Quantum Algorithms for nonlinear partial differential equations

Shi Jin, Nana Liu

Shanghai Jiao Tong University

Looking beyond classical infrastructure…

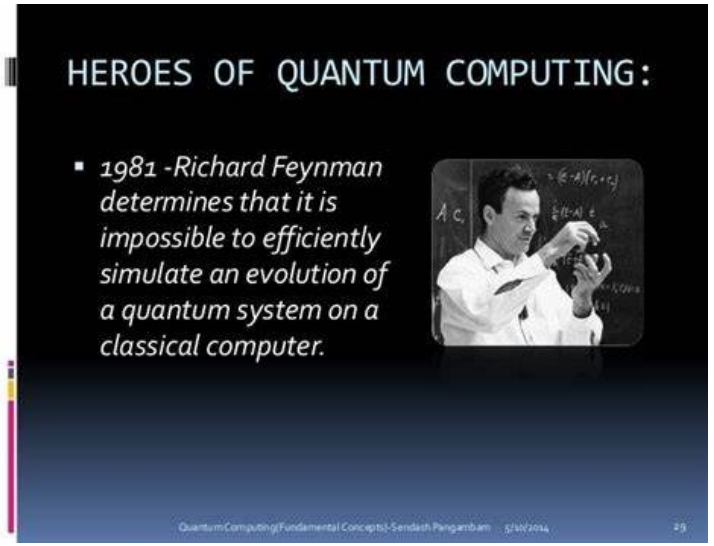
Reaching classical bottlenecks in

a) Size of chips (Moore's law)
b) Communication (bandwidth)
c) Energy cost
d) Memory
e) Time

Also we have naturally occurring quantum data…

Simulate physics with computers

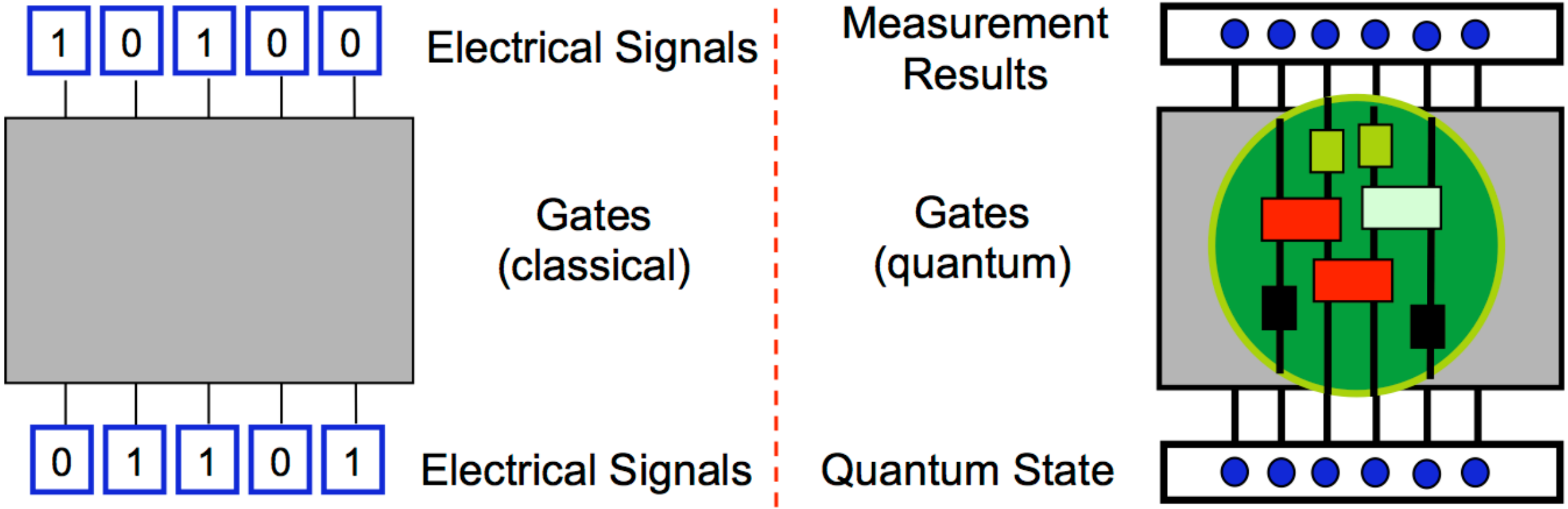
- Can a computer efficiently simulate quantum mechanics?
- Let the computer itself be built of quantum mechanical elements



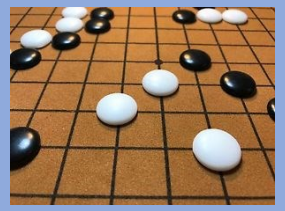
Feynman's conjecture:

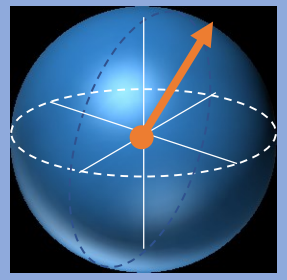
- Quantum systems can simulate other quantum systems
- Possibility of a "universal quantum simulator"
  suggest array of spin ½ particles (now called "qubits")

quantum computation survival kit


OUTPUT
1 0 1 0 0
Electrical Signals
Measurement Results
Gates (classical)
Gates (quantum)
0 1 1 0 1
Electrical Signals
Quantum State
INPUT
Classical Computer
Quantum Computer

Computation is made up of three parts





Bits

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Qubit

$$|\psi\rangle = a\,|0\rangle + b\,|1\rangle = \begin{pmatrix} a \\ b \end{pmatrix}, \quad a, b \in \mathbb{C},$$

$$|a|^2 + |b|^2 = 1.$$

Qubits

$$|\psi\rangle = |\psi_0\rangle \otimes \cdots \otimes |\psi_{m-1}\rangle$$

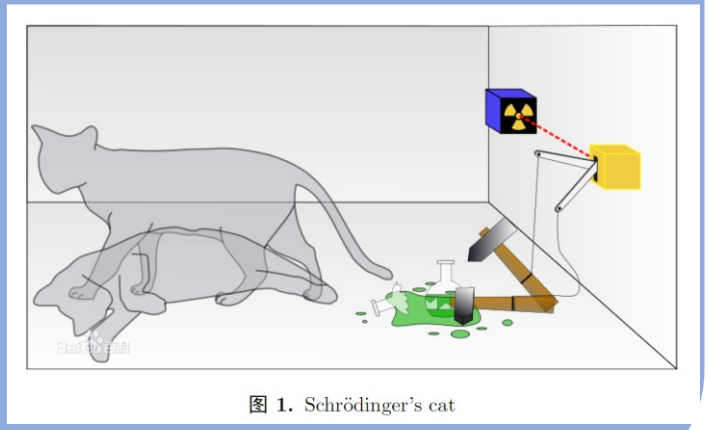


图 1. Schrödinger's cat

$$\mathcal{H} = \mathbb{C}^N \qquad N = 2^n$$

Quantum algorithm cost poly(n): potential exponential speed up

Quantum gates: unitary matrices

Pauli matrices

$$\sigma_x = X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = Y = \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix}, \quad \sigma_z = Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Hadamar gate

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

CNOT gate

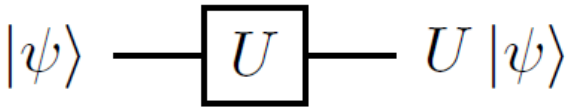
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

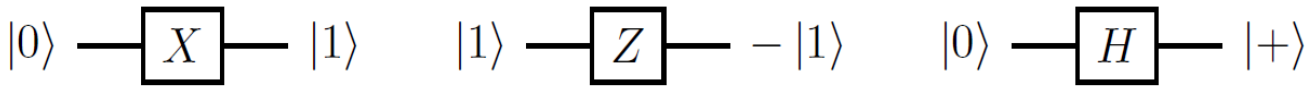
Two-bits

$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

$$\mathrm{CNOT} \begin{cases} |00\rangle & = |00\rangle \\ |01\rangle & = |01\rangle \\ |10\rangle & = |11\rangle \\ |11\rangle & = |10\rangle \end{cases}$$

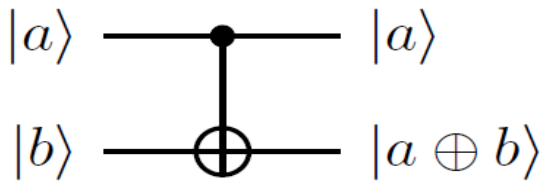
Quantum circuits:



