Tensor Network Theory

An introduction to DMRG and MPS methods

Quantum and Kinetic Problems: Modeling, Analysis, Numerics and Applications
IMS Singapore

Dieter Jaksch

University of Oxford

Special thanks to: Stephen Clark, University of Bristol

Outline of lectures

- Lecture 1

Introduction to strong correlations, many-body problem, recap on essential linear algebra we will need later.

- Lecture 2

Tensors and contractions, product states and variational principle, matrix product states (MPS), and their entanglement properties.

- Lecture 3

The “calculus” of MPS, algorithms for their variational optimisation, and algorithms for time-evolution of MPS.

- Lecture 4

Moving to finite temperatures, renormalisation approaches to tensor networks, extension to 2D with projected entangled pairs.

Lecture 1

Key references

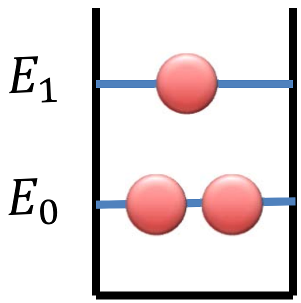
- **U. Schollwock,**
  The density-matrix renormalization group in the age of matrix product states, Annals of Physics **326**, 96 (2011).
- **F. Verstraete, V. Murg and J.I. Cirac,**
  Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems, Advances in Physics **57**, 143 (2008).
- **J.I. Cirac and F. Verstraete,**
  Renormalization and tensor product states in spin chains and lattices, J. Phys. A: Math. Theor. **42**, 504004 (2009).
- **G. Evenbly and G. Vidal,**
  Quantum criticality with the multi-scale entanglement renormalization ansatz, arxiv:1109.5334
- **Roman Orus,**
  A *practical introduction to tensor networks: matrix product states and projected entangled pair states,* arxiv:1306.2164

1: Strong correlations

Two particles in two boxes

$$a|n\rangle = \sqrt{n}|n-1\rangle$$
$$a^\dagger|n\rangle = \sqrt{n+1}|n+1\rangle$$

- Potential energy

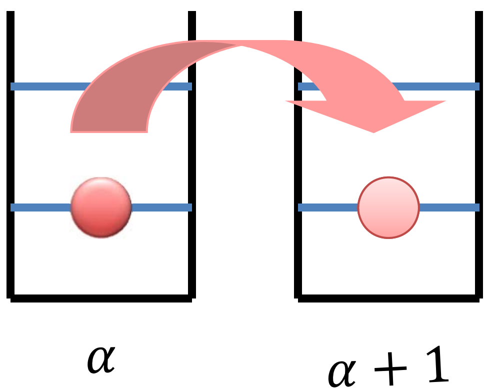


$$E_p = 2\,E_0 + E_1$$

$$H_p = E_0\, a_0^\dagger a_0 + E_1 a_1^\dagger a_1$$

Hamiltonian counts number of particles in each state

- Kinetic energy

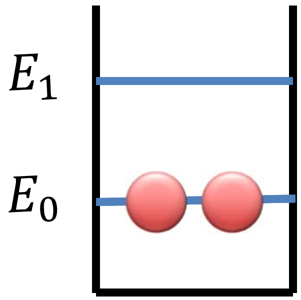


$$H_k = -J\, a_{\alpha+1}^\dagger a_\alpha + \text{h.c.}$$

A particle gains energy by hopping between different states

Two particles in two boxes

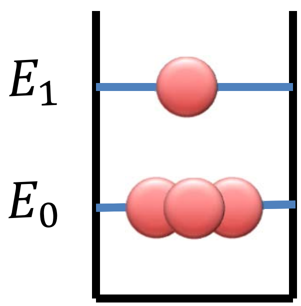
- Interaction energy
  - n particles in the same state



$$H_{int} = \frac{U}{2} a_0^{\dagger} a_0 (a_0^{\dagger} a_0 - 1)$$

Each particle interacts with n-1 particles in the same state

  - Interactions between particles in different states



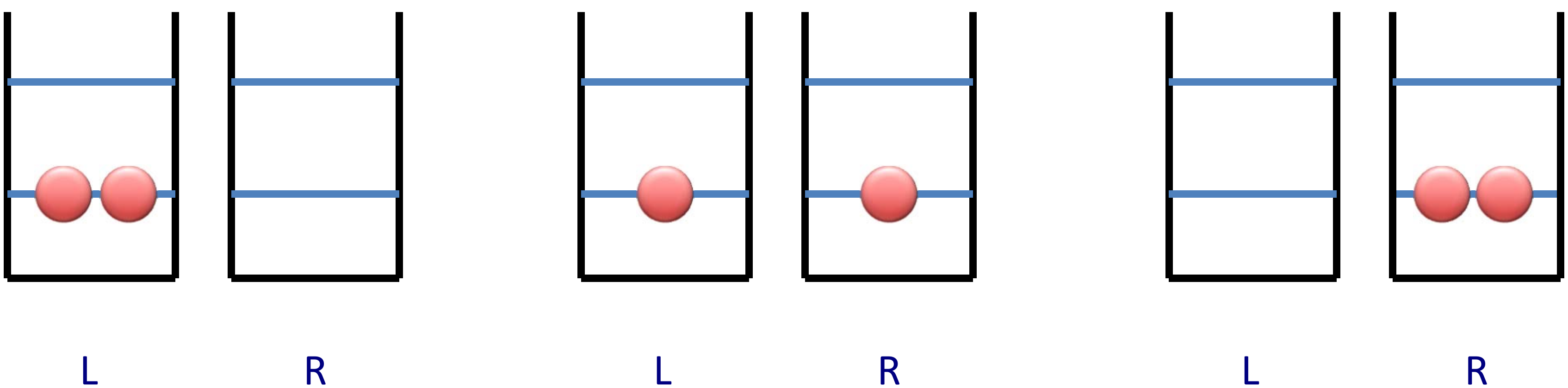
$$H_{int} = U\, a_0^{\dagger} a_0 a_1^{\dagger} a_1$$

Each particle interacts with all particles in the other state

Two particles in two boxes

$$H_{\mathrm{BH}} = -J\left(a_R^\dagger a_L + a_L^\dagger a_R\right) + \frac{U}{2}\sum_{i=L,R}\left[a_i^\dagger a_i\left(a_i^\dagger a_i - 1\right)\right]$$

Basis states



$|20\rangle$ $|11\rangle$ $|02\rangle$

Hamiltonian and ground state

- Matrix form

$$\begin{matrix} & \langle 20| & \langle 11| & \langle 02| & \\ H_{\mathrm{BH}} = & \left(\begin{matrix} U \\ -J\sqrt{2} \\ 0 \end{matrix}\right. & \begin{matrix} -J\sqrt{2} \\ 0 \\ -J\sqrt{2} \end{matrix} & \left.\begin{matrix} 0 \\ -J\sqrt{2} \\ U \end{matrix}\right) & \begin{matrix} |20\rangle \\ |11\rangle \\ |02\rangle \end{matrix} \end{matrix}$$

$$H_{\mathrm{BH}} = U(|20\rangle\langle 20| + |02\rangle\langle 02|) - \sqrt{2}\mathrm{J}(|20\rangle\langle 11| + |02\rangle\langle 11| + \mathrm{h.c.})$$

- Ground state

$$|\Psi\rangle \propto \begin{pmatrix} 2 \\ 4\sqrt{1+(U/4J)^2} + U/J \\ 2 \end{pmatrix} \begin{matrix} |20\rangle \\ |11\rangle \\ |02\rangle \end{matrix}$$

$$|\Psi\rangle \propto 2|20\rangle + \left(4\sqrt{1+\left(\frac{U}{4J}\right)^2} + \frac{U}{J}\right)|11\rangle + 2|02\rangle$$

