多体問題の計算科学
Computational Science for Many-Body Problems

#2 Why many-body problem is hard to solve

15:10-16:40 April 13, 2021

0. Computer

Why many-body problem is hard to solve

1. *N*-body Newtonian equation of motion
2. *N*-body classical statistical mechanics
3. *N*-body Schrödinger equation
4. *N*-body quantum statistical mechanics

The slides of this lecture are uploaded in ITC-LMS & Github
https://github.com/compsci-alliance/many-body-problems

Computer

High Performance Computing Infrastructure

High Performance Computing Infra. (HPCI) in Japan
-Flagship (Fugaku@RIKEN)
-2nd tier (GC@Hokudai, Oakforest-PACS, OBCX@UTokyo, …)

For daily use: PC, PC cluster, cloud (AWS, GCP, Azure, …)

Latest TOP500 (Flops)
https://www.top500.org/lists/top500/2020/11/

1. Supercomputer Fugaku (A64FX)
2. Summit (POWER9+NVIDIA Volta*)
3. Sierra (POWER9+NVIDIA Volta*)
4. Sunway TaihuLight (Sunway)
5. Selene (AMD EPYC+NVIDIA A100*)

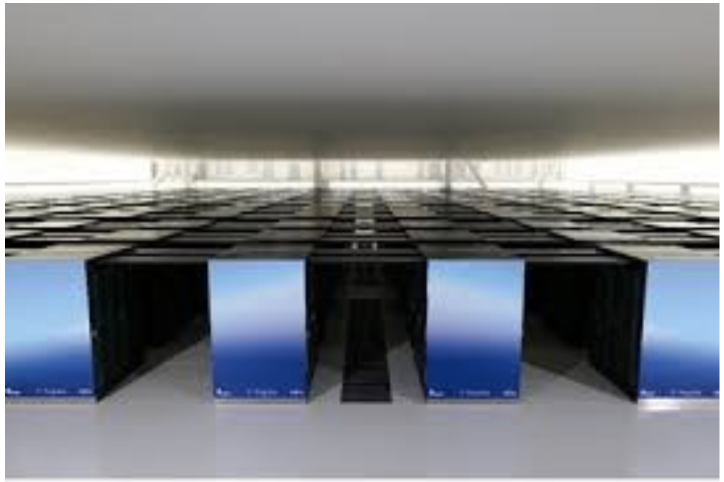
*CPU+Accelerator

Accelerator: GPGPU, NISQ,…

See also Green500 (Flops/W)

Flagship: “Fugaku” Supercomputer

Fugaku 富岳 @RIKEN, Kobe
-158,976 nodes
A64FX
48 cores/node
32 GiB/node



- 1st in Top500 & 9th in Green500 on June 2020
- 1st in Top500 & 10th in Green500 on November 2020

2nd Tier Supercomputers in UTokyo

Ohtaka (CPU server)

ISSP, UTokyo@Kashiwa

Theoretical Peak: 6,881TFlop/s
1,680 nodes
-AMD EPYC 7720 (2.0GHz, 64 cores) x 2
-Memory/node: 258GiB

Oakforest-PACS

JCAHPC@Kashiwa (Utokyo & Tsukuba U)

Theoretical Peak: 24,913.5TFlop/s
8,208 nodes
-Intel Xeon Phi 7250 (1.4GHz, 68 cores)
-Memory/node: 112GiB

Oakbridge-CX (You can use this when you join CSA!)

Intel Xeon
1,368 nodes

How Computer Handles Numbers

Numbers in Computers

Integer

Integer(4): 31bit+1bit

$$-2^{31} \leq j \leq 2^{31} - 1 \ (j \in \mathbb{Z})$$

Real number

IEEE Std 754-2008 binary64
(double-precision floating-point number)

Double precision/real(8):
Sign (1 bit) + Exponent (11 bit) + Significand (52 bit)

$$(-1)^s \times 2^e \times m$$

$$-1022 \leq e \leq 1023 \ (e \in \mathbb{Z}) \qquad m = \frac{\sum_{\ell=0}^{p-1} d_\ell \cdot 2^\ell}{2^p} \qquad p = 53$$

Performance of computer

How many times does the computer multiply/add per second?

How much data does the computer memorize?

How much data does the computer read/write per second?

FLOP/s

Floating-point Operations Per Second

An example: Intel Xeon Phi Nights Landing

Intel Xeon Phi 7250 (1.4GHz, 68 cores, 112GB)

-1.4×10^9 instructions per second

-Intel AVX-512 instruction

double precision floating point number (8byte=64bit)

8 double-precision multiply-add operations

$\rightarrow 1.4 \times 10^9 \times 16 \times 68$ FLOP/s per processor

cf.) Multiply-add: $a \leftarrow a + (b \times c)$

Increasing FLOP/s

Clock rate saturated

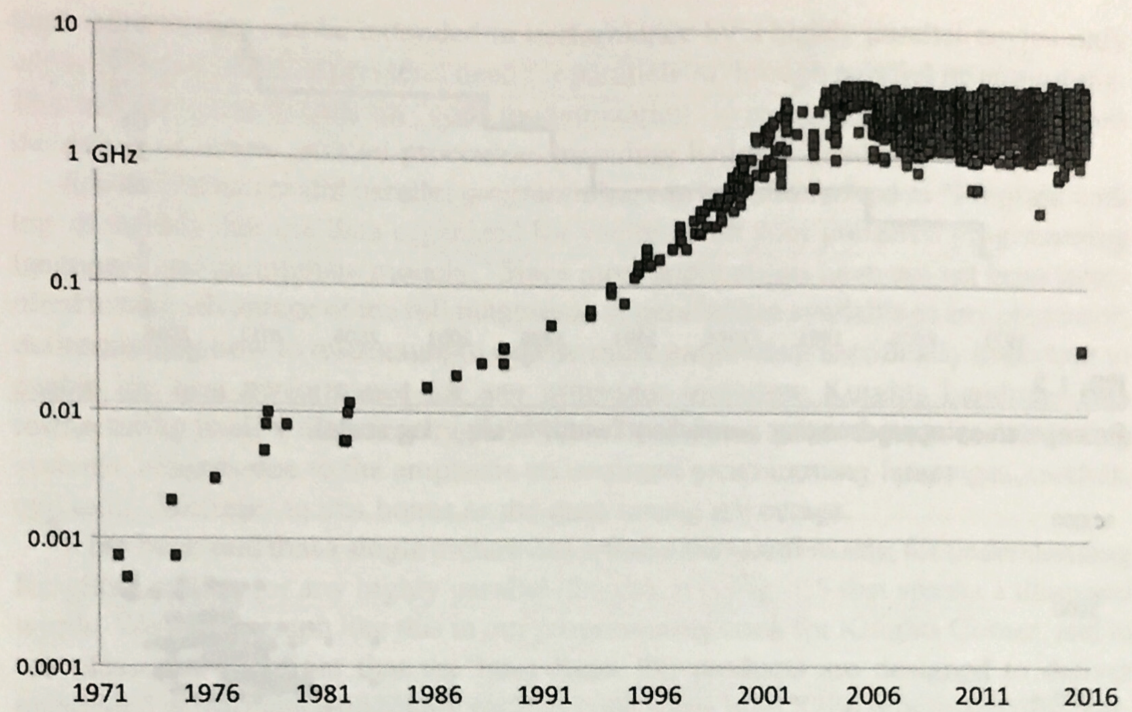


FIG. 1.1

Processor/coprocessor speed era (log scale).