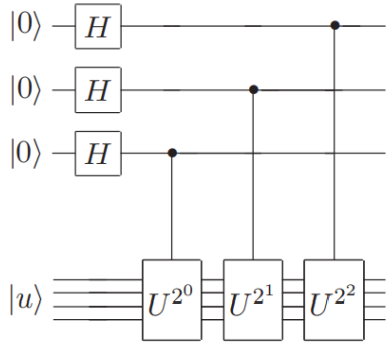


$$X|0\rangle = |1\rangle\,, \quad Z|1\rangle = -|1\rangle\,, \quad H|0\rangle = |+\rangle$$

CNOT gate



Quantum phase estimation



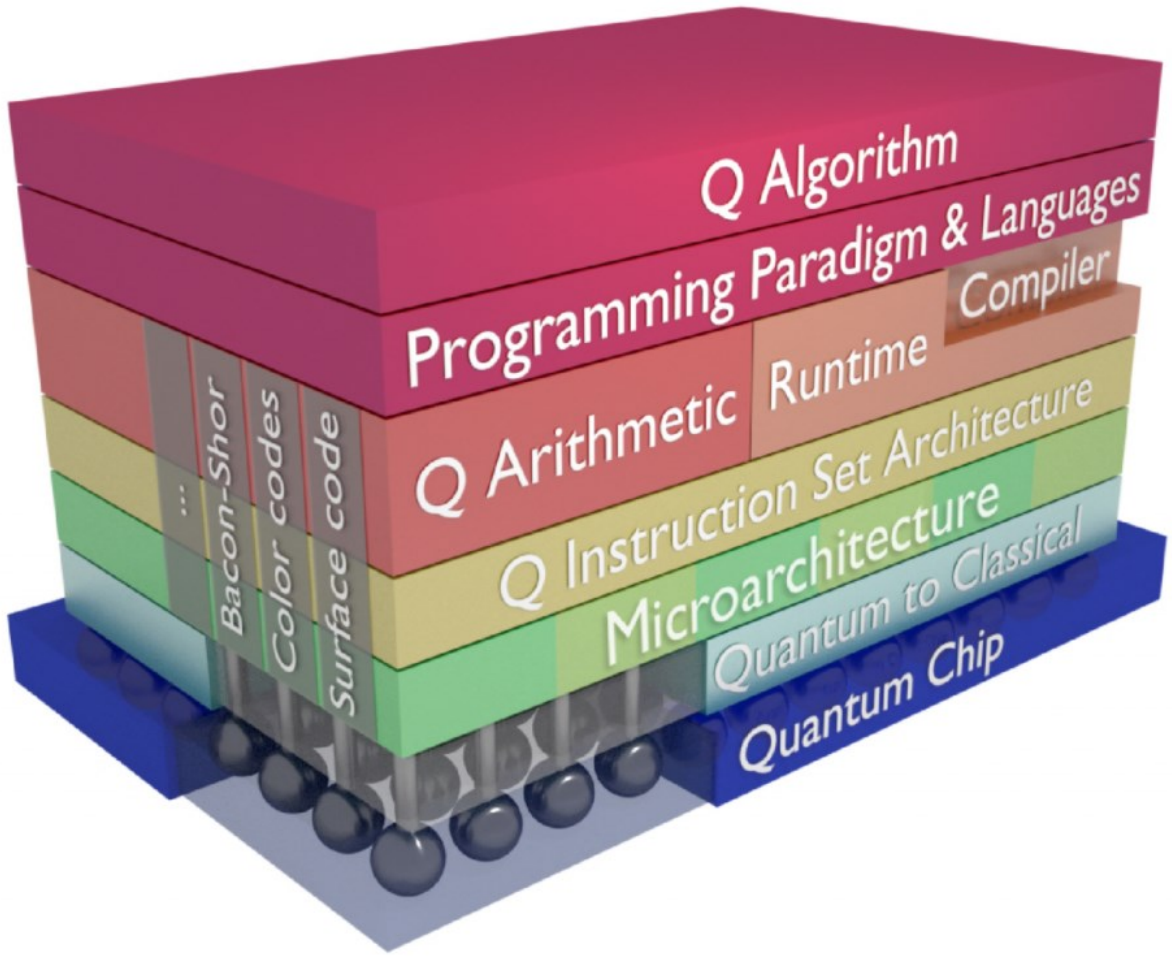
measurement

- The quantum computer outputs a quantum state—a quantum subroutine; one needs measurement to get the physical observables which can be compared with the classical solution

- It measures the probability of certain outcome in the computational basis

- The cost of measurement can be much larger than the quantum subroutines

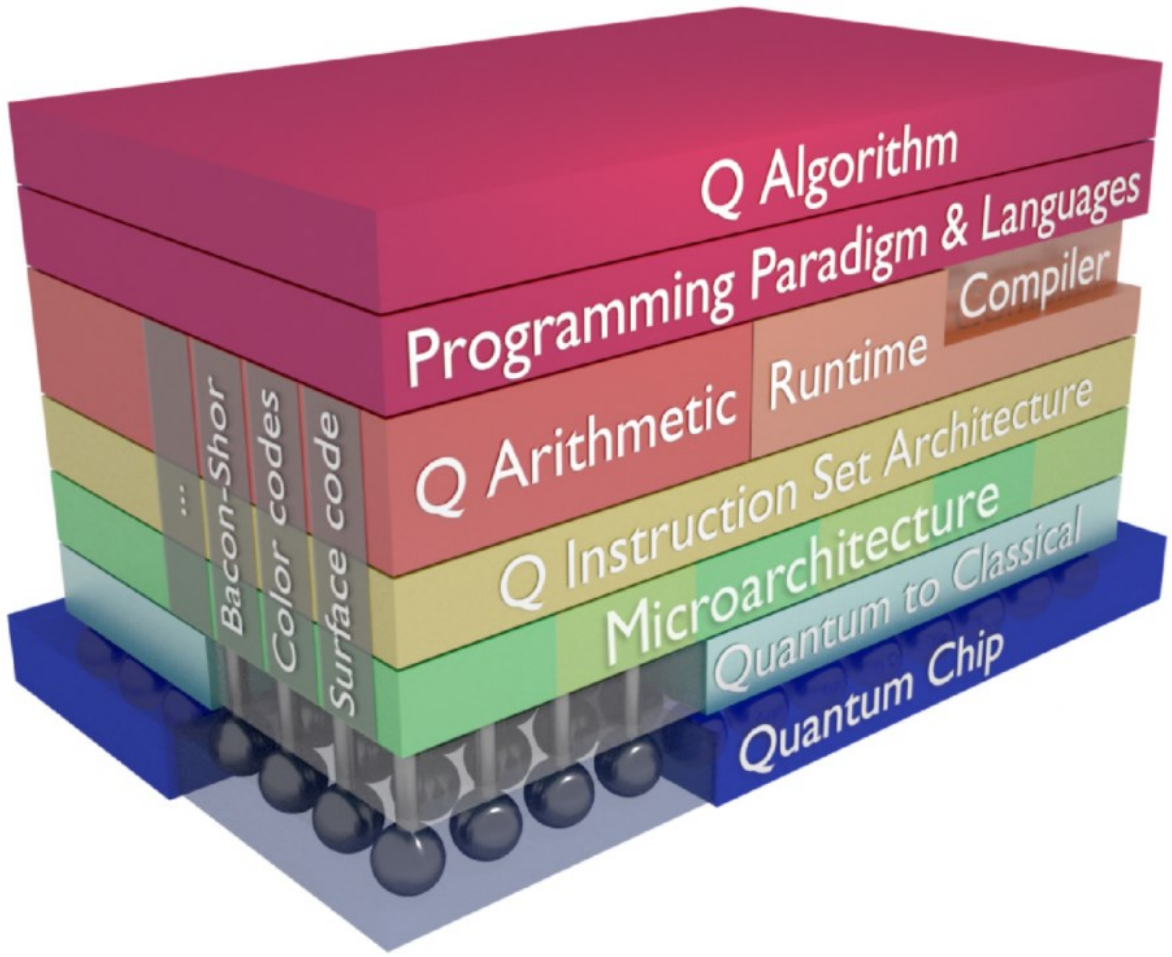
Hardware and software challenges



1. Qubits are noise-prone
2. Error-correction is difficult
3. Loading classical data is difficult

Hardware challenges

Hardware and software challenges



Software challenges

1. New software stack is needed
2. Debugging would be difficult: since cannot read-out intermediate parts of quantum computer
3. Algorithm design is difficult

Linear algebra solvers

So far almost all the quantum algorithms have been designed for linear problems:

- Linear algebra:

  . Harrow-Hassidim-Lloyd 09 (HHL): $O((s\kappa^2\|\mathcal{M}\|_{max}/\eta)poly\log(s\kappa\|\mathcal{M}\|_{max}/\eta))$

  Childs-Kothari-Somma 17 (CKS): $\mathcal{O}(s\kappa\|\mathcal{M}\|_{max}poly\log(s\kappa\|\mathcal{M}\|_{max}/\eta))$

- Linear PDEs→ discretization→LA problem

  usually condition number is O(N) or O(N^2)

  quantum advantage for high dimensional problem: only for

  quantum subroutines (no measurement studied)

  output $|x\rangle$, not x

Tab. 1: Summary of the time complexities of classical and quantum difference methods

Equation	**Classical difference methods**		**Quantum difference methods**		
ODE	Forward Euler	C-N		Explicit	C-N
			Chebyshev	$\widetilde{\mathcal{O}}(N_t^4)$	$\widetilde{\mathcal{O}}(N_t^2)$
	$\mathcal{O}(N_t^2)$	$\mathcal{O}(N_t)$	VTAA	$\widetilde{\mathcal{O}}(N_t)$	$\widetilde{\mathcal{O}}(N_t)$
Heat equation	Explicit	C-N		Explicit	C-N
			HHL	$\widetilde{\mathcal{O}}(d^4N_x^6)$	$\widetilde{\mathcal{O}}(d^4N_x^4)$
	$\mathcal{O}(d^2N_x^{d+2})$	$\mathcal{O}(d^2N_x^{d+1})$	Chebyshev	$\widetilde{\mathcal{O}}(d^4N_x^4)$	$\widetilde{\mathcal{O}}(d^4N_x^2)$
			VTAA	$\widetilde{\mathcal{O}}(d^2N_x^2)$	$\widetilde{\mathcal{O}}(d^2N_x)$
Hyperbolic equation	$\mathcal{O}(d^2N_x^{d+1})$		Chebyshev		VTAA
			$\mathcal{O}(d^4N_x^2\log^{4.5}(d^2N_x/\delta))$		$\widetilde{\mathcal{O}}(d^2N_x)$
Multiscale telegraph equation			Chebyshev		VTAA
	IMEX ($\tau\sim h^2$)	$\mathcal{O}(N_x^3)$	$\mathcal{O}(N_x^4\log^{4.5}(N_x^2/\delta))$		$\widetilde{\mathcal{O}}(N_x^2)$
	Relaxation ($\tau\sim h^2$)	$\mathcal{O}(N_x^3)$	$\widetilde{\mathcal{O}}(N_x^4\log^{4.5}(N_x^2/\delta))$		$\widetilde{\mathcal{O}}(N_x^2)$
	Penalized ($\tau\sim h^2$)	$\mathcal{O}(N_x^3\log(1/\delta))$	$\widetilde{\mathcal{O}}(N_x^4)\log^{4.5}(N_x^2/\delta)$		$\widetilde{\mathcal{O}}(N_x^2)$
	Penalized ($\tau\sim h$)	$\mathcal{O}(N_x^{2.5}\log(1/\delta))$	$\mathcal{O}(N_x^2)\log^{4.5}(N_x/\delta)$		$\widetilde{\mathcal{O}}(N_x)$
	Explicit ($\tau=\varepsilon h$)	$\mathcal{O}(N_x^2/\varepsilon)$	$\geq\mathcal{O}(\log^2(\sqrt{\varepsilon}/\delta)\log(N_x)/\varepsilon)$		$\geq\widetilde{\mathcal{O}}(1/\sqrt{\varepsilon})$

[a] $\widetilde{\mathcal{O}}$ means some logarithmic terms are ignored.

[b] δ is the desired error bound of the quantum linear solver.

[c] ε is the relaxation time or the scaling parameter.

Jin-Liu-Yu '22

Nonlinear ODEs and PDEs?

D ODEs

$$\frac{dX^{[k]}(t)}{dt} = F(X^{[k]}(t)), \qquad X^{[k]} \in \mathbb{R}^D$$
$$X^{[k]}(0) = X_0^{[k]}, \qquad k = 1, \cdots, M$$

Nonlinear F

$(d+1)$-dim PDEs

$$\frac{\partial u^{[k]}}{\partial t} + F(u^{[k]}, \nabla u^{[k]}, \nabla^2 u^{[k]}, ...) = 0$$
$$u^{[k]} \in \mathbb{R}^d \qquad k = 1, ..., M$$

Nonlinear F

Nonlinear systems: challenges

Applications in fluid dynamics (Navier-Stokes), gas dynamics, molecular dynamics, financial markets, machine learning…etc

Appearance of discontinuities and shock solutions and singularities; curse of dimensionality

Often statistical methods are employed: to understand ensemble behaviour (e.g. statistical behaviour of fluids)

Parameters: d, D, ϵ, T, M

$(d+1)$-dimensional nonlinear PDE with M initial conditions

General:

$$\frac{\partial u^{[k]}}{\partial t} + F(u^{[k]}, \nabla u^{[k]}, \nabla^2 u^{[k]}, \cdots) = 0, \quad t \in \mathbb{R}^+, x \in \mathbb{R}^d, u^{[k]} \in \mathbb{R}^d, k = 1, ..., M$$

Here $t \geq 0$ is time, x is the spatial variable, while F is a nonlinear function or functional.

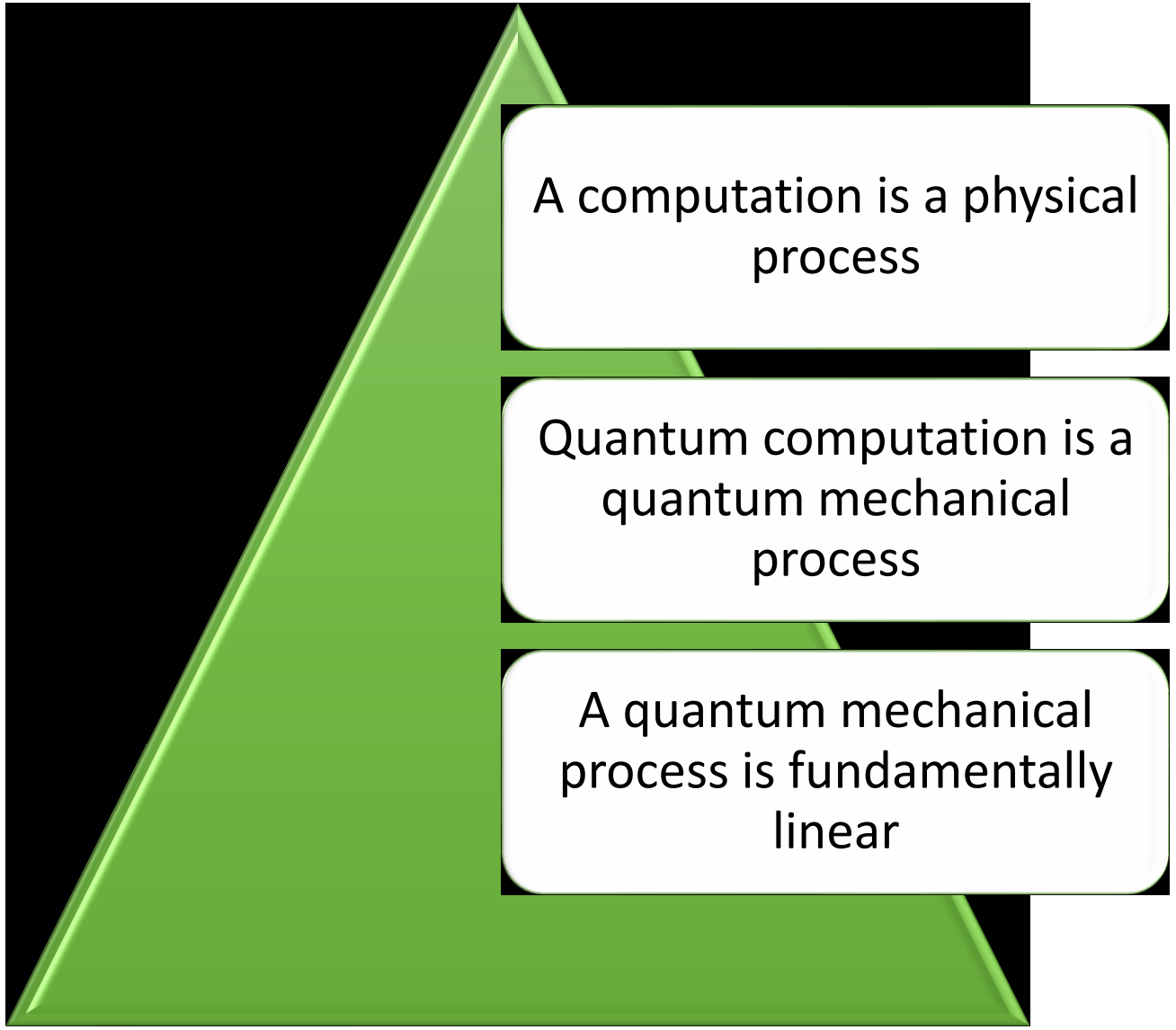
Hamilton-Jacobi:

$$\partial_t S^{[k]} + H(\nabla S^{[k]}, x) = 0, \quad t \in \mathbb{R}^+, x \in \mathbb{R}^d, S^{[k]}(t, x) \in \mathbb{R},$$
$$S^{[k]}(0, x) = S_0^{[k]}(x), \quad k = 1, ..., M$$