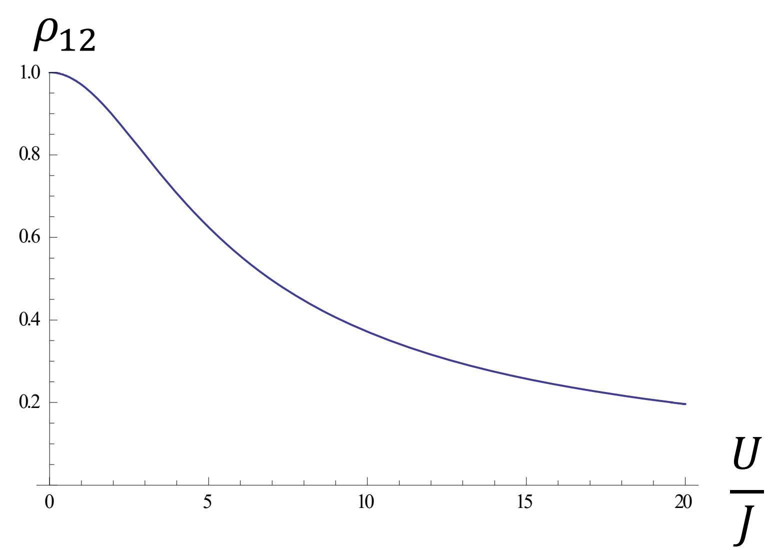
Single particle density matrix

- This is defined by

$$\rho_1 = \begin{pmatrix} \langle a_L^\dagger \, a_L \rangle & \langle a_R^\dagger \, a_L \rangle \\ \langle a_L^\dagger \, a_R \rangle & \langle a_R^\dagger \, a_R \rangle \end{pmatrix}$$

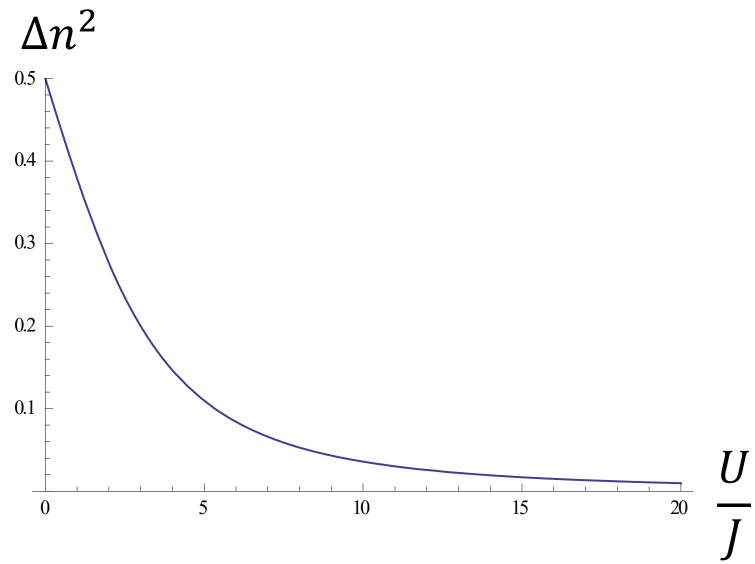
- For the ground state

$$\rho_1 = \begin{pmatrix} 1 & \frac{1}{\sqrt{1+(U/4J)^2}} \\ \frac{1}{\sqrt{1+(U/4J)^2}} & 1 \end{pmatrix}$$



Particle number fluctuations

- How well defined is the particle number in each site?



- This is given by $\Delta n^2 = \langle n_L^2 \rangle - \langle n_L \rangle^2$

$$\Delta n^2 = \frac{1}{2}\left(1 - \frac{1}{\sqrt{1+(4J/U)^2}}\right)$$

Compare: Mean-field theory

- Assume that sites are not correlated

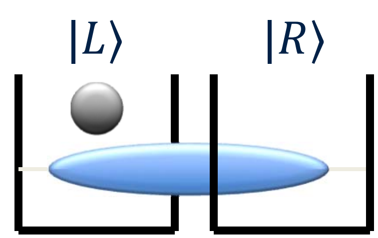
$$|\Psi\rangle = (c_L|L\rangle_1 + c_R|R\rangle_1) \otimes (c_L|L\rangle_2 + c_R|R\rangle_2)$$

$$|\Psi\rangle = c_L^2|LL\rangle + c_L c_R(|LR\rangle + |RL\rangle) + c_R^2|RR\rangle$$

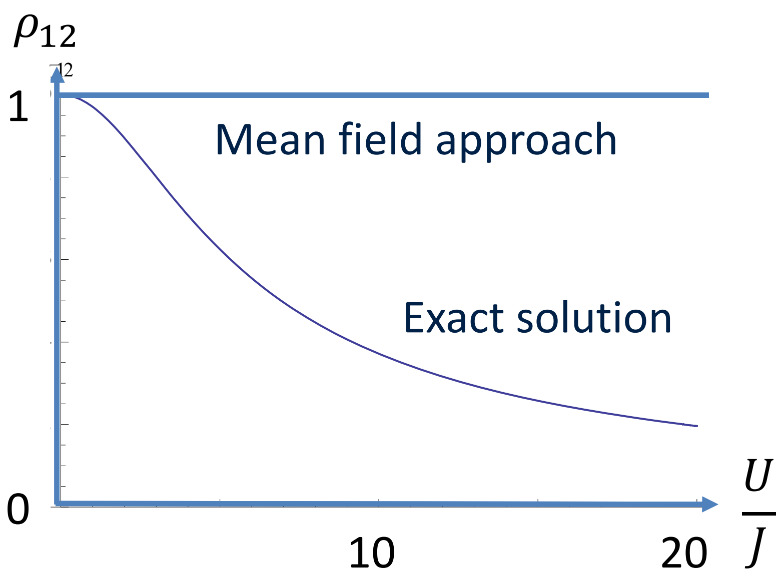
- With this ansatz the Schrödinger equation (obtained from the principle of least action) becomes non-linear in the remaining coefficients ($U > 0$)

$$H = \begin{pmatrix} |c_L|^2 U & -J \\ -J & |c_R|^2 U \end{pmatrix} \qquad |\Psi_{MF}\rangle = \begin{pmatrix} c_L \\ c_R \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

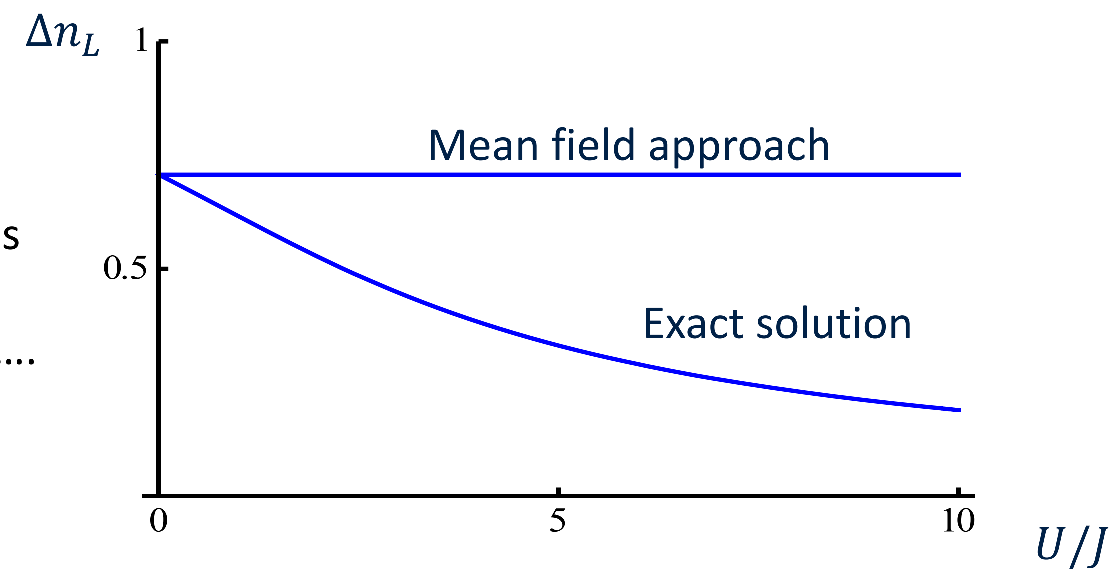
- The resulting approximate ground state is not correlated and describes one particle moving in a mean background field created by the other particle
- Dimension of vectors increases linearly with size.
- Reduction of degrees of freedom is achieved by ignoring correlations and introducing non-linearity.
- Often valid for sufficiently weak interactions



Compare: Mean-field theory



Coherences are not describable by mean field theory – they cannot be distance dependent



Particle number fluctuations are not captured by mean-field theory, no squeezing, ….