J. Jeffers, J. Reinders, and A. Sodani,
Intel Xeon Phi Processor High Performance Programing

Increasing FLOP/s

Number of cores

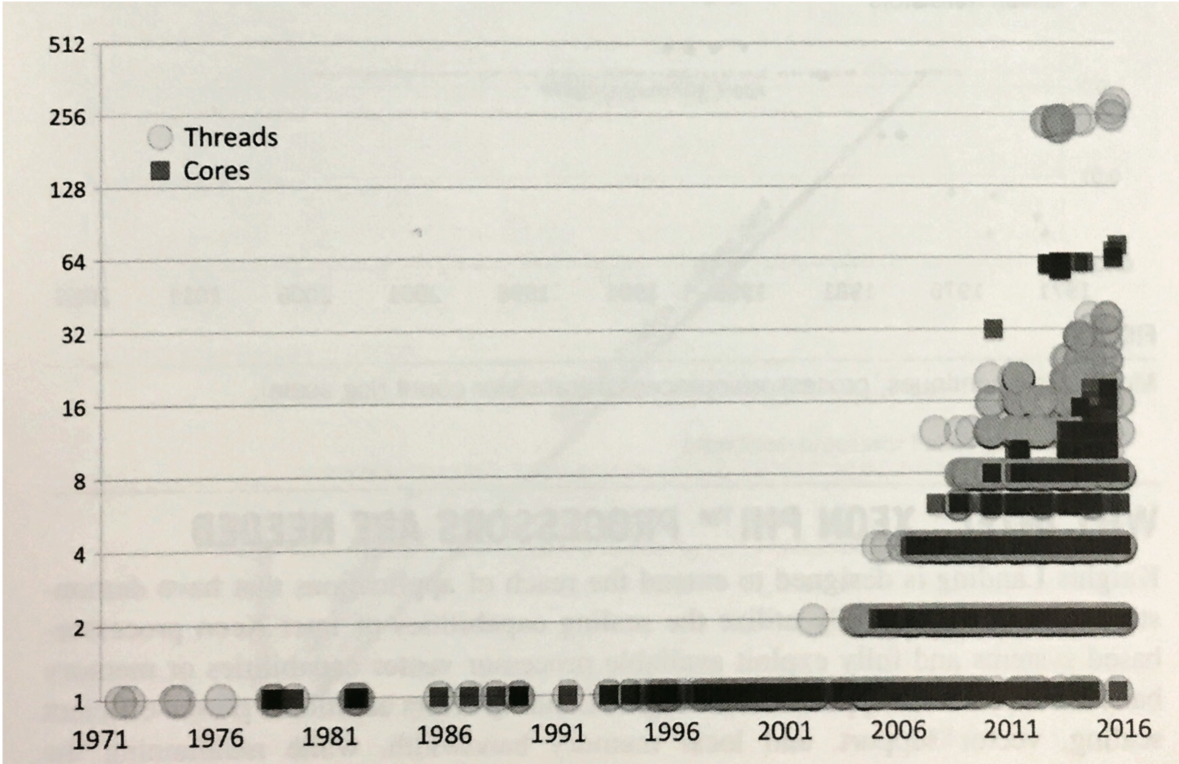


FIG. 1.2

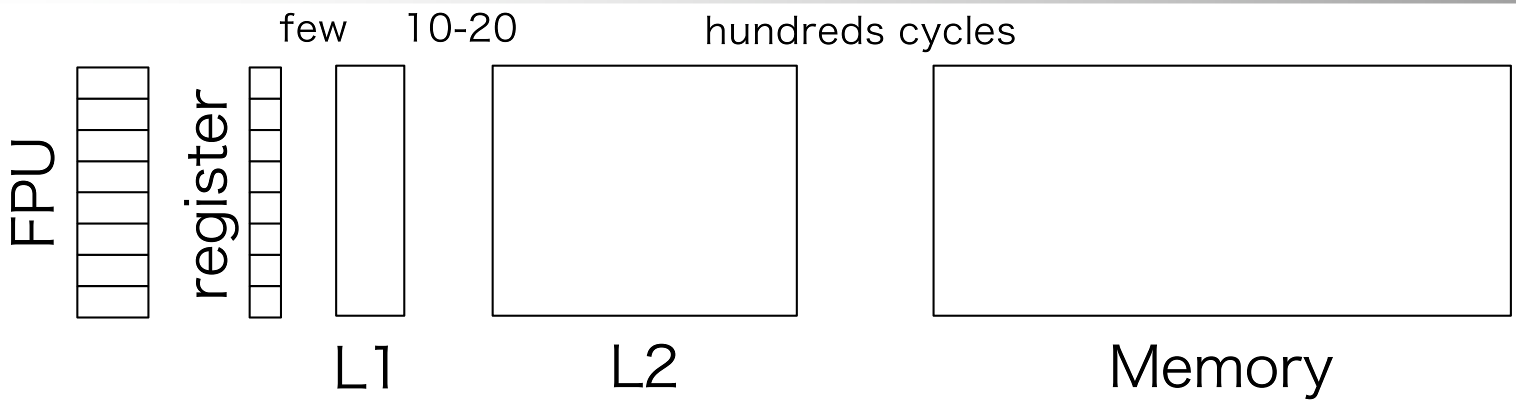
Processor/coprocessor core/thread parallelism (log scale).

J. Jeffers, J. Reinders, and A. Sodani,
Intel Xeon Phi Processor High Performance Programing

Performance of computer

How many times does the computer multiply/add per second?
How much data does the computer memorize?
How much data does the computer read/write per second?