Hyperbolic:

$$\partial_t u^{[k]} + F(u^{[k]}) \cdot \nabla_x u^{[k]} + Q(x, u^{[k]}) = 0, \quad t \in \mathbb{R}^+, \quad x \in \mathbb{R}^d,$$
$$u^{[k]}(0, x) = u_0^{[k]}(x), \quad k = 1, ..., M.$$

Are nonlinear problems suitable for quantum computation?



Nonlinear Quantum Mechanics Implies Polynomial-Time Solution for *NP*-Complete and #*P* Problems

Daniel S. Abrams*
Department of Physics, MIT 12-128b, Cambridge, Massachusetts 02139

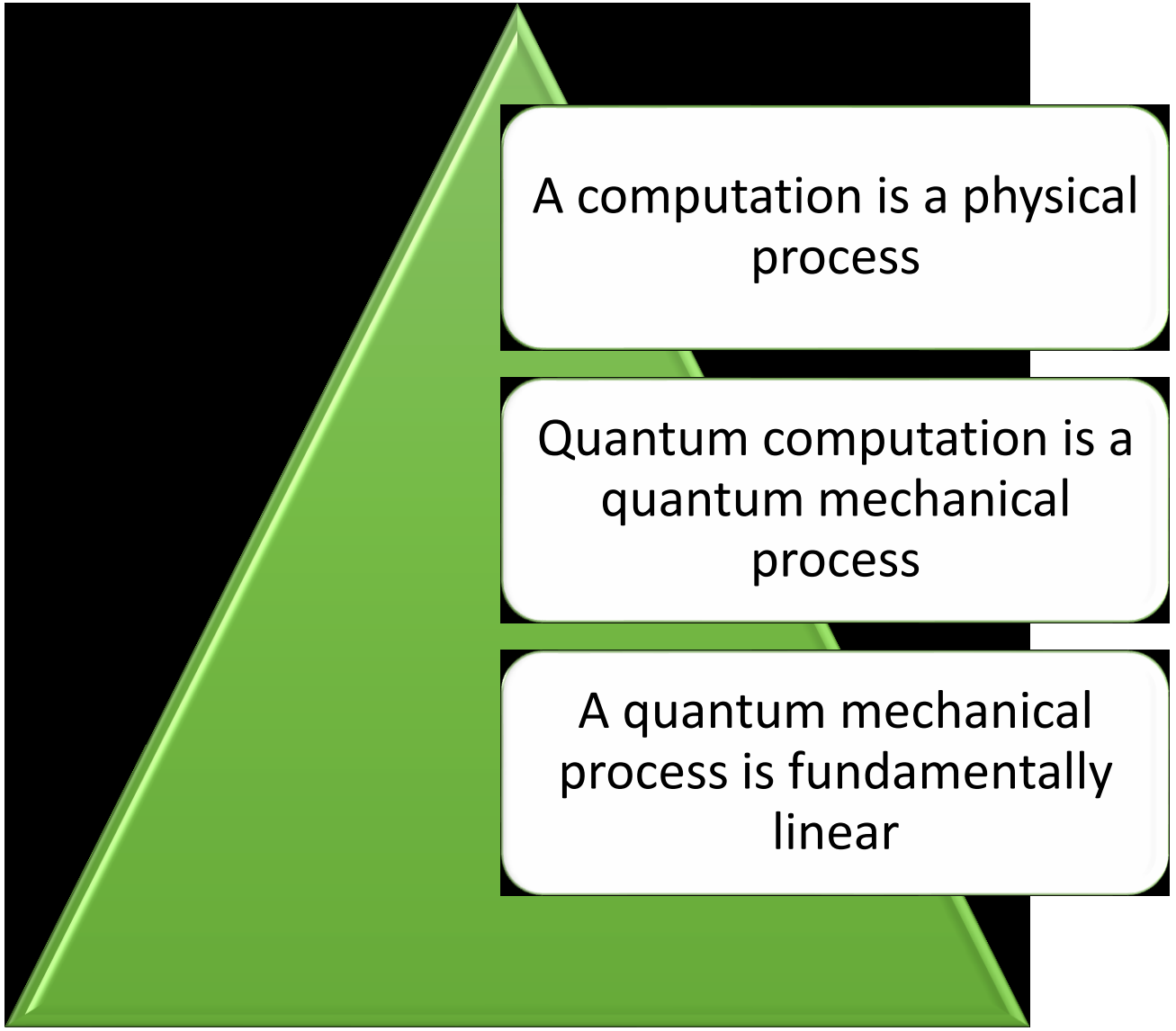
Seth Lloyd†
d'Arbeloff Laboratory for Information Sciences and Technology, Department of Mechanical Engineering, MIT 3-160, Cambridge, Massachusetts 02139
(Received 27 January 1998)

berg models of nonlinear quantum mechanics. Finally, we would like to note that we believe that quantum mechanics is in all likelihood exactly linear, and that the above conclusions might be viewed most profitably as further evidence that this is indeed the case. Nevertheless, the theoretical implications and practical applications that would result from a discovery to the contrary may warrant further investigation into the matter.

Two routes:

1) Make the problem linear

2) Don’t use fundamental quantum mechanics
(which works at best for small and special nonlinearity (Abram-Lloyd))



Basic roadmap:

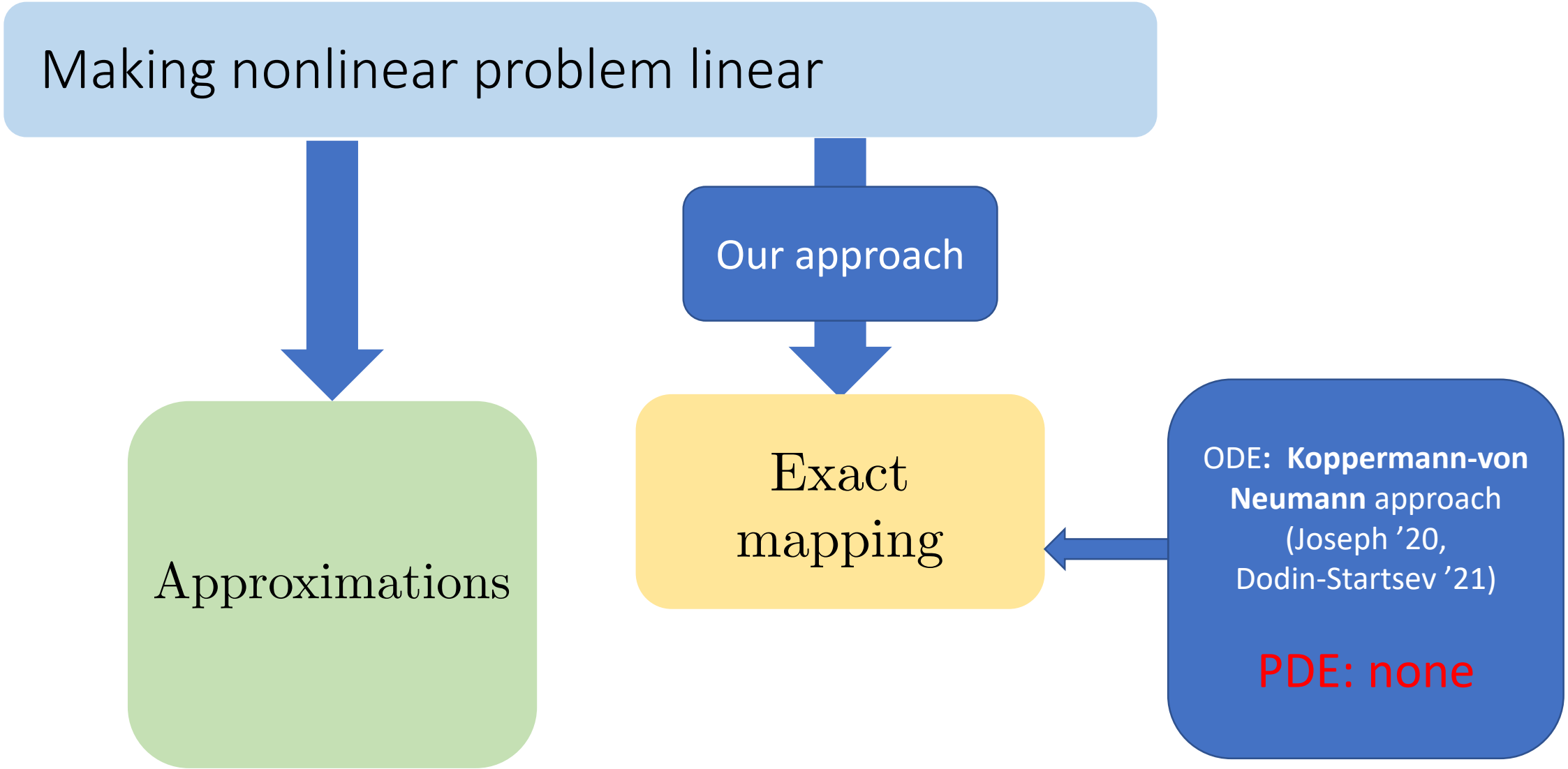
Making nonlinear problem linear

Approximations

1) Based on linearization (e.g. Carlemann) and truncation
2) Works only on weak or quadratic nonlinearity
3) Valid for short time

Leyton & Osborne, 2008; Lloyd et al, 2020; Joseph, 2020; Liu et al, 2021…etc

Basic roadmap:



Part A: Hamilton-Jacobi and hyperbolic PDEs

Level set mapping

$$\partial_t S^{[k]} + H(\nabla S^{[k]}, x) = 0, \quad t \in \mathbb{R}^+, x \in \mathbb{R}^d, S^{[k]}(t,x) \in \mathbb{R},$$
$$S^{[k]}(0,x) = S_0^{[k]}(x), \quad k = 1, ..., M$$

$$\text{Define } u^{[k]} = \nabla S^{[k]} \in \mathbb{R}^d.$$

The level set function $\phi_i^{[k]}(t,x,p)$ can be defined by

$$\phi_i^{[k]}(t,x,p = u^{[k]}(t,x)) = 0$$

Level set function lives in a space with twice the *dimension*

Then $\phi^{[k]} = (\phi_1^{[k]}, ..., \phi_d^{[k]})$ solves the *linear* PDE

$$\partial_t \phi^{[k]} + \nabla_p H \cdot \nabla_x \phi^{[k]} - \nabla_x H \cdot \nabla_p \phi^{[k]} = 0.$$

$$\phi_i^{[k]}(0,x,p) = p_i - u_i^{[k]}(0,x), \quad i = 1, \cdots, d.$$

Jin-Osher, CMS '03

Liouville equation

Quantum computer does not know how to find the zero level set!

Level set mapping

Instead solve for ψ

$$\psi(t,x,p) = \frac{1}{M}\sum_{k=1}^{M}\delta(\phi^{[k]}(t,x,p))\,.$$

$$\partial_t\psi + \nabla_p H\cdot\nabla_x\psi - \nabla_x H\cdot\nabla_p\psi = 0$$

$$\psi(0,x,p) = \frac{1}{M}\sum_{k=1}^{M}\prod_{i=1}^{d}\delta(p_i - u_i^{[k]}(0,x)).$$

M different initial data computed once!

Physical observables and ensemble average

Given any function $G : \mathbb{R}^d \to \mathbb{R}$

$$\langle G(t,x)\rangle \equiv \int_{\mathbb{R}^d} G(p)\psi(t,x,p)dp = \frac{1}{M}\sum_{k=1}^{M}\int_{\mathbb{R}^d} G(p)\delta(\phi^{[k]}(t,x,p))dp$$

ψ acts like the Wigner function
for nonlinear HJE in WKB approx

Example:

$$i\hbar\partial_t\Psi = -\frac{\hbar^2}{2}\Delta\Psi + V(x)\Psi\,, \qquad \Psi(0,x) = A_0(x)e^{i\frac{S_0(x)}{\hbar}}$$

S satisfies HJ PDE with nonlinear $H = (1/2)|\nabla S|^2 + V(x)$

Physical observables

Given any function $G : \mathbb{R}^d \to \mathbb{R}$

$$\langle G(t,x)\rangle \equiv \int_{\mathbb{R}^d} G(p)\psi(t,x,p)dp = \frac{1}{M}\sum_{k=1}^{M}\int_{\mathbb{R}^d} G(p)\delta(\phi^{[k]}(t,x,p))dp$$

ψ acts like the Wigner function
for nonlinear HJE in WKB approx

Example: $w \leftrightarrow \psi$

$$\partial_t w + \nabla_p H \cdot \nabla_x w - \nabla_x H \cdot \nabla_p w = 0$$

$$w(0,x,p) = |A_0(x)|^2 \delta(p - \nabla S_0(x))$$

Physical observables

Given any function $G : \mathbb{R}^d \to \mathbb{R}$

$$\langle G(t,x)\rangle \equiv \int_{\mathbb{R}^d} G(p)\psi(t,x,p)dp = \frac{1}{M}\sum_{k=1}^{M}\int_{\mathbb{R}^d} G(p)\delta(\phi^{[k]}(t,x,p))dp$$

ψ acts like the Wigner function
for nonlinear HJE in WKB approx

Example: $G(p) = 1, p, |p|^2$

$$\rho(t,x) = \int w\,dp\,, \quad \rho(t,x)u(t,x) = \int pw\,dp\,, \quad \frac{1}{2}\rho(t,x)u^2(t,x) = \int \frac{|p|^2}{2} w\,dp$$

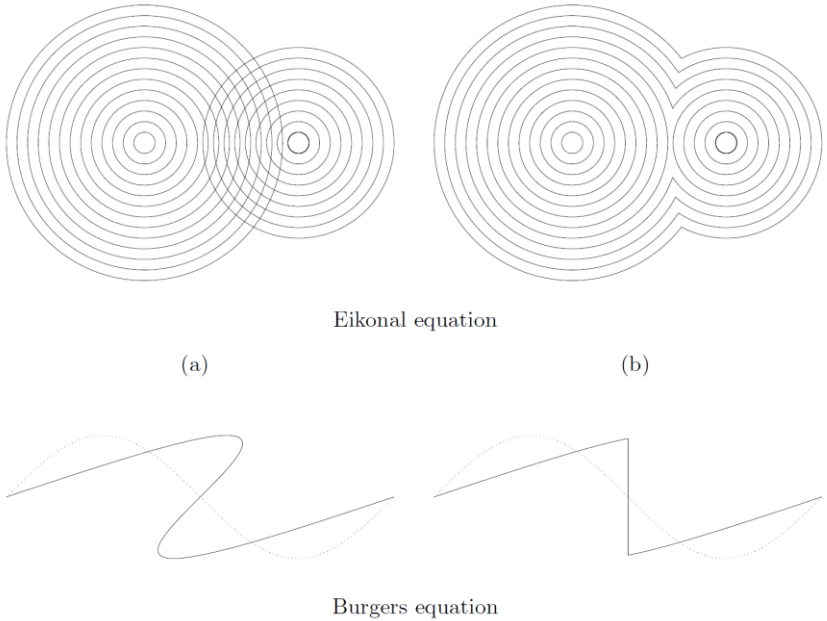
First 3 moments give classical limits to
density, momentum and kinetic energy

$$|\Psi|^2, \hbar\mathrm{Im}(\bar{\Psi}\nabla\Psi), (\hbar^2/2)|\nabla\Psi|^2$$

Remarks

- The solutions we compute are multivalued solutions, not viscosity solutions!

- Multivalued solutions are the correct solutions for non-dissipative systems (geometric optics, Hamiltonian systems, semi-classical limit of quantum dynamics, high frequency limit of elastic, electromagnetic waves, Dirac equations, etc.