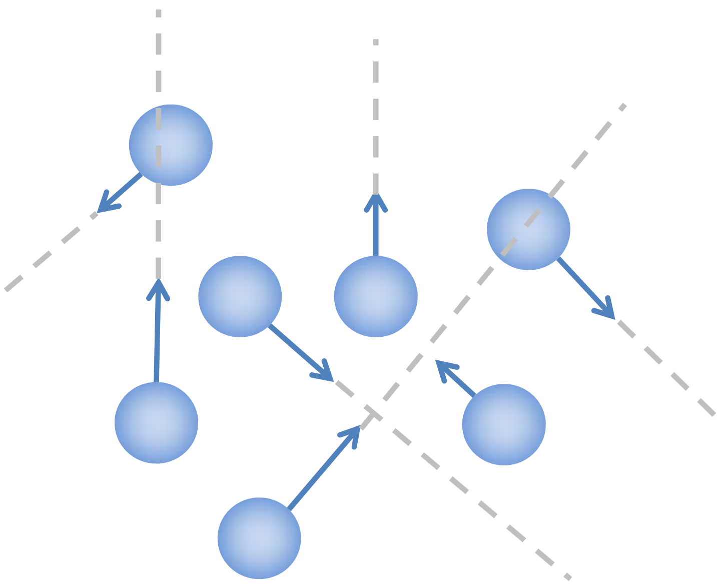
Strong correlations?

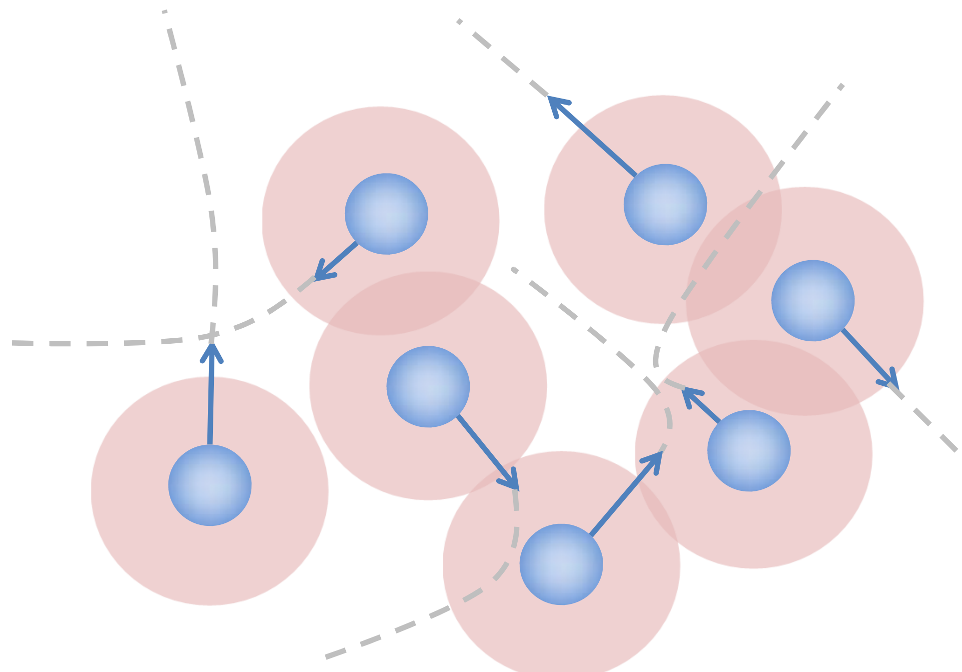
Tensor Network Theory (TNT) is a powerful toolbox for dealing with strong correlations. So what are *strong correlations* exactly?

"correlations" is a synonym for "interactions"

weak interactions

- can extrapolate *en masse* behaviour from one particle



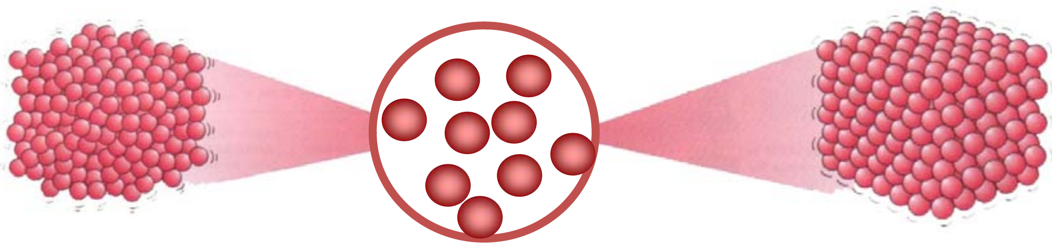


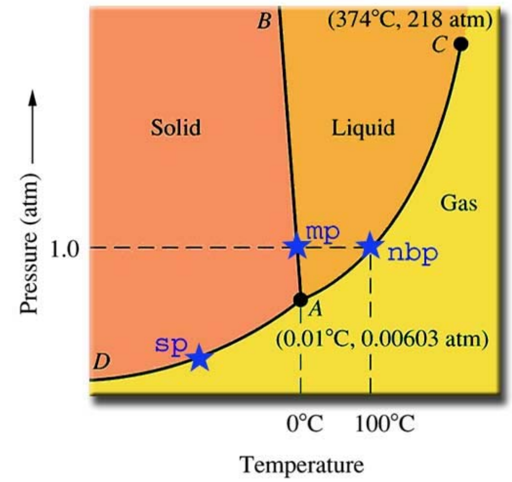
strong interactions

- particles do not move independently

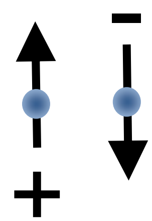
Dramatic consequences

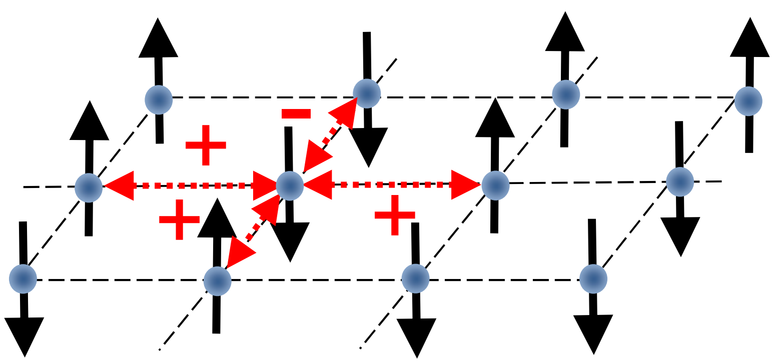
Interplay of microscopic interactions with external influences can lead to abrupt macroscopic changes …

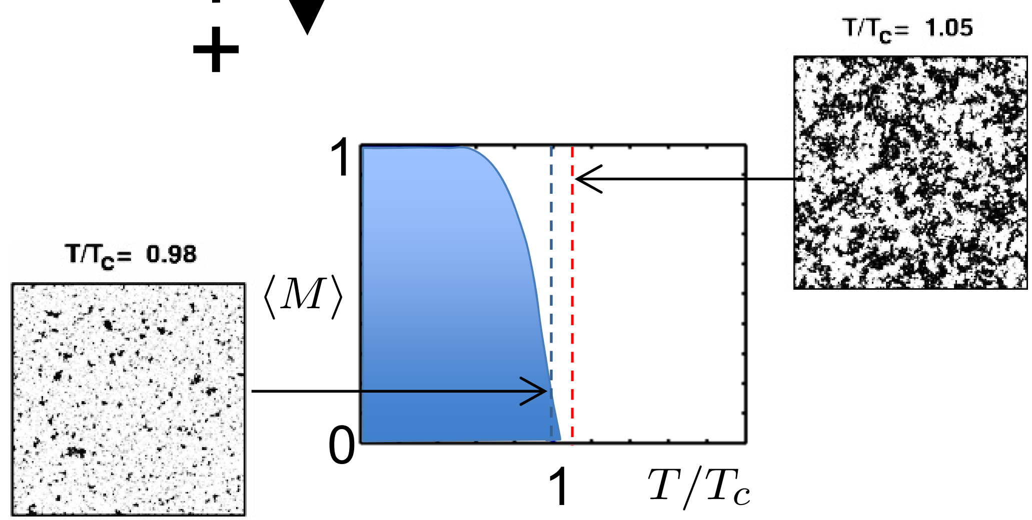




Simple “classical” Ising magnet:







Strong correlations

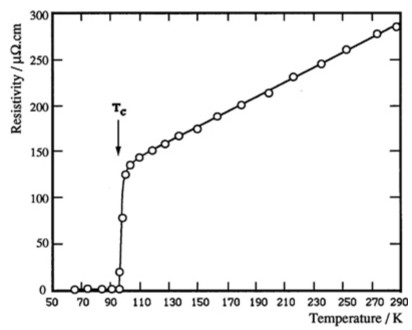
Beyond physics strong correlations appear pervasive …