Cache, Memory, and Disk



Hierarchy:
Register
L1 Cache
L2 Cache
Memory
Disk
Network

Example: Intel Xeon Phi 7250

32KB per core
512KB per core
～1.6GB per core (～10GB/s per core)

Byte per flop (B/F)

Exercises (Not report problems)

1. Status of N qubits is represented as a complex vector in 2^N-dimensional Hilbert space. If you can use whole system of Oakforest-PACS, how many qubits can you store in the memory?

2. Usually, B/F of modern supercomputers is less than 1. Which kind of calculations is suitable for them?

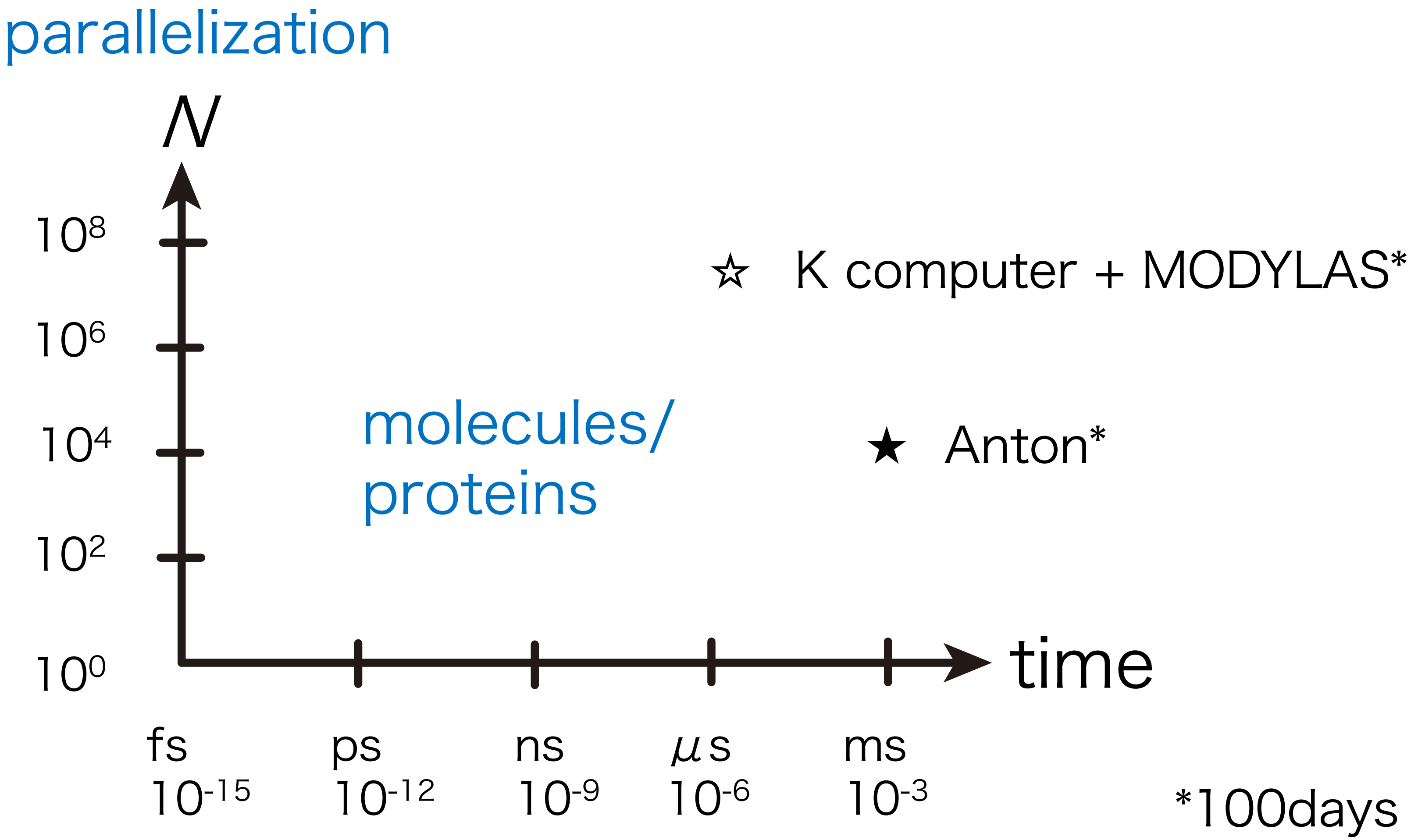
Why Many-Body Problem Is Hard to Solve

Target of this lecture:

1. N-body Newtonian equation of motion
2. N-body classical statistical mechanics
3. N-body Schrödinger equation
4. N-body quantum statistical mechanics

Why Many-Body Problem Is Hard to Solve

1. N-body Newtonian equation of motion
-Time evolution of 6N degrees of freedom



An Example of Many-Body Problems: Proteins

Proteins in water

David E. Shaw *et al*.,
D. E. Shaw Research
SC09 (2009)