- Same ideas also work for scalar nonlinear hyperbolic equations, general nonlinear ODEs

Part B: quantum algorithms

Transform to a linear algebra problem

- Reducing *linear* PDEs to the basic matrix inversion problem through suitable discretisation that also obeys stability conditions, e.g., upwind scheme + CFL condition

$$\frac{\partial \Phi(t_n, q)}{\partial t} \to \frac{\Phi_{n+1,\boldsymbol{j}} - \Phi_{n,\boldsymbol{j}}}{\Delta t}$$

$$\frac{\partial}{\partial q_i}(F_i(q)\Phi(t_n, q)) \to \frac{1}{h}\left[(F_i^-(q_{i+1/2})T_i^+\Phi_n)_{\boldsymbol{j}} - (F_i^+(q_{i-1/2})T_i^-\Phi_{n,\boldsymbol{j}} + [(F_i^+(q_{j+1/2}) - F_i^-(q_{j-1/2}))\Phi_n]_{\boldsymbol{j}}\right]$$

Hermitian $\mathcal{M}$

$$\mathcal{K}\begin{pmatrix} \Phi_{1,\boldsymbol{j}} \\ \Phi_{2,\boldsymbol{j}} \\ \vdots \\ \Phi_{N_t-1,\boldsymbol{j}} \\ \Phi_{N_t,\boldsymbol{j}} \end{pmatrix} = \begin{pmatrix} \Phi_{0,\boldsymbol{j}} \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

$$\mathcal{M} = \begin{pmatrix} 0 & \mathcal{K} \\ \mathcal{K}^\dagger & 0 \end{pmatrix}$$

quantum linear systems problem (QLSP)

- Quantum computers are good at performing matrix inversion:

  `solving' linear equations $\mathcal{M}x = y$

$$2^m \times 2^m \text{ matrix}$$

$$\text{Classical algorithm} \quad \mathcal{O}((2^m)^3)$$

$$\text{Quantum algorithm} \quad \mathcal{O}(\log(2^m))$$

Caveats! [2] Aaronson, Nature, 2014

Brief into to quantum linear systems problem (QLSP)

Problem 1. ***(QLSP)*** *Let $\mathcal{M}$ be a $2^m \times 2^m$ Hermitian matrix such that $\|\mathcal{M}\| \leq 1$. Assume vectors x and y with elements $\{x_i\}$, $\{y_i\}$ that satisfy $\mathcal{M}x = y$. One can then define the following m-qubit quantum states $|x\rangle \equiv \sum_i x_i/N_x|i\rangle$, $|y\rangle \equiv \sum_i y_i/N_y|i\rangle$ where $N_x = \sqrt{\sum_i |x_i|^2}$, $N_y = \sqrt{\sum_i |y_i|^2}$ are normalisation constants. The aim of any QLSP algorithm is, when given access to $\mathcal{M}$ and unitary $U_{initial}$ (where $U_{initial}|0\rangle = |y\rangle$), to prepare the quantum state $|x'\rangle$ that is η-close to $|x\rangle$, i.e., $\||x'\rangle - |x\rangle\| \leq \eta$.*

$\mathcal{M}x = y$ → $\mathcal{M}|x\rangle = |y\rangle$ → $|x\rangle = \mathcal{M}^{-1}|y\rangle$

$\mathcal{M}$ accessed through oracles

How many times is the oracle queried?

Not the solution of the original classical problem!

State preparation not enough: need to solve the system of linear equations problem (SLEP)

Problem 2. ***(SLEP)*** *Given a Hermitian matrix $\mathcal{G}$, which is of the same size as $\mathcal{M}$, access to $\mathcal{M}$ and $U_{initial}$, the aim of SLEP is to compute the expectation value $(\mathcal{M}^{-1}y)^T\mathcal{G}(\mathcal{M}^{-1}y) = x^T\mathcal{G}x$ to precision ϵ'.*

$$\mathcal{M}x = y$$

$$x^T\mathcal{G}x \propto \langle x|\mathcal{G}|x\rangle$$

$\mathcal{M}$ accessed through oracles

$\mathcal{G}$ accessed through oracles

How many times are the oracles queried?

Simulation through oracle models

Simulating approximation to $\mathcal{M}^{-1}$ through sparse access:

Sparse access to a Hermitian matrix $\mathcal{M}$ is a 4-tuple $(s, \|\mathcal{M}\|_{max}, O_{\mathcal{M}}, O_F)$ and the $(i,j)^{th}$ entry of $\mathcal{M}$ is denoted $\mathcal{M}_{ij}$. Here s is the sparsity of $\mathcal{M}$ and $\|\mathcal{M}\|_{max} = \max_{i,j}(|\mathcal{M}_{ij}|)$ is the max-norm of $\mathcal{M}$. O_M and O_F are unitary black boxes which can access the matrix elements $\mathcal{M}_{ij}$ such that

$$O_M|j\rangle|k\rangle|z\rangle = |j\rangle|k\rangle|z \oplus \mathcal{M}_{jk}\rangle$$
$$O_F|j\rangle|l\rangle = |j\rangle|F(j,l)\rangle$$

where the function F takes the row index j and a number $l = 1, 2, ..., s$ and outputs the column index of the l^{th} non-zero elements in row j.

Our work:
Computing physical observable

Discretising ψ and the delta function

$$\psi^{\omega}_{0,\boldsymbol{j},\boldsymbol{l}} = \frac{1}{M}\sum_{k=1}^{M}\prod_{i=1}^{d}\delta_{\omega}(l_i h - u_i^{[k]}(n=0,h\boldsymbol{j})).$$

$$\begin{pmatrix}\mathbf{0}\\ \psi_{n,\boldsymbol{j},\boldsymbol{l}}\end{pmatrix} = \mathcal{M}^{-1}\begin{pmatrix}\psi_{0,\boldsymbol{j},\boldsymbol{l}}\\ \mathbf{0}\end{pmatrix}$$

Given any function $G:\mathbb{R}^d\to\mathbb{R}$

$$\langle G(t,x)\rangle \equiv \int_{\mathbb{R}^d} G(p)\psi(t,x,p)dp = \frac{1}{M}\sum_{k=1}^{M}\int_{\mathbb{R}^d} G(p)\delta(\phi^{[k]}(t,x,p))dp$$

$$\langle G(t,x)\rangle \approx \ \langle G^{\omega}_{n,\boldsymbol{j}}\rangle \equiv \frac{1}{N^d}\sum_{\boldsymbol{l}}^{N} G_{\boldsymbol{l}}\psi_{n,\boldsymbol{j},\boldsymbol{l}}.$$

Computing physical observable

Estimate with a quantum algorithm:

$$\langle G^{\omega}_{n,\boldsymbol{j}} \rangle \equiv \frac{1}{N^d} \sum_{\boldsymbol{l}}^{N} G_{\boldsymbol{l}} \psi_{n,\boldsymbol{j},\boldsymbol{l}}.$$

Main players:

$$|\psi_0\rangle = \frac{1}{N_{\psi_0}} \sum_{\boldsymbol{j}}^{N} \sum_{\boldsymbol{l}}^{N} \psi_{0,\boldsymbol{j},\boldsymbol{l}} |\boldsymbol{j}\rangle |\boldsymbol{l}\rangle |n=0\rangle$$

$$|G_{n,\boldsymbol{j}}\rangle \equiv \frac{1}{N_G} \sum_{\boldsymbol{l}}^{N} G^*_{\boldsymbol{l}} |\boldsymbol{l}\rangle |\boldsymbol{j}\rangle |n\rangle \qquad \mathcal{G} = |G_{n,\mathbf{j}}\rangle\langle G_{n,\mathbf{j}}|$$

$\mathcal{M}$: sparse oracle access

$$\langle G^{\omega}_{n,\boldsymbol{j}} \rangle \equiv \frac{1}{N^d} \sum_{\boldsymbol{l}=1}^{N} G_{\boldsymbol{l}} \psi_{n,\boldsymbol{j},\boldsymbol{l}} = \frac{1}{N^d} N_{\psi_0} N_G |\sqrt{\Upsilon}| = n_{\psi_0} n_G |\sqrt{\Upsilon}|$$

$$\Upsilon \equiv \langle \psi_0 | (\mathcal{M}^{-1})^{\dagger} \mathcal{G} \mathcal{M}^{-1} | \psi_0 \rangle$$