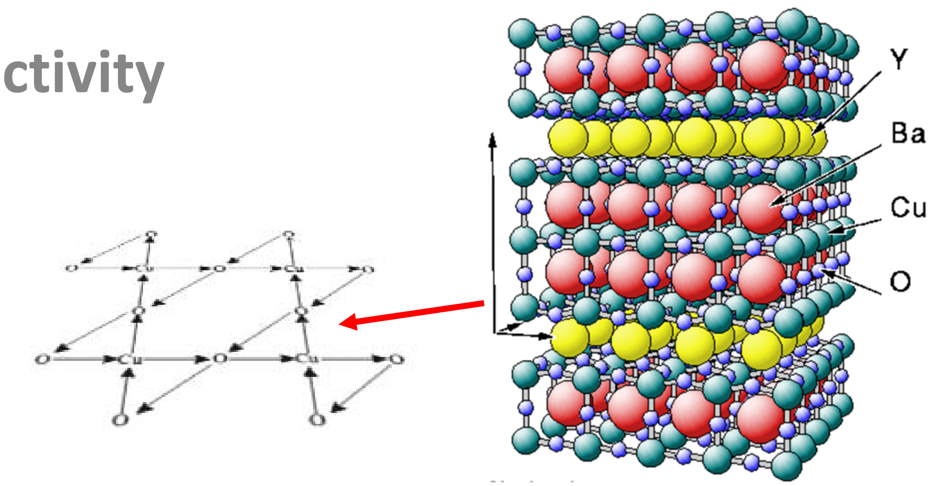
Add quantum mechanics …

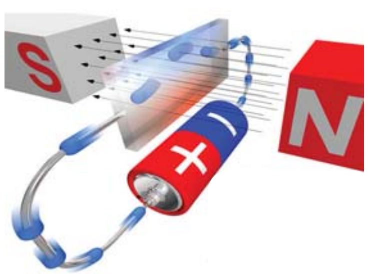
Major interest in *quantum* many-body problems arises in lattice systems – trying to understand the remarkable properties of electrons in some materials …

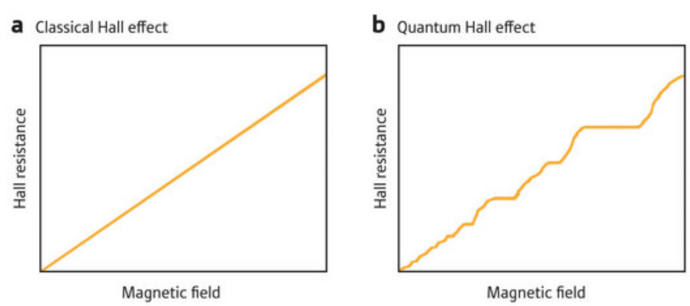
- **High-temperature superconductivity**

What is the pairing mechanism?









- **Quantum Hall effect**

What are the topological properties of fractional QH states?

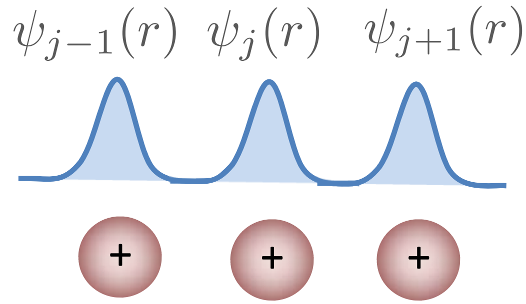
These are seminal strongly-correlated phenomena.

2: The many-body problem

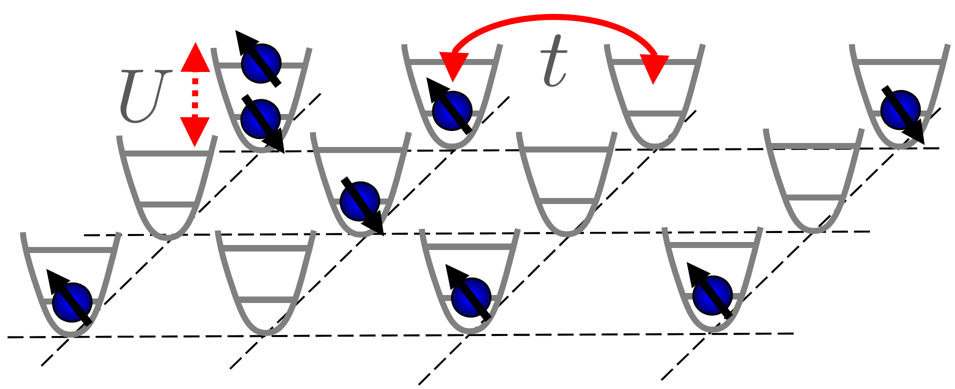
Strong correlations

Interactions are dominant in materials like transition metal oxides, and the CuO_2 planes in high-T_c superconductors. The reason is:

- "core-like" *d* or *f* valence orbitals
- small overlap = narrow bands
- confinement = large repulsion

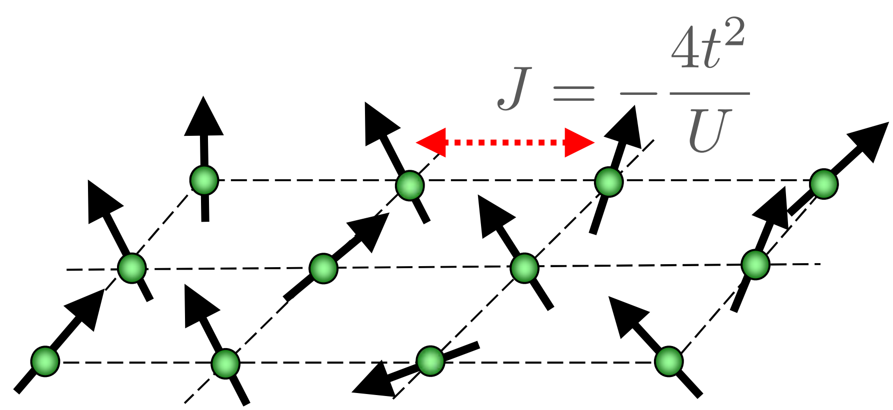


Gives the Hubbard model: $\hat{H}_{\rm HUB} = \hat{H}_{\rm TB} + U \sum_j \hat{a}^\dagger_{j,\uparrow}\hat{a}_{j,\uparrow}\hat{a}^\dagger_{j,\downarrow}\hat{a}_{j,\downarrow}$



Interactions significant – **no** simple quasi-particle picture.

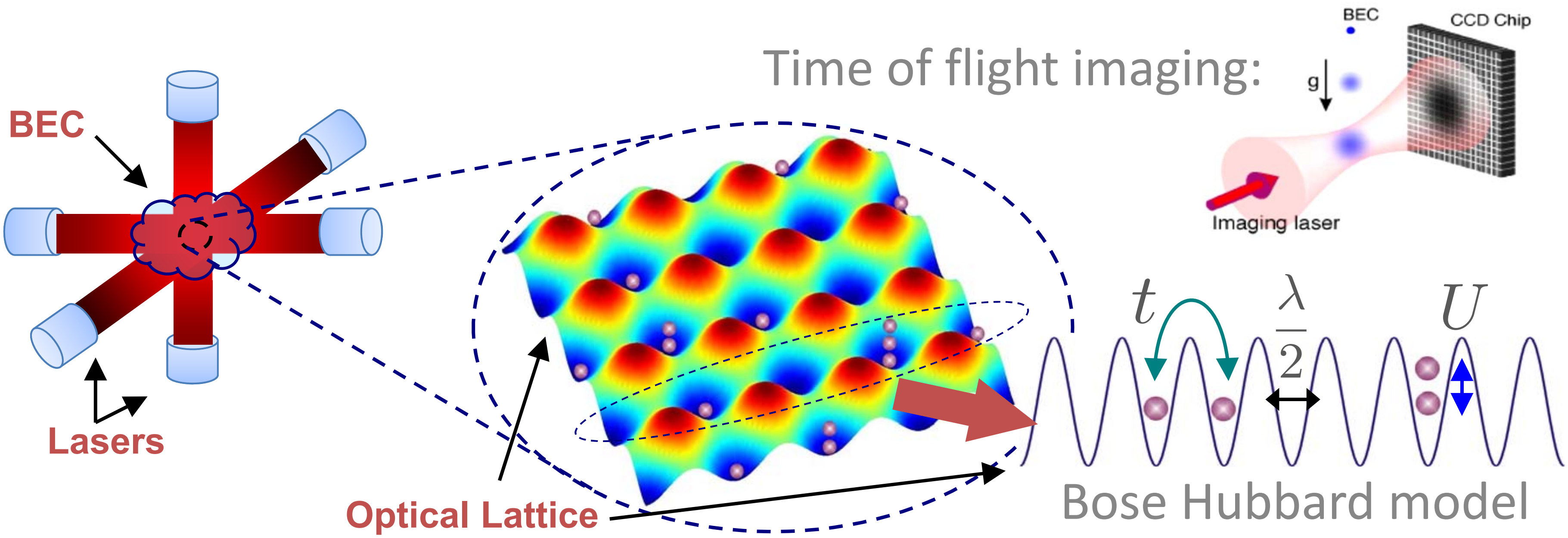
Half-filled strong-coupling limit is a Heisenberg anti-ferromagnet:



Quantum simulations

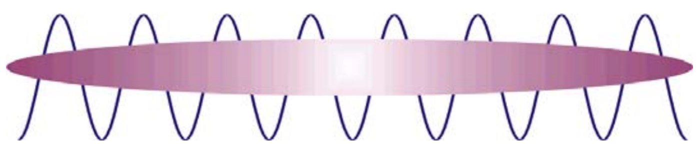
Such model Hamiltonian are now accurately realisable with cold atoms:

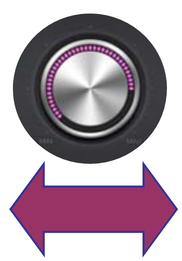
Time of flight imaging:



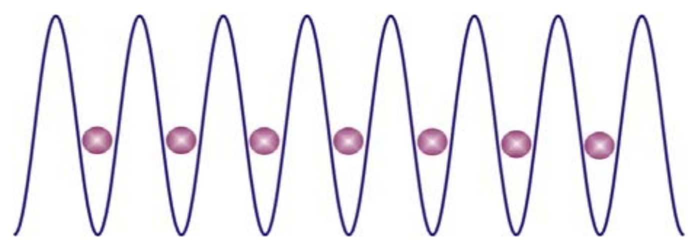
Bose Hubbard model

Superfluid : $t \gg U$





Mott-Insulator : $U \gg t$



quantum phase transition

You will hear more about these AMO systems in other lectures.

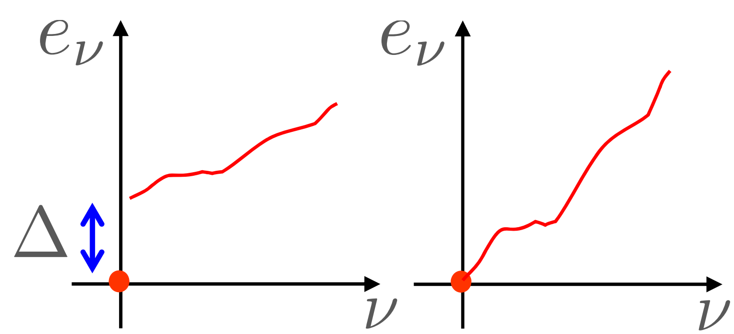
What do we want to compute?

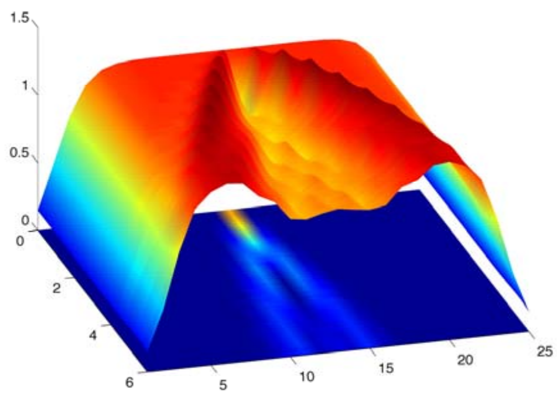
Given a Hamiltonian $\hat{H}$ describing our strongly-correlated system we typically want to compute:

(1) Solve $\hat{H}|\phi_\nu\rangle = e_\nu|\phi_\nu\rangle$

Find ground state and low-lying excitations (also thermal):

gapped or gapless?





(2) Solve $i\frac{d}{dt}|\psi(t)\rangle = \hat{H}(t)|\psi(t)\rangle$

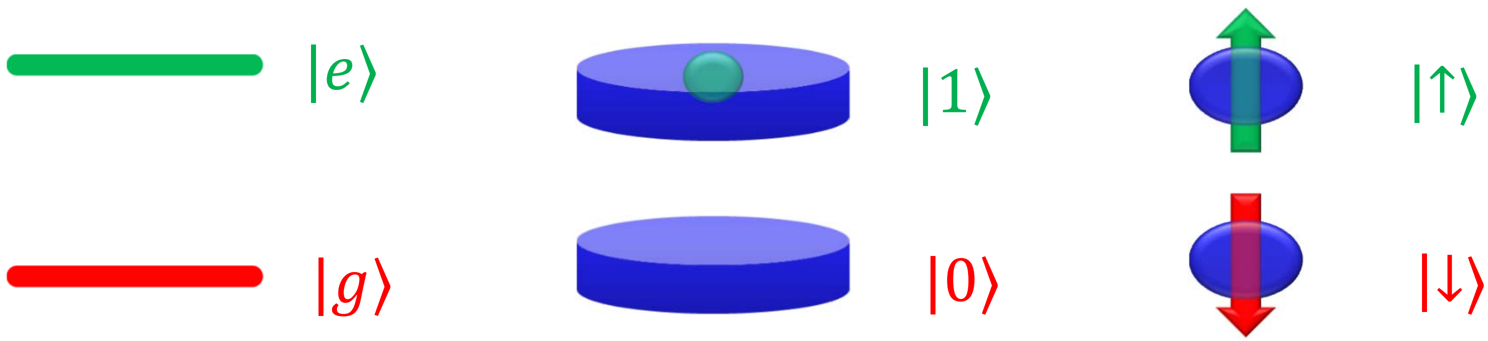
Dynamics of quenches, driving, …

In both cases, we want to compute local observables and long-ranged correlations and other properties.

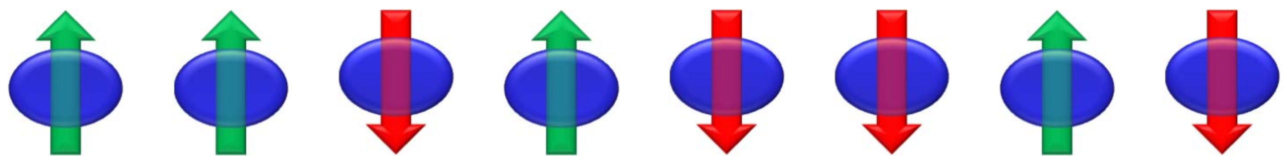
These reduce to linear algebra problems we'll review shortly …

This lecture: spin chains only

- Consider two level systems



- arranged along a one dimensional chain labelled by index l



$$|\Psi\rangle = |\uparrow\rangle \otimes |\uparrow\rangle \otimes |\downarrow\rangle \otimes |\uparrow\rangle \otimes |\downarrow\rangle \otimes |\downarrow\rangle \otimes |\uparrow\rangle \otimes |\downarrow\rangle$$

- Pauli operators

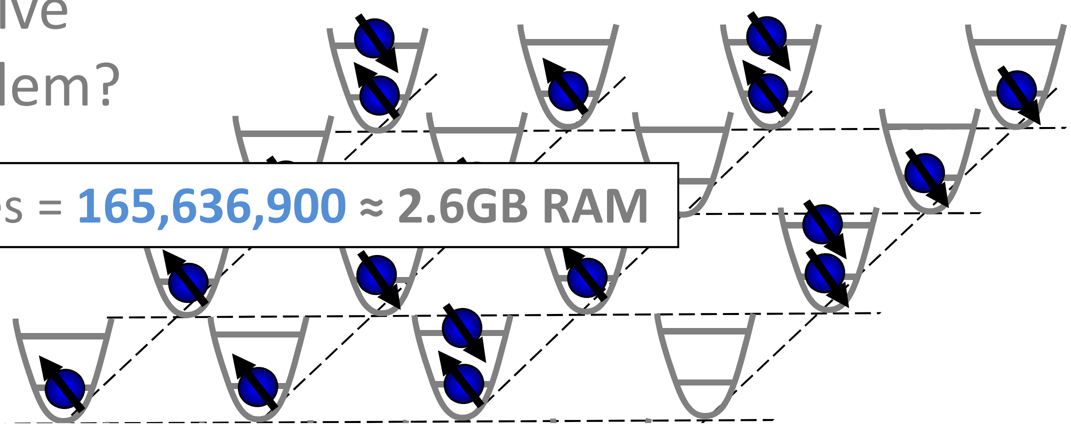
$$\sigma_l^x|\uparrow\rangle_l = |\downarrow\rangle_l \qquad \sigma_l^y|\uparrow\rangle_l = i|\downarrow\rangle_l \qquad \sigma_l^z|\uparrow\rangle_l = |\uparrow\rangle_l$$

$$\sigma_l^x|\downarrow\rangle_l = |\uparrow\rangle_l \qquad \sigma_l^y|\downarrow\rangle_l = -i|\uparrow\rangle_l \qquad \sigma_l^z|\downarrow\rangle_l = -|\downarrow\rangle_l$$

Curse of dimensionality

But, classically simulating many-body quantum systems *seems* to be very hard: *Description* (storage) / *Computational effort* } Scale **exponentially** with size!