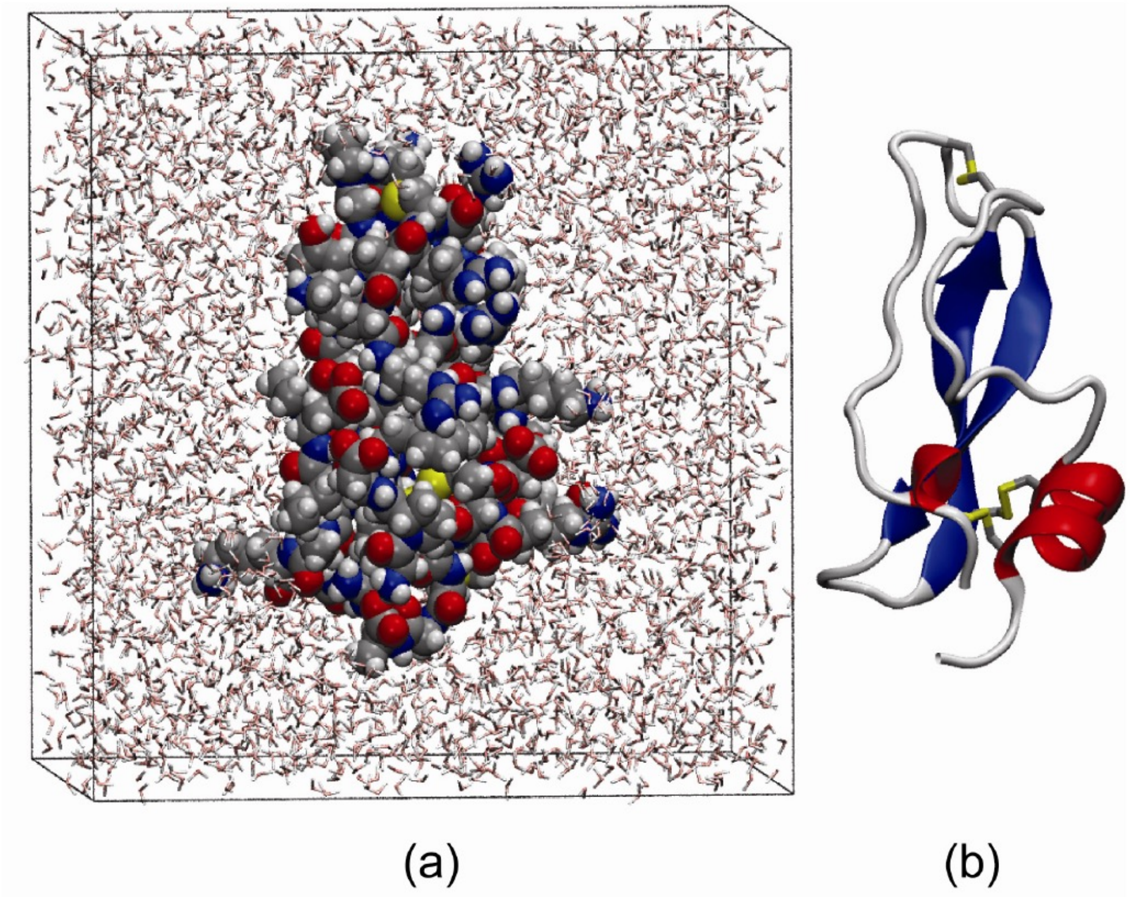


Figure 1: Two renderings of a protein (BPTI) taken from a molecular dynamics simulation on Anton. (a) The entire simulated system, with each atom of the protein represented by a sphere and the surrounding water represented by thin lines. For clarity, water molecules in front of the protein are not pictured. (b) A "cartoon" rendering showing important structural elements of the protein (secondary and tertiary structure).

Why Many-Body Problem Is Hard to Solve

1. N-body Newtonian equation of motion

-Time evolution of $6N$ degrees of freedom:

*Larger Δt is desirable

$\Delta t = 1–2$ fs

For hydrogen atom 0.1fs

*If the system can be divided into independent subsystems, it is easy to treat the system in parallel

However, ions interact each other through long-range Coulomb repulsion

First Principle of Molecular Mechanics: Newtonian/Hamiltonian Mechanics

Brief summary of Hamiltonian mechanics

Hamilton's eqs.

$$\frac{d}{dt}\begin{bmatrix} q \\ p \end{bmatrix} = \begin{bmatrix} +\partial H/\partial p \\ -\partial H/\partial q \end{bmatrix}$$

Operator representation

$$\rightarrow \frac{d}{dt}\begin{bmatrix} q \\ p \end{bmatrix} = \hat{D}_H \begin{bmatrix} q \\ p \end{bmatrix}$$

* Poisson braket

$$\hat{D}_g f = \{f, g\} = \frac{\partial f}{\partial q}\frac{\partial g}{\partial p} - \frac{\partial g}{\partial q}\frac{\partial f}{\partial p}$$

$$\frac{d}{dt} f\left(q(t), p(t)\right) = \{f, H\}\left(q(t), p(t)\right)$$

linear

$$\hat{D}_{g+h} f = \hat{D}_g f + \hat{D}_h f$$

Difference Equation for Hamiltonian Mechanics

Formal solution (Not easy to calulate)

$$\begin{bmatrix} q(t+\Delta t) \\ p(t+\Delta t) \end{bmatrix} = \exp\left[\Delta t \hat{D}_H\right] \begin{bmatrix} q(t) \\ p(t) \end{bmatrix}$$

Forward difference

$$\begin{aligned} \begin{bmatrix} q(t+\Delta t) \\ p(t+\Delta t) \end{bmatrix} &= \exp\left[\Delta t \hat{D}_H\right] \begin{bmatrix} q(t) \\ p(t) \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} q(t) \\ p(t) \end{bmatrix} + \Delta t \hat{D}_H \begin{bmatrix} q(t) \\ p(t) \end{bmatrix}}_{\text{Euler method}} + \frac{(\Delta t)^2}{2!} \hat{D}_H^2 \begin{bmatrix} q(t) \\ p(t) \end{bmatrix} + \mathcal{O}(\Delta t^3) \end{aligned}$$

Do not use Euler!

Difference Equation for Hamiltonian Mechanics

Another formulation of forward difference

Hamiltonian split into kinetic energy T and potential V

$$H = T(p) + V(q)$$

$$\exp\left[\Delta t \hat{D}_H\right] \simeq \exp\left[\frac{\Delta t}{2}\hat{D}_T\right]\exp\left[\Delta t \hat{D}_V\right]\exp\left[\frac{\Delta t}{2}\hat{D}_T\right]$$

$$\exp\left[\Delta t \hat{D}_T\right]\begin{bmatrix} q(t) \\ p(t) \end{bmatrix} = \begin{bmatrix} q(t) + \Delta t \left.\frac{\partial T}{\partial p}\right|_{p=p(t)} \\ p(t) \end{bmatrix}$$

$$\exp\left[\Delta t \hat{D}_V\right]\begin{bmatrix} q(t) \\ p(t) \end{bmatrix} = \begin{bmatrix} q(t) \\ p(t) - \Delta t \left.\frac{\partial V}{\partial q}\right|_{q=q(t)} \end{bmatrix}$$

No series expansion !

Why Many-Body Problem Is Hard to Solve

2. *N*-body classical statistical mechanics

Example: 1 D Ising Model

$$H = J \sum_{i=0}^{L-1} \sigma_i \sigma_{i+1} - B \sum_{i=0}^{L-1} \sigma_i$$

Ising spin: $\sigma_i = \pm 1$

Periodic boundary: $i + 1 \to \mathrm{mod}(i + 1, L)$

Ising Model 1

History of the Lenz-Ising Model, S. G. Brush, Rev. Mod. Phys. 39, 883 (1967)