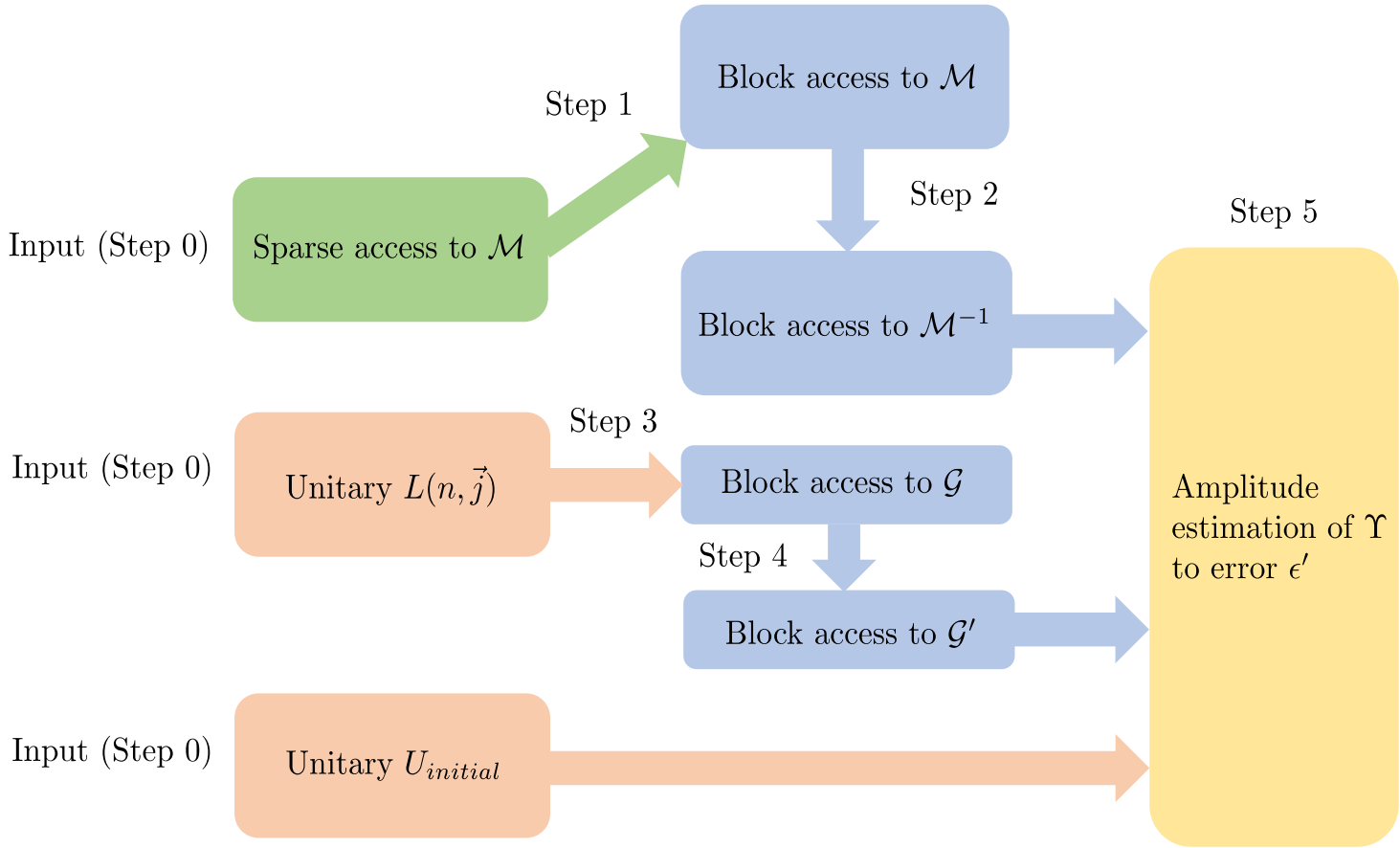
To design quantum algorithm for this...

Computing physical observable

Sanders ‘22

To design quantum algorithm for this...

$$\Upsilon \equiv \langle \psi_0 | (\mathcal{M}^{-1})^\dagger \mathcal{G} \mathcal{M}^{-1} | \psi_0 \rangle$$



Main theorems:

A quantum algorithm that takes sparse access $(s = O(d), \|\mathcal{M}\|_{max} = O(1), O_M, O_F)$ *to* $\mathcal{M}$*, where* $\|\mathcal{M}\| = O(1)$*, and access to the unitaries* $L(t_n, \boldsymbol{j}/N)$*, where* $L(t_n, \boldsymbol{j}/N)|0\rangle = |G_{n,\boldsymbol{j}}\rangle$ *and* $U_{initial}$*, where* $U_{initial}|0\rangle = |\psi_0\rangle$*, is able to estimate the ensemble average* $\langle G(T,x)\rangle$ *at time* $T = t_n$*, with* M *initial data, to precision* ϵ *with an upper bound on the query complexity* $\mathcal{Q}$

$$\mathcal{Q} = \mathcal{O}\left(\frac{n_{\psi_0}^2 d^7 T^3}{\epsilon^{10}} \log\left(\frac{n_{\psi_0}^2 d^4 T^2}{\epsilon^7}\right)\right)$$

and the same order of additional 2-qubit gates, where we suppress all $O(1)$ *terms except* $T = O(1)$ *and* $n_{\psi_0} \geq O(1)$.

Main theorems:

Comparison to classical algorithm $\mathcal{C} = O(MTd^{d+4}(1/\epsilon)^{d+1})$

Let $\mathcal{C}$ be the cost to compute an observable from a $(d+1)$-dimensional Hamilton-Jacobi equation with purely classical methods and $\mathcal{Q}$ be the cost of our quantum algorithm. We say there is a quantum advantage in estimating the observables when $\mathcal{Q} = o(\mathcal{C})$. To attain a quantum advantage it is then sufficient for the following condition to hold

$$o\left(\frac{Md^{d-4}}{n_{\psi_0}^2 T^2}\left(\frac{1}{\epsilon}\right)^{d-9}\right) = \tilde{O}(1)$$

where $\tilde{O}$ suppresses all logarithmic terms in $d, 1/\epsilon, T$.