Can we solve
4 × 4 problem?



amplitudes = **165,636,900** ≈ **2.6GB RAM**

What about a 6 × 6 problem? | # amplitudes = **82,358,080,713,306,090,000**

Brute-force approach extremely limited. New methods needed …

3: Recap on Linear Algebra

Representations

To do numerical calculations we need an explicit representation mapping our description in one basis to $\mathbb{C}^n$. To do this we write:

Abstract | **Explicit** (Basis dependent)

$$|v_i\rangle \longrightarrow \vec{e}_i = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1_i \\ \vdots \\ 0 \end{pmatrix}$$

Thus any state maps to a vector in $\mathbb{C}^n$

$$|x\rangle = \sum_i x_i |v_i\rangle \longrightarrow \vec{x} = \sum_i x_i \vec{e}_i = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Representations

The "native" inner-product in $\mathbb{C}^n$ is simply: $\vec{x}^\dagger \vec{y} = \sum_i x_i^* y_i$

since $\vec{e}_i^\dagger \vec{e}_j = \delta_{ij}$ as the standard basis is natively orthonormal.

How our actual inner-product $\langle x|y\rangle$ appears in $\mathbb{C}^n$ depends on our choice of basis. Suppose:

$$\langle v_i|v_j\rangle = S_{ij} \quad \text{i.e. a general non-orthogonal basis.}$$

We then encode these overlaps as a matrix $\mathbf{S} = \sum_{ij} \vec{e}_i S_{ij} \vec{e}_j^\dagger$

So given $|x\rangle = \sum_i x_i |v_i\rangle$ and $|y\rangle = \sum_i y_i |v_i\rangle$

We get $\langle x|y\rangle = \sum_{ij} x_i^* S_{ij} y_j = \vec{x}^\dagger \mathbf{S} \vec{y}$

Representations

So the actual inner-product becomes a "weighted" inner-product in $\mathbb{C}^n$. For now we will choose an **orthonormal** basis

$$\langle v_i | v_j \rangle = S_{ij} = \delta_{ij} \longrightarrow \langle x | y \rangle = \sum_i x_i^* y_i$$

so such distinctions disappear. However non-orthogonal bases will raise their head later so beware (!)

Given an operator $\hat{A}$ on $\mathbb{V}$ which maps one state to another as:

$$\hat{A}|x\rangle = |y\rangle$$

compute matrix elements as

$$A_{ij} = \langle v_i | \hat{A} | v_j \rangle$$

Then we represent it as a matrix $\mathbf{A} = \sum_{ij} \vec{e}_i^{\dagger} A_{ij} \vec{e}_j$

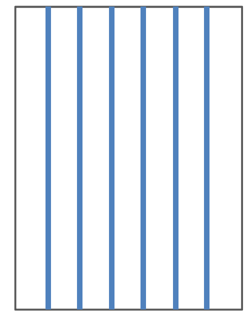
For an orthogonal basis the operator acts as $\mathbf{A}\vec{x} = \vec{y}$ in $\mathbb{C}^n$.

Some basic linear algebra

Rank of matrix

It's the number of linearly independent columns of $\mathbf{A} \in \mathbb{C}^{m \times n}$

Full rank means that $\mathrm{rank}(\mathbf{A}) = n$



Unitary matrices

It is a square matrix $\mathbf{U} \in \mathbb{C}^{m \times m}$ obeying

$$\mathbf{U}^\dagger \mathbf{U} = \mathbf{U}\mathbf{U}^\dagger = \mathbb{1}$$ which implies $\mathbf{U}^{-1} = \mathbf{U}^\dagger$

or in other words its row or column vectors $\{\vec{u}_j\}$ are orthonormal.

Unitaries preserve inner-products

$$(\mathbf{U}\vec{x})^\dagger(\mathbf{U}\vec{y}) = \vec{x}^\dagger \mathbf{U}^\dagger \mathbf{U}\vec{y} = \vec{x}^\dagger \vec{y}$$

Some basic linear algebra

Vector and matrix norms

The notion of size or distance will be crucial for us. Generally one can define a *p*-norm as:

$$||\vec{x}||_p = \left(\sum_{j=1}^{m} |x_j|^p\right)^{1/p} \longrightarrow ||\vec{x}||_2 = \sqrt{\vec{x}^\dagger \vec{x}}$$

The 2-norm defined from our inner-product will often be sufficient.

For matrices we can define a direct generalisation called the Frobenius norm:

$$||\mathbf{A}||_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |A_{ij}|^2}$$

Matrix decompositions

Later we will repeatedly exploit ways of breaking matrices up into parts with special properties. Let's review these here:

Eigenvalue decomposition

Obviously central to QM. More formally $\vec{x}$ is an eigenvector and λ is its eigenvalue for a square matrix if:

$$\mathbf{A}\vec{x} = \lambda\vec{x}$$

Have to solve linear system of equations $(\mathbf{A} - \lambda\mathbb{1})\,\vec{x} = 0$

Non-trivial solutions are the roots of $p_{\mathbf{A}}(z) = \det(\mathbf{A} - z\mathbb{1})$

$$p_{\mathbf{A}}(z) = (z - \lambda_1)(z - \lambda_2)\cdots(z - \lambda_m)$$

A square matrix $\mathbf{A} \in \mathbb{C}^{m\times m}$ has m eigenvalues.

Matrix decompositions

(*algebraic multiplicity*) = # of times eigenvalue λ appears
(*geometric multiplicity*) = # linearly independent eigenvectors $\vec{x}$

If (*algebraic multiplicity*) = (*geometric multiplicity*)

Then it's a **non-defective matrix**

$$\mathbf{AX} = \mathbf{X\Lambda}$$

$\mathbf{X} \in \mathbb{C}^{m \times m}$

matrix of eigenvectors $\vec{x}$

$\mathbf{\Lambda} \in \mathbb{C}^{m \times m}$

diagonal matrix of eigenvalues λ

Since $\mathbf{X}$ is non-singular we have the eigenvalue decomposition:

$$\mathbf{A} = \mathbf{X\Lambda X}^{-1}$$

Matrix decompositions

The matrix $\mathbf{A}$ is said to be *diagonalisable* with $\mathbf{X}$ containing its *right* eigenvectors. The *left* eigenvectors follow similarly:

$$\vec{y}^{\dagger}\mathbf{A} = \lambda \vec{y}^{\dagger} \longrightarrow \mathbf{Y}^{\dagger}\mathbf{A} = \mathbf{\Lambda}\mathbf{Y}^{\dagger} \longrightarrow \mathbf{Y}^{\dagger} = \mathbf{X}^{-1}$$

If $\mathbf{A}^{\dagger}\mathbf{A} = \mathbf{A}\mathbf{A}^{\dagger}$ then it's *normal* and $\mathbf{X} \equiv \mathbf{U}$ is unitary

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\dagger}$$ *unitarily diagonalisable*

Since $\mathbf{Y} = (\mathbf{X}^{-1})^{\dagger} = \mathbf{U}$ left - right eigenvectors coincide and form an orthonormal basis set.

Hermitian matrices are manifestly normal *and* have real eigenvalues.

If eigenvalues are non-negative then its *positive semi-definite* matrix.