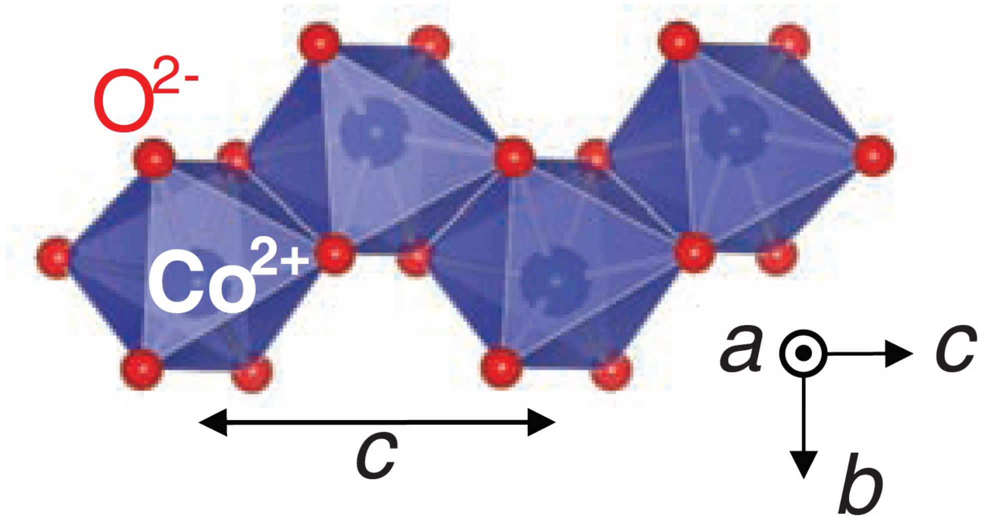
Many physico-chemical systems can be represented more or less accurately by a lattice arrangement of molecules with nearest-neighbor interactions.
The simplest and most popular version of this theory is the so-called "Ising model," discussed by Ernst Ising in 1925 but suggested earlier (1920) by Wilhelm Lenz.

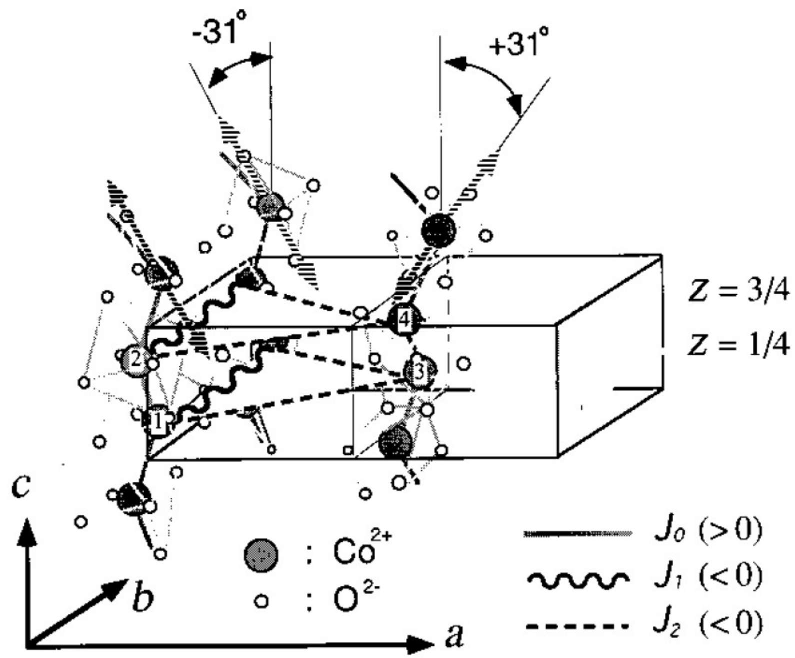
After many years of being scorned or ignored by most scientists, the so-called "Ising model" has recently enjoyed increased popularity and may, if present trends continue, take its place as the preferred basic theory of all cooperative phenomena.

Ising Model 2

A realization of Ising models

Quasi-1D Ising Ferromagnet: $CoNb_2O_6$





S. Kobayashi, *et al.*,
Phys. Rev. B 60, 3331 (1999)

Recent progress, *i.e.*,
R. Coldea, *et al.*,
Science 327, 177 (2010)

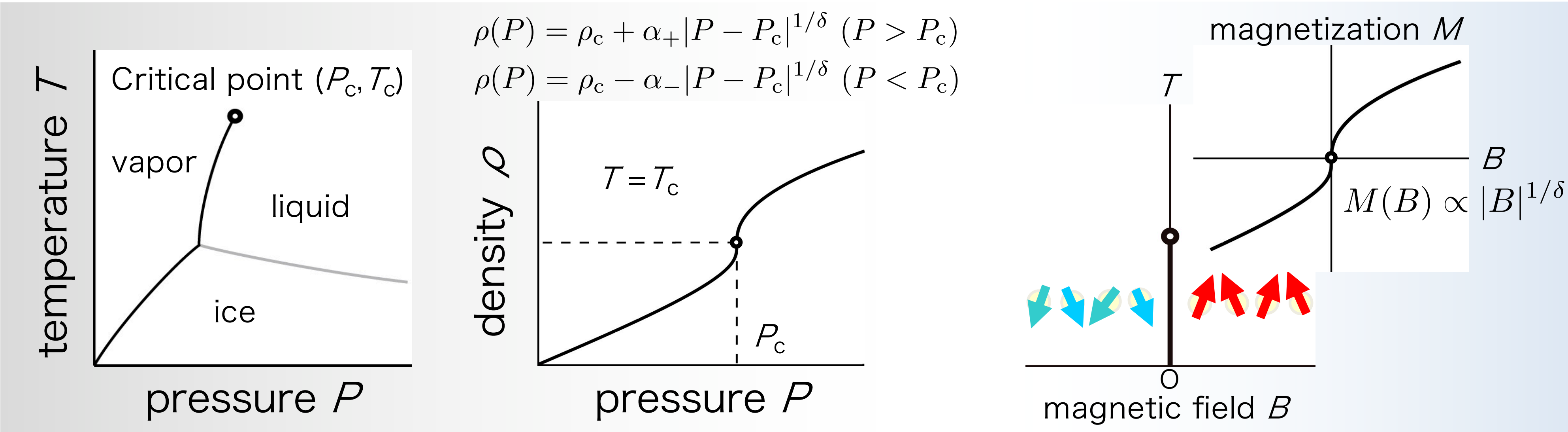
Ising Model 3

The archetype of critical phenomena

Critical phenomena of liquid/vapor water is characterized by 3D Ising universality although *prediction* of T_c and P_c requires QM/MM