Similar results for scalar hyperbolic PDEs

Part B: General nonlinear PDEs

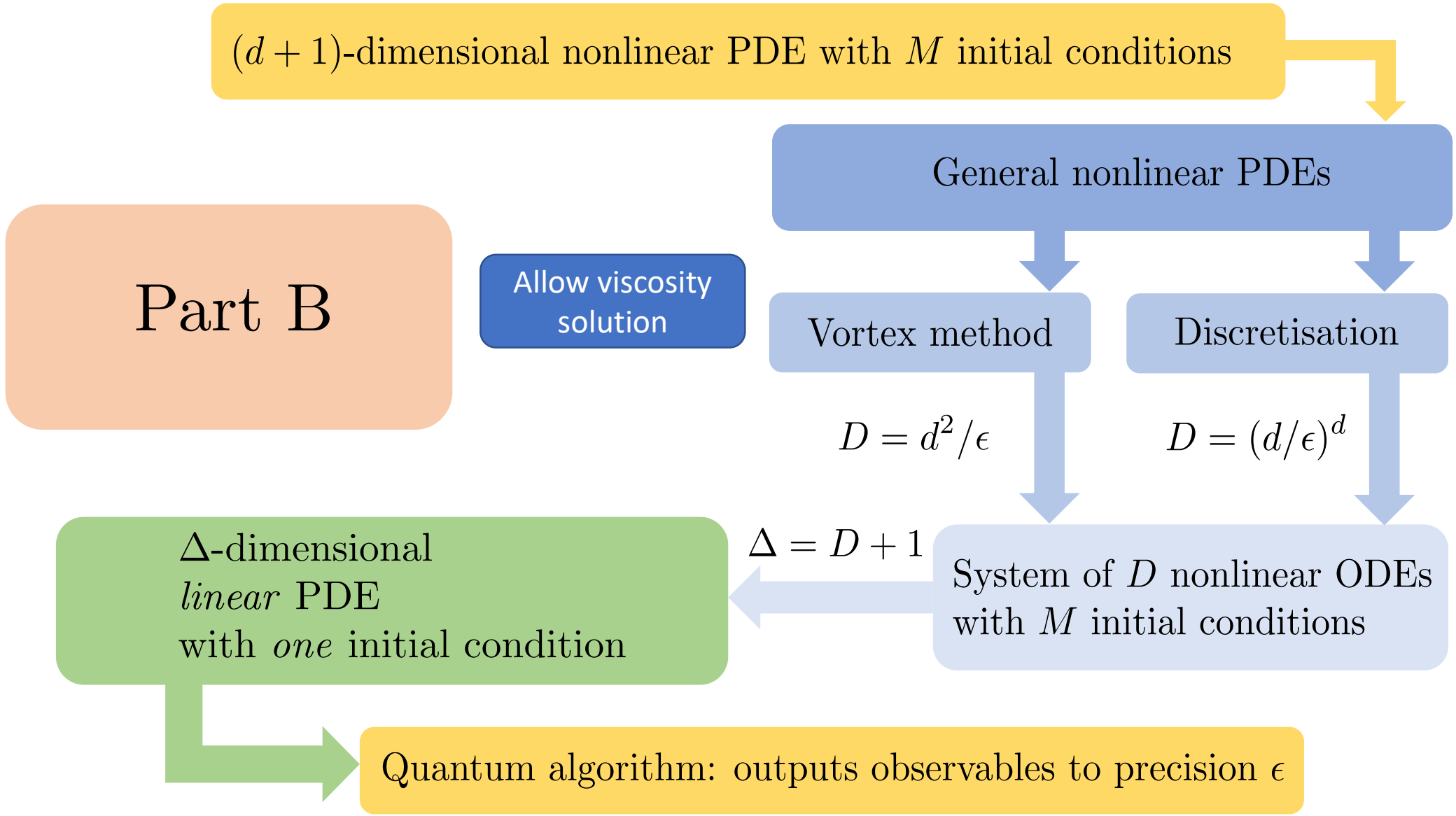
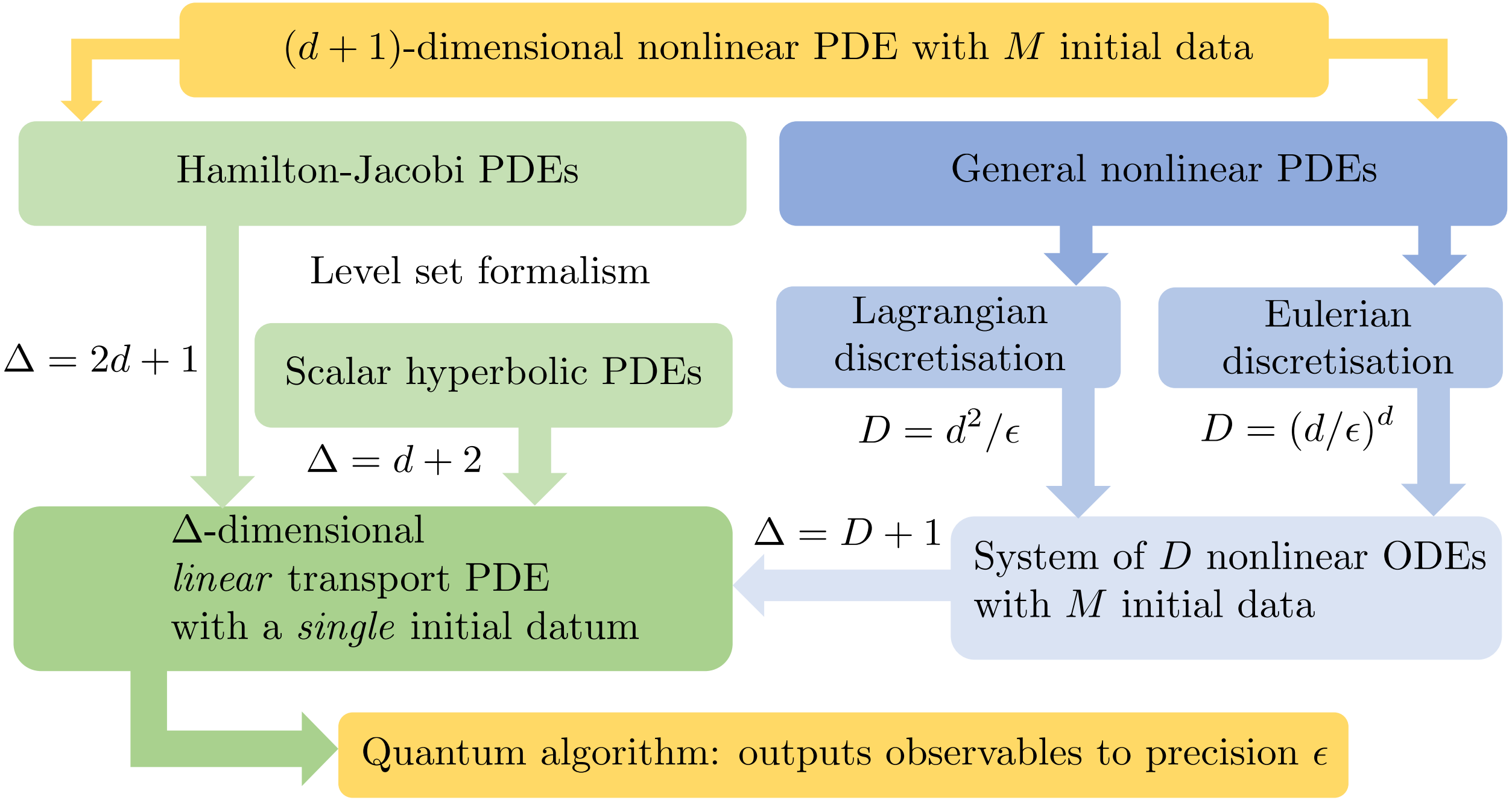

$(d+1)$-dimensional nonlinear PDE with M initial conditions
General nonlinear PDEs
Part B
Allow viscosity solution
Vortex method
Discretisation
$D = d^2/\epsilon$
$D = (d/\epsilon)^d$
Δ-dimensional
linear PDE
with *one* initial condition
$\Delta = D + 1$
System of D nonlinear ODEs
with M initial conditions
Quantum algorithm: outputs observables to precision ϵ

TABLE I. Quantum and classical cost comparison in computing observables to precision ϵ

$$\text{PDE: } \mathcal{O}\left(\frac{\mathcal{C}}{\mathcal{Q}}\right) = \tilde{O}\left(\frac{M}{T^2} d^{r_1} \left(\frac{1}{\epsilon}\right)^{r_2}\right)$$

$$\text{ODE: } \mathcal{O}\left(\frac{\mathcal{C}}{\mathcal{Q}}\right) = \tilde{O}\left(\frac{M}{T^2} D^{r_1} \left(\frac{1}{\epsilon}\right)^{r_2}\right)$$

Nonlinear equations (M initial conditions)	r_1	r_2	b range (initial condition-dependent)	Quantum advantage (possible)
$(d+1)$-dimensional Hamilton-Jacobi PDE	$d-4-b$	$d-9-3b$	$b \in [0, \frac{d}{3}-3)$	M, d, ϵ
$(d+1)$-dimensional hyperbolic PDE	$d-5-b$	$d-9-3b$	$b \in [0, \frac{d}{3}-3)$	M, d, ϵ
System of D ODEs	-5	-9	$b=0$	M
$(d+1)$-dimensional general PDE (vortex)	-7	-13	$b=0$	M
$(d+1)$-dimensional general PDE (discretisation)	$-4d$	$-9-4d$	$b=0$	Large M


$(d+1)$-dimensional nonlinear PDE with M initial data
Hamilton-Jacobi PDEs
General nonlinear PDEs
Level set formalism
$\Delta = 2d+1$
Scalar hyperbolic PDEs
$\Delta = d+2$
Lagrangian discretisation
$D = d^2/\epsilon$
Eulerian discretisation
$D = (d/\epsilon)^d$
Δ-dimensional *linear* transport PDE with a *single* initial datum
$\Delta = D+1$
System of D nonlinear ODEs with M initial data
Quantum algorithm: outputs observables to precision ϵ

Final comments

- The physical world is fundamentally linear, at the quantum (and microscopic--Newton's level-via the Liouville equation)
- The macroscopic models are often nonlinear, due to taking mean-field limit (use of molecular chaos), moment closures, etc. It reduces the dimension (good for classical computation: microscopic to macroscopic) but introduces nonlinearity (mathematical analysis more challenging)
- For quantum computation curse-of-dimensionality is less of a concern while nonlinearity is; so we move in backward direction: macroscopic to microscopic
- Dimension cannot be too high:

  d→2d OK (Hamilton-Jacobi, scalar nonlinear hyperbolic equations);

  d→N^d not ok if one discretizes nonlinear PDE spatially by N mesh points

  and solve the resulting nonlinear ODEs (general nonlinear PDEs)

Summary:

Claim:

- **first quantum nonlinear PDE solver based on linear representation—exact formulation without error before numerical discretization**

  possible quantum advantage in M, d, ϵ

  (before: only used linear approximation thus models are valid only for short time)

- **these quantum advantages are achieved even by including the measurement of physical observables**

  (before: only provided quantum subroutines and no measurement step which could lose the quantum advantage)

Open:

general PDEs, viscosity solutions, high dimensional dynamical systems, …