Matrix decompositions

Now consider a different perspective. Take a hermitian operator $\hat{A}$ and suppose we want to find the state $|x\rangle$ which minimises:

$$R = \frac{\langle x|\hat{A}|x\rangle}{\langle x|x\rangle} \xrightarrow{\mathbb{C}^m} R = \frac{\vec{x}^\dagger \mathbf{A}\vec{x}}{\vec{x}^\dagger \vec{x}}$$

Rayleigh quotient

Formulate as a constrained minimisation with Lagrange multiplier λ

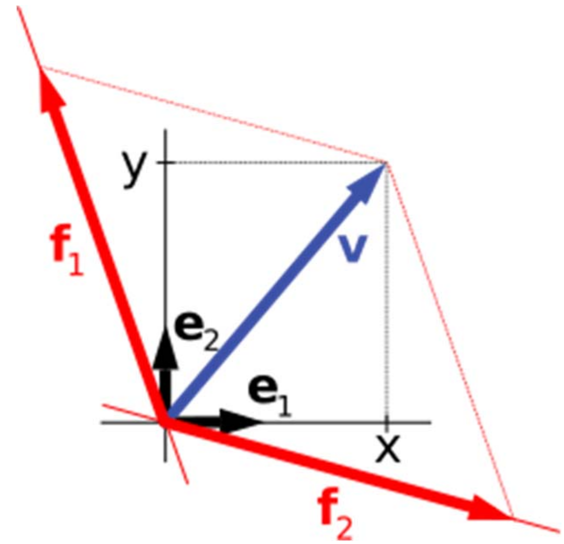
$$F = \vec{x}^\dagger \mathbf{A}\vec{x} - \lambda(\vec{x}^\dagger \vec{x} - 1)$$

$$\frac{\partial F}{\partial \vec{x}^\dagger} = \mathbf{A}\vec{x} - \lambda \vec{x} = 0, \quad \Rightarrow \quad \mathbf{A}\vec{x} = \lambda \vec{x}$$

Stationarity occurs for eigenvectors of $\mathbf{A}$ with a value equal to the eigenvalue. Compute the smallest eigenvalue to minimise.

Matrix decompositions

What happens if we didn't use an orthonormal basis $\{|v_i\rangle\}$?



When we map to $\mathbb{C}^m$ we get instead:

$$R = \frac{\vec{x}^\dagger \mathbf{A} \vec{x}}{\vec{x}^\dagger \mathbf{S} \vec{x}}, \Rightarrow F = \vec{x}^\dagger \mathbf{A} \vec{x} - \lambda(\vec{x}^\dagger \mathbf{S} \vec{x} - 1)$$

Here $\mathbf{A}$ is still the matrix made from $A_{ij} = \langle v_i | \hat{A} | v_j \rangle$ and so is still hermitian.

The overlap matrix $\mathbf{S}$ is by its definition hermitian and positive semi-definite matrix.

Matrix decompositions

Solution to this optimisation problem becomes:

$$\mathbf{A}\vec{x} = \lambda \mathbf{S}\vec{x}$$

This is a *generalised eigenvalue problem*. It will appear frequently ...

Since $\mathbf{S}$ is invertible (it's a similarity transform) we can formally convert it back to a normal eigenvalue problem as

$$\mathbf{S}^{-1}\mathbf{A}\vec{x} = \lambda\vec{x}$$

Note that $\mathbf{S}^{-1}\mathbf{A}$ still has real eigenvalues, but **not** orthonormal eigenvectors according to the native inner-product of $\mathbb{C}^m$. Rather

$$\vec{x}_i^\dagger \mathbf{S}\vec{x}_j = \delta_{ij}$$

so the eigenvectors **are** orthonormal in the actual inner-product, as they must be since the original problem was hermitian.

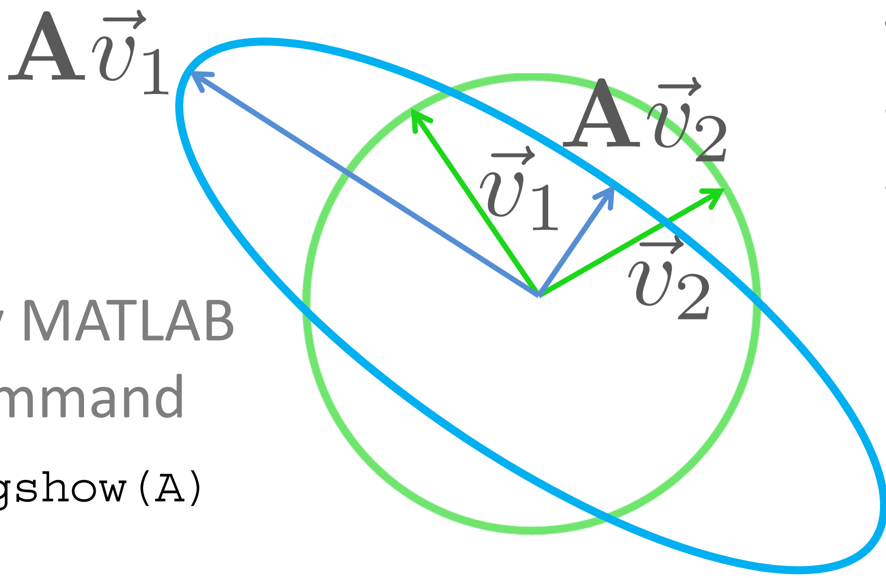
Matrix decompositions

Singular value decomposition

The SVD is a slightly less well known decomposition. Its motivated from the following geometrical fact:

“The image of the unit sphere under any matrix is a hyperellipse”

This picture illustrates the point for a 2 × 2 matrix $\mathbf{A}$:



Try MATLAB command

```
eigshow(A)
```

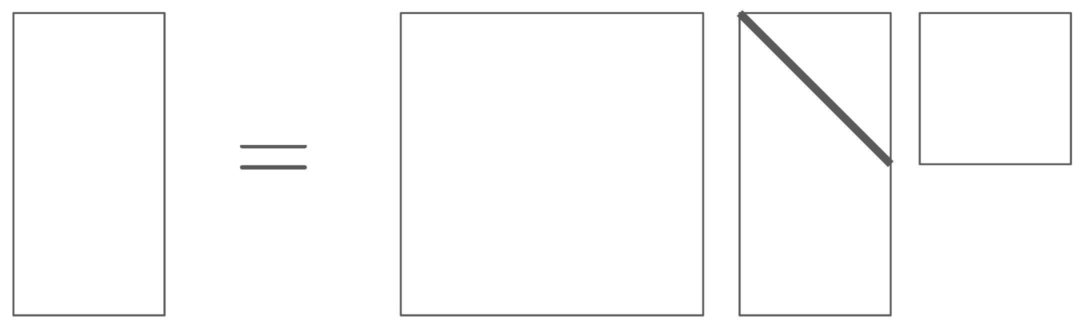
The green unit circle is mapped to the blue ellipse under the transformation by $\mathbf{A}$.

For a certain input $\vec{v}_1$ the image $\mathbf{A}\vec{v}_1$ is major/minor axis of the ellipse.

Matrix decompositions

Formally stated the SVD guarantees that for **any** $\mathbf{A} \in \mathbb{C}^{m \times n}$

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\dagger}$$



where $\mathbf{D} \in \mathbb{R}_{+}^{m \times n}$ and diagonal

$\mathbf{U} \in \mathbb{C}^{m \times m}$, $\mathbf{V} \in \mathbb{C}^{n \times n}$ } and unitary

$$\mathbf{U}^{\dagger}\mathbf{U} = \mathbf{U}\mathbf{U}^{\dagger} = \mathbb{1}$$

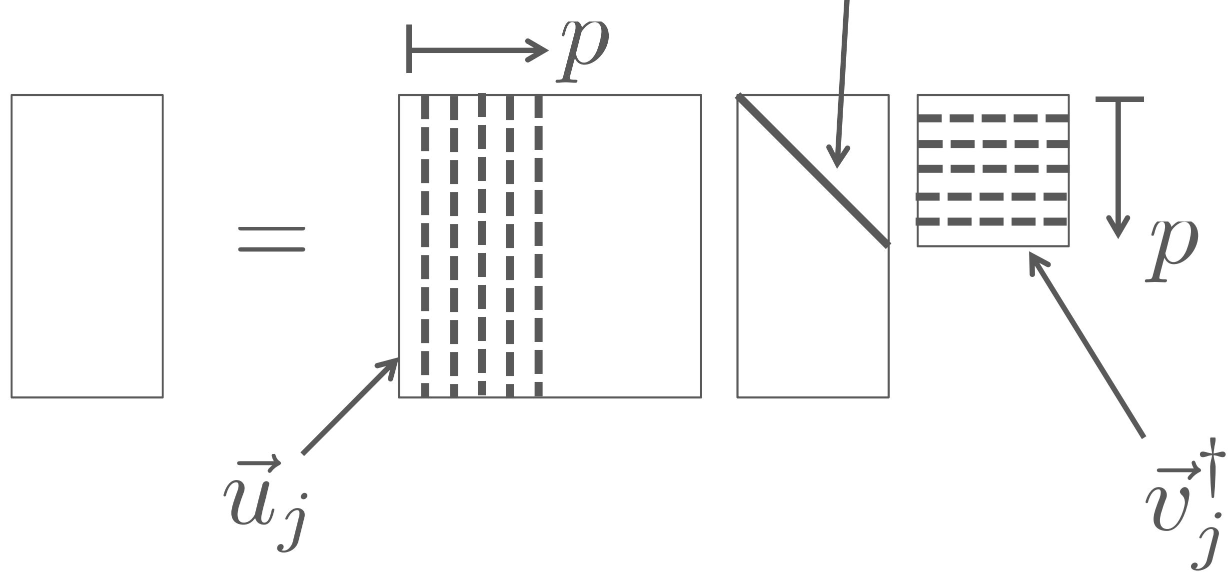
$$\mathbf{V}^{\dagger}\mathbf{V} = \mathbf{V}\mathbf{V}^{\dagger} = \mathbb{1}$$