Water

3D Ising



Why Many-Body Problem Is Hard to Solve

2. *N*-body classical statistical mechanics

Example: 1 D Ising Model

$$H = J \sum_{i=0}^{L-1} \sigma_i \sigma_{i+1} - B \sum_{i=0}^{L-1} \sigma_i$$

Ising spin: $\sigma_i = \pm 1$

Periodic boundary: $i+1 \to \mathrm{mod}(i+1, L)$

Partition function: Summation over 2^L configurations

$$Z = \sum_{\sigma_0=\pm 1} \sum_{\sigma_1=\pm 1} \cdots \sum_{\sigma_{L-1}=\pm 1} \exp\left(-H[\{\sigma\}]/k_{\mathrm{B}}T\right)$$

$$\{\sigma\} = \{\sigma_0, \sigma_1, \ldots, \sigma_{L-1}\}$$

Why Many-Body Problem Is Hard to Solve

3&4. *N*-body Schrödinger equation & quantum statistical mechanics

Example: 1 D Transverse Field Ising Model (TFIM)

$$\hat{H} = J\sum_{i=0}^{L-1}\hat{S}_i^z\hat{S}_{i+1}^z - \Gamma\sum_{i=0}^{L-1}\hat{S}_i^x$$

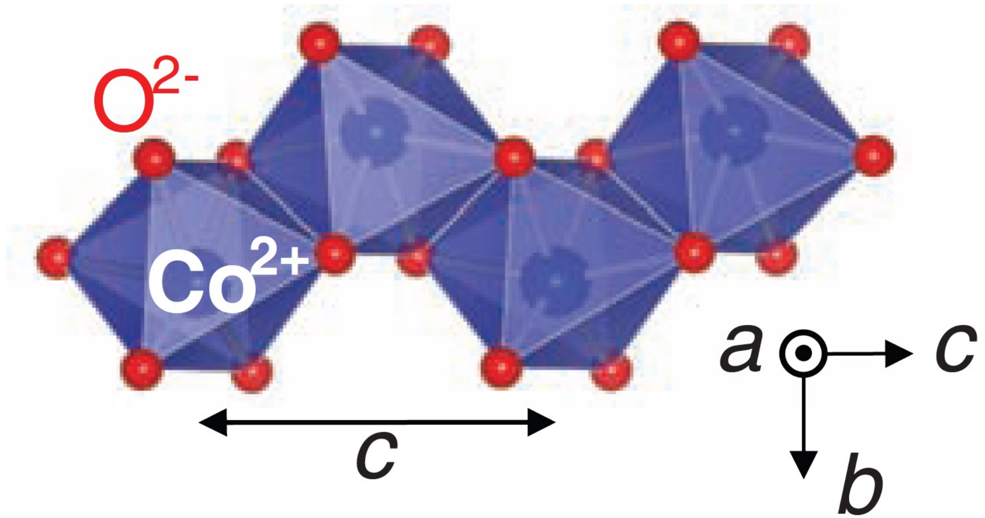
Quantum Spin 1/2 or Qubit $\quad |\uparrow\rangle,\ |\downarrow\rangle \quad |1\rangle,\ |0\rangle$

Periodic boundary: $i+1 \to \mathrm{mod}(i+1, L)$

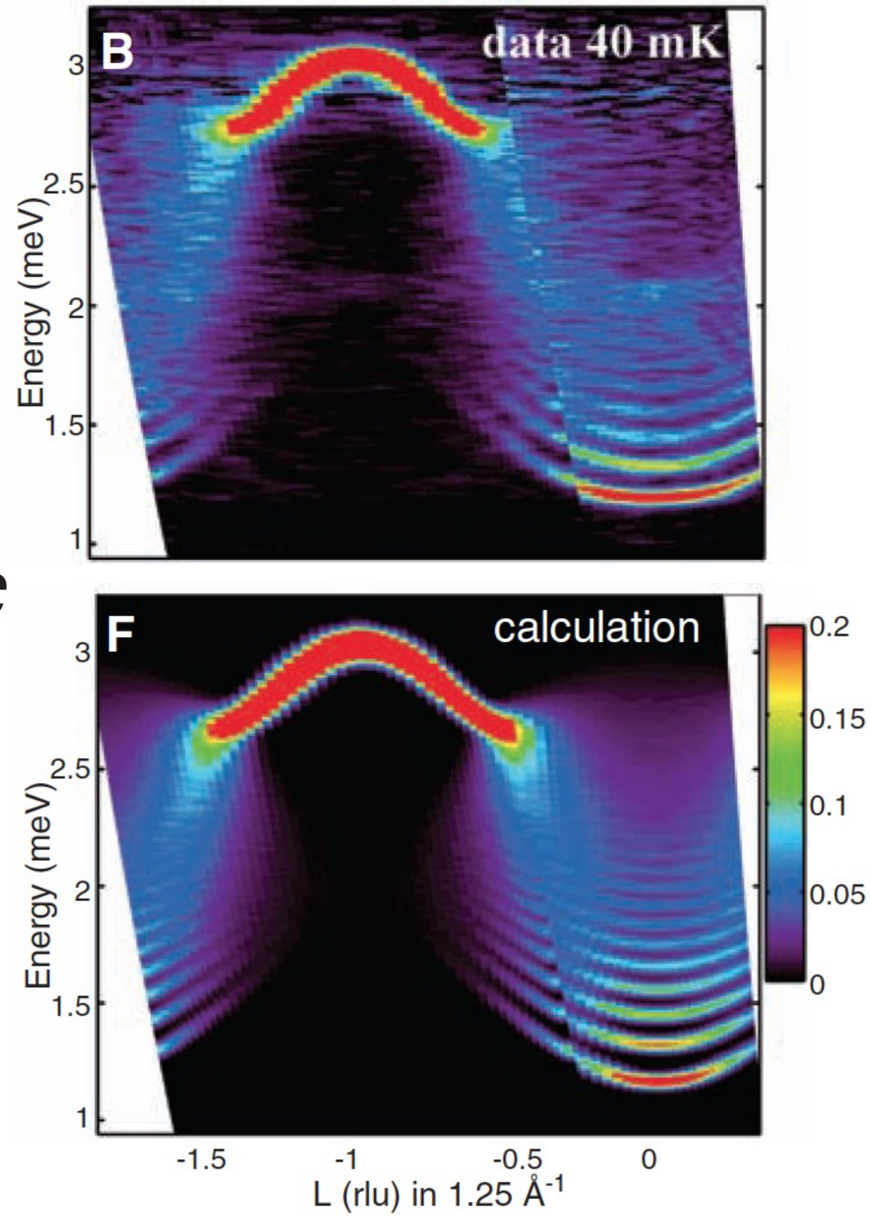
Transverse Field Ising Model

A realization of TFIM:

$CoNb_2O_6$



R. Coldea, *et al.*,
Science 327, 177 (2010)



Quantum Spin S=1/2 or Qubit

Operators acting on a single qubit

$$|1\rangle = |\uparrow\rangle, \;\; |0\rangle = |\downarrow\rangle$$

A two dimensional representation of Lie algebra SU(2)

$$[\hat{S}_j^x, \hat{S}_j^y] = i\hat{S}_j^z$$
$$[\hat{S}_j^y, \hat{S}_j^z] = i\hat{S}_j^x$$
$$[\hat{S}_j^z, \hat{S}_j^x] = i\hat{S}_j^y$$
$$[\hat{A}, \hat{B}] \equiv \hat{A}\hat{B} - \hat{B}\hat{A}$$

$$\hat{S}_j^x|1\rangle = +\frac{1}{2}|0\rangle$$
$$\hat{S}_j^x|0\rangle = +\frac{1}{2}|1\rangle$$
$$\hat{S}_j^y|1\rangle = +i\frac{1}{2}|0\rangle$$
$$\hat{S}_j^y|0\rangle = -i\frac{1}{2}|1\rangle$$
$$\hat{S}_j^z|1\rangle = +\frac{1}{2}|1\rangle$$
$$\hat{S}_j^z|0\rangle = -\frac{1}{2}|0\rangle$$

Quantum Spin S=1/2 or Qubit

$$|\phi\rangle = c_\uparrow|1\rangle + c_\downarrow|0\rangle$$

$$\hat{S}_j^\alpha|\phi\rangle = c'_\uparrow|1\rangle + c'_\downarrow|0\rangle$$

$$\left(\begin{array}{c} c'_\uparrow \\ c'_\downarrow \end{array}\right) = \frac{1}{2}\hat{\sigma}^\alpha \left(\begin{array}{c} c_\uparrow \\ c_\downarrow \end{array}\right)$$

$$\hat{\sigma}^x = \left(\begin{array}{cc} 0 & +1 \\ +1 & 0 \end{array}\right)$$

$$\hat{\sigma}^y = \left(\begin{array}{cc} 0 & -i \\ +i & 0 \end{array}\right)$$

$$\hat{\sigma}^z = \left(\begin{array}{cc} +1 & 0 \\ 0 & -1 \end{array}\right)$$

$$\hat{S}_j^\alpha \doteq \frac{1}{2}\hat{\sigma}^\alpha$$

Quantum Spins: Two Site TFIM

Decimal representation of orthonormalized basis

		0 th site		1 st site
$\lvert 0\rangle_{\mathrm{d}}$	$=$	$\lvert\downarrow\rangle$	$\otimes$	$\lvert\downarrow\rangle$
$\lvert 1\rangle_{\mathrm{d}}$	$=$	$\lvert\uparrow\rangle$	$\otimes$	$\lvert\downarrow\rangle$
$\lvert 2\rangle_{\mathrm{d}}$	$=$	$\lvert\downarrow\rangle$	$\otimes$	$\lvert\uparrow\rangle$
$\lvert 3\rangle_{\mathrm{d}}$	$=$	$\lvert\uparrow\rangle$	$\otimes$	$\lvert\uparrow\rangle$

$$L = 2 \qquad \hat{H} = J\sum_{i=0}^{L-1}\hat{S}_i^z\hat{S}_{i+1}^z - \Gamma\sum_{i=0}^{L-1}\hat{S}_i^x$$

$$\hat{H} \doteq \begin{pmatrix} +J/2 & -\Gamma/2 & -\Gamma/2 & 0 \\ -\Gamma/2 & -J/2 & 0 & -\Gamma/2 \\ -\Gamma/2 & 0 & -J/2 & -\Gamma/2 \\ 0 & -\Gamma/2 & -\Gamma/2 & +J/2 \end{pmatrix} \quad {}_{\mathrm{d}}\langle i|\hat{H}|j\rangle_{\mathrm{d}}$$

Larger TFIM

$$\hat{H} = J \sum_{i=0}^{L-1} \hat{S}_i^z \hat{S}_{i+1}^z - \Gamma \sum_{i=0}^{L-1} \hat{S}_i^x$$

-Non-commutative

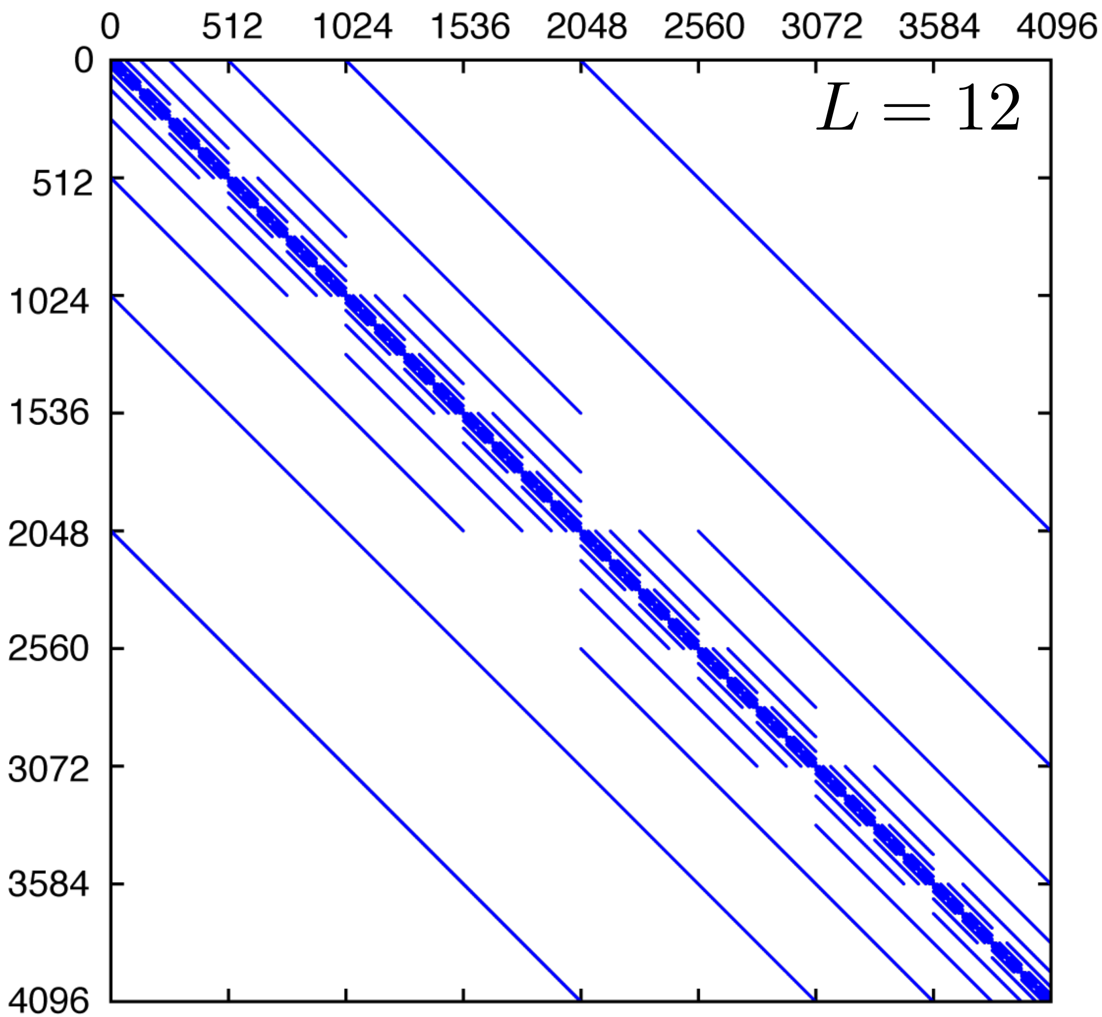
$$\left[\sum_{i=0}^{L-1} \hat{S}_i^z \hat{S}_{i+1}^z, \sum_{i=0}^{L-1} \hat{S}_i^x \right] \neq 0$$

