroparticles as highly localized objects. Use expansion in global modes (e.g. Gauss–Hermite) → **finite-D conservative linear system**:

$$i\partial_t \begin{pmatrix} \psi_1 \\ \dots \\ \psi_r \end{pmatrix} = \hat{\mathsf{H}} \begin{pmatrix} \psi_1 \\ \dots \\ \psi_r \end{pmatrix}, \qquad \hat{\mathsf{H}}_{jk} = \langle e_j | \hat{\mathcal{H}} | e_k \rangle = \hat{\mathsf{H}}^*_{kj}$$

- Can this provide quantum advantage in practice? This remains to be seen.




progress
low-hanging fruits
low-hanging fruits
hardware
toy problems
practical problems
contrived problems
abstract theorems

## Summary (and the obligatory comic strip)


- The hype aside, quantum computing remains a **legitimate physics problem**.
- Possibly promising directions of research for plasma applications:
  - algorithms: **physics-based linear embeddings, hybrid computing**;
  - circuit engineering: automation of circuit development (ML/AI);
  - hardware: specialized gates for classical problems, e.g. multi-controlled gates.
- Little is done in the area of **practical algorithms** – may be a land of opportunity.
- Will it ever work? We'll never know unless we try hard, *and we have not yet*.

## References


[An *et al.* (2023)]
D. An, J.-P. Liu, and L. Lin, *Linear combination of Hamiltonian simulation for nonunitary dynamics with optimal state preparation cost*, Phy. Rev. Lett. **131**, 150603 (2023).

[Andrade and Rauh (1981)]
R. F. S. Andrade and A. Rauh, *The Lorenz model and the method of Carleman embedding*, Phys. Lett. A **82**, 276 (1981).

[Arute *et al.* (2019)]
F. Arute et al., *Quantum supremacy using a programmable superconducting processor*, Nature **574**, 505 (2019).

[Bluvstein *et al.* (2023)]
D. Bluvstein et al., *Logical quantum processor based on reconfigurable atom arrays*, Nature **626**, 58 (2023).

[Childs *et al.* (2018)]
A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, *Toward the first quantum simulation with quantum speedup*, Proceedings of the National Academy of Sciences **115**, 9456 (2018).

[Clader *et al.* (2013)]
B. D. Clader, B. C. Jacobs, and C. R. Sprouse, *Preconditioned quantum linear system algorithm*, Phys. Rev. Lett. **110**, 250504 (2013).

[Dodin and Startsev (2021)]
I. Y. Dodin and E. A. Startsev, *On applications of quantum computing to plasma simulations*, Phys. Plasmas **28**, 092101 (2021).

[Dong *et al.* (2021)]
Y. Dong, X. Meng, K. B. Whaley, and L. Lin, *Efficient phase-factor evaluation in quantum signal processing*, Phys. Rev. A **103**, 042419 (2021).

[Engel *et al.* (2019)]
A. Engel, G. Smith, and S. E. Parker, *Quantum algorithm for the Vlasov equation*, Phys. Rev. A **100**, 062315 (2019).

[Engel *et al.* (2021)]
A. Engel, G. Smith, and S. E. Parker, *Linear embedding of nonlinear dynamical systems and prospects for efficient quantum algorithms*, Phys. Plasmas **28**, 062305 (2021).

[Feynman (1982)]
R. P. Feynman, *Simulating physics with computers*, Int. J. Theor. Phys. **21**, 467 (1982).

[Gilyén *et al.* (2018)]
A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, *Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics*, arXiv:1806.01838 (2018).

[Harrow *et al.* (2009)]
A. W. Harrow, A. Hassidim, and S. Lloyd, *Quantum algorithm for linear systems of equations*, Phys. Rev. Lett. **103**, 150502 (2009).

[Joseph (2020)]
I. Joseph, *Koopman–von Neumann approach to quantum simulation of nonlinear classical dynamics*, Phys. Rev. Res. **2**, 043102 (2020).

[Leyton and Osborne (2008)]
S. K. Leyton and T. J. Osborne, *A quantum algorithm to solve nonlinear differential equations*, arXiv:0812.4423 (2008).

[Lin *et al.* (2022)]
Y. T. Lin, R. B. Lowrie, D. Aslangil, Y. Subasi, and A. T. Sornborger, *Koopman von Neumann mechanics and the Koopman representation: A perspective on solving nonlinear dynamical systems with quantum computers*, arXiv:2202.02188 (2022).


Back

## References

[Liu *et al.* (2021)]
J.-P. Liu, H. Ø. Kolden, H. K. Krovi, N. F. Loureiro, K. Trivisa, and A. M. Childs, *Efficient quantum algorithm for dissipative nonlinear differential equations*, Proc. Nat. Acad. Sci. **118**, e2026805118 (2021).

[Lloyd *et al.* (2020)]
S. Lloyd, G. De Palma, C. Gokler, B. Kiani, Z.-W. Liu, M. Marvian, F. Tennie, and T. Palmer, *Quantum algorithm for nonlinear differential equations*, arXiv:2011.06571 (2020).

[Low and Chuang (2019)]
G. H. Low and I. L. Chuang, *Hamiltonian simulation by qubitization*, Quantum **3**, 163 (2019).

[Martyn *et al.* (2021)]
J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, *Grand unification of quantum algorithms*, PRX Quantum **2**, 040203 (2021).

[Novikau *et al.* (2023)]
I. Novikau, I. Y. Dodin, and E. A. Startsev, *Simulation of linear non-Hermitian boundary-value problems with quantum singular-value transformation*, Phys. Rev. Appl. **19**, 054012 (2023).

[Novikau *et al.* (2024)]
I. Novikau, I. Y. Dodin, and E. A. Startsev, *Encoding of linear kinetic plasma problems in quantum circuits via data compression*, arXiv:2403.11989 (2024), to appear in J. Plasma Phys.

[Novikau *et al.* (2022)]
I. Novikau, E. A. Startsev, and I. Y. Dodin, *Quantum signal processing for simulating cold plasma waves*, Phys. Rev. A **105**, 062444 (2022).

[Parker and Joseph (2020)]
J. B. Parker and I. Joseph, *Quantum phase estimation for a class of generalized eigenvalue problems*, Phys. Rev. A **102**, 022422 (2020).

[Pednault *et al.* (2019)]
E. Pednault, D. Maslov, J. Gunnels, and J. Gambetta, *On "quantum supremacy"*, Quantum Research Blog (2019), https://www.ibm.com/quantum/blog/on-quantum-supremacy

[Preskill (2021)]
J. Preskill, *Quantum computing 40 years later*, arXiv:2106.10522 (2021).

[Scherer *et al.* (2017)]
A. Scherer, B. Valiron, S.-C. Mau, S. Alexander, E. van den Berg, and T. E. Chapuran, *Concrete resource analysis of the quantum linear-system algorithm used to compute the electromagnetic scattering cross section of a 2D target*, Quantum Inf. Process. **16**, 59 (2017).

[Shi *et al.* (2021)]
Y. Shi et al., *Simulating non-native cubic interactions on noisy quantum machines*, Phys. Rev. A **103**, 062608 (2021).

[Suzuki (1990)]
M. Suzuki, *Fractal decomposition of exponential operators with applications to many-body theories and Monte Carlo simulations*, Phys. Lett. A **146**, 319 (1990).

Back