Matrix decompositions

The diagonal matrix $\mathbf{D}$ has $p = \min(m, n)$ non-negative values:

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0$$

These are the "singular values" of $\mathbf{A}$



The $1 \leq j \leq p$ columns of $\mathbf{U}$ and $\mathbf{V}$ form the left and right "singular vectors" of $\mathbf{A}$, respectively.

Matrix decompositions

We can thus write the SVD as a sum of rank-1 projectors:

$$\mathbf{A} = \sum_{j=1}^{p} \sigma_j \vec{u}_j \vec{v}_j^\dagger$$

This generalises our earlier 2 × 2 example. Acting $\mathbf{A}$ on $\vec{v}_j$ gives

$$\mathbf{A}\vec{v}_j = \sigma_j \vec{u}_j$$

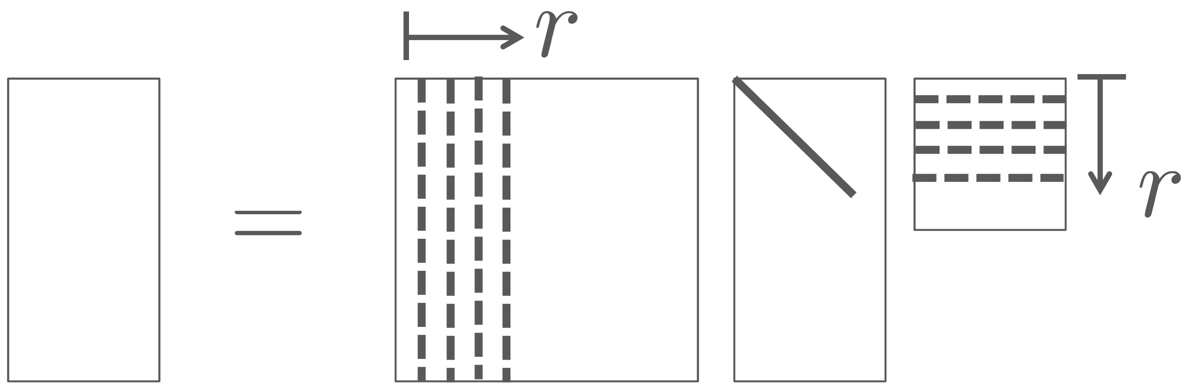
Each unit vector $\vec{v}_j$ is mapped to a unit vector $\vec{u}_j$ rescaled by σ_j which form the semiaxes of the hyperellipse.

Unlike an eigenvalue decomposition **every** matrix has an SVD and its singular values are uniquely determined.

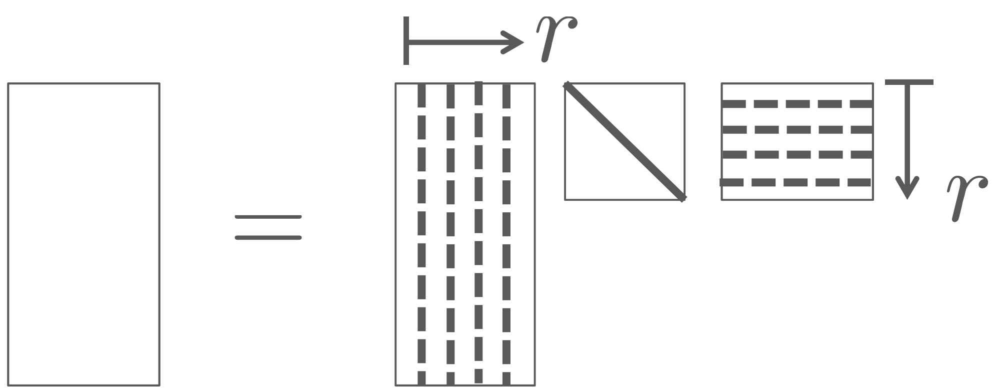
For every distinct σ_j the singular vectors are also uniquely determined up to a phase: $\vec{u}_j \mapsto e^{i\phi}\vec{u}_j, \ \vec{v}_j \mapsto e^{i\phi}\vec{v}_j$

Matrix decompositions

As it stand the SVD contains redundant information since we only used at most the first p columns. Suppose $\mathbf{A}$ has rank $r < p$ then



So let's simply discard the redundant bits to get:



$$\mathbf{U} \in \mathbb{C}^{m \times r}$$

$$\mathbf{D} \in \mathbb{R}_{+}^{r \times r}$$

$$\mathbf{V} \in \mathbb{C}^{n \times r}$$

Matrix decompositions

This is called a *reduced* SVD. Since $\mathbf{U}$ and $\mathbf{V}$ are no longer square they cannot be unitary. However their columns are still orthonormal so they continue to obey:

$$\mathbf{U}^\dagger\mathbf{U} = \mathbb{1} \quad \text{and} \quad \mathbf{V}^\dagger\mathbf{V} = \mathbb{1}$$

The SVD has some connection to eigenvalue decompositions. Taking $\mathbf{A}$ we can form two new positive semi-definite matrices:

$$\mathbf{B} = \mathbf{A}^\dagger\mathbf{A} \quad \text{and} \quad \mathbf{C} = \mathbf{A}\mathbf{A}^\dagger$$

Using the SVD of $\mathbf{A}$ we then get that:

$$\mathbf{B} = \mathbf{V}\mathbf{D}^2\mathbf{V}^\dagger \quad \text{and} \quad \mathbf{C} = \mathbf{U}\mathbf{D}^2\mathbf{U}^\dagger$$

So both share non-negative eigenvalues $\sigma_1^2 \geq \sigma_2^2 \geq \cdots \geq \sigma_p^2$ and their eigenvectors are the left and right singular vectors.

Matrix decompositions

Perhaps the most important property of the SVD is its ability to provide *low-rank approximations* of matrices. Take the following:

Given $\mathbf{A}$ has rank *r* what is the "best" approximation $\tilde{\mathbf{A}}$ to $\mathbf{A}$ with rank $\chi < r$. By "best" we mean it minimises:

$$\epsilon = ||\mathbf{A} - \tilde{\mathbf{A}}||_F$$

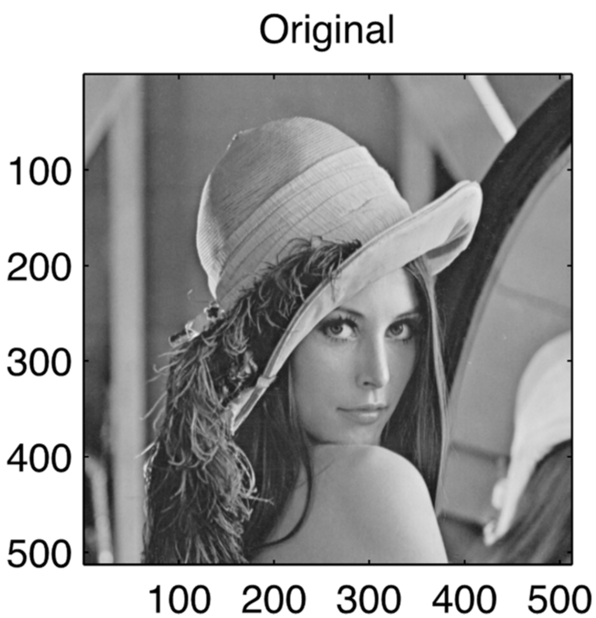
The *remarkable* result is that $\tilde{\mathbf{A}}$ is just the partial sum of the SVD up to the χ-th term:

$$\tilde{\mathbf{A}} = \sum_{j=1}^{\chi} \