→Quantum fluctuations
or Zero point motion

-Sparse
# of elements ∝ $O(2^L)$

-Solvable

-Hierarchical matrix?



Eigenvalue Problems

Diagonalizing Hermitian matrices

Standard approach:
Call LAPACK (Linear Algebra PACKage)

subroutine zheev:
Householder reflection + QR algorithm
$\rightarrow O(N^3)$ numerical cost

Computational and Memory Costs

1. Vector-vector product $$\sum_{j=0}^{N_{\mathrm{H}}-1} u_j^* v_j$$

   Computational: $\mathcal{O}(N_{\mathrm{H}})$
   Memory: $\mathcal{O}(N_{\mathrm{H}})$

2. Matrix-vector product $$v_i = \sum_{j=0}^{N_{\mathrm{H}}-1} A_{ij} u_j$$

   Computational: $\mathcal{O}(N_{\mathrm{H}}^2)$
   Memory: $\mathcal{O}(N_{\mathrm{H}}^2)$

3. Matrix-matrix product $$C_{ij} = \sum_{k=0}^{N_{\mathrm{H}}-1} A_{ik} B_{kj}$$

   Computational: $\mathcal{O}(N_{\mathrm{H}}^3)$
   Memory: $\mathcal{O}(N_{\mathrm{H}}^2)$

Eigenvalue Problems

1. Tridiagonalization
   by Householder reflection

$$\hat{P}_j = \hat{1} - 2\frac{\vec{v}_j \vec{v}_j^\dagger}{\|\vec{v}_j\|^2}$$

$$\hat{H}_{\mathrm{td}} = \hat{P}_{N_{\mathrm{H}}-3} \cdots \hat{P}_1 \hat{P}_0 \hat{H} \hat{P}_0 \hat{P}_1 \cdots \hat{P}_{N_{\mathrm{H}}-3} \qquad \mathcal{O}(N_{\mathrm{H}}^3)$$

$N_{\mathrm{H}} = 8$

$$\hat{H}_{\mathrm{td}} = \begin{pmatrix} \alpha_0 & \beta_0 & & & & & & \\ \beta_0 & \alpha_1 & \beta_1 & & & & & \\ & \beta_1 & \alpha_1 & \beta_2 & & & & \\ & & \beta_2 & \alpha_3 & \beta_3 & & & \\ & & & \beta_3 & \alpha_4 & \beta_4 & & \\ & & & & \beta_4 & \alpha_5 & \beta_5 & \\ & & & & & \beta_5 & \alpha_6 & \beta_6 \\ & & & & & & \beta_6 & \alpha_7 \end{pmatrix}$$

Eigenvalue Problems

2. QR algorithm

$$\hat{A}_0 = \hat{H}_{\mathrm{td}}$$

$$\hat{A}_k = \hat{Q}_k \hat{R}_k \qquad \mathcal{O}(N_{\mathrm{H}})$$

$$\hat{A}_{k+1} = \hat{R}_k \hat{Q}_k \quad \mathcal{O}(N_{\mathrm{H}}^3)$$

$$\hat{U} = \lim_{k \to +\infty} \hat{A}_k$$

$\rightarrow$ Diagonal elements are eigenvalues of *H*

Upper triangle matrix

$$N_{\mathrm{H}} = 8$$

$$\hat{R} = \begin{pmatrix} a_0 & b_0 & c_0 & d_0 & e_0 & f_0 & g_0 & h_0 \\ & a_1 & b_1 & c_1 & d_1 & e_1 & f_1 & g_1 \\ & & a_1 & b_2 & c_2 & d_2 & e_2 & f_2 \\ & & & a_3 & b_3 & c_3 & d_3 & e_3 \\ & & & & a_4 & b_4 & c_4 & d_4 \\ & & & & & a_5 & b_5 & c_5 \\ & & & & & & a_6 & b_6 \\ & & & & & & & a_7 \end{pmatrix}$$

Difficulties in Many-Body Problems

-Summation over exponetially large
 # of configurations

←Monte Carlo (3rd to 9th lecture), ⋯

-$O(N^3)$ numerical cost and $O(N^2)$ memory

←Krylov subspace method (10th to 12th), ⋯

Lecture Schedule

Classical

Quantum

#1 Many-body problems in physics
#2 Why many-body problem is hard to solve
#3 Classical statistical model and numerical simulation
#4 Classical Monte Carlo method and its applications
#5 Molecular dynamics and its application
#6 Extended ensemble method for Monte Carlo methods
#7 Quantum lattice models and numerical approaches
#8 Quantum Monte Carlo methods
#9 Applications of quantum Monte Carlo methods
#10 Linear algebra of large and sparse matrices for quantum many-body problems
#11 Krylov subspace methods and their applications to quantum many-body problems
#12 Large sparse matrices and quantum statistical mechanics
#13 Parallelization for many-body problems

Answer of Exercise

1. Status of N qubits is represented as a complex vector in 2^N-dimensional Hilbert space. If you can use whole system of Oakforest-PACS, how many qubits can you store in the memory?

   $112 \times 10^9 \times 8208 \sim 1.6 \times 16 \times 2^{45}$

   A. 45 qubits

2. Usually, B/F of modern supercomputers is less than 1. Which kind of calculations is suitable for them?

   A. Matrix-matrix product

   Computational: $\mathcal{O}(N_{\mathrm{H}}^3)$
   Memory: $\mathcal{O}(N_{\mathrm{H}}^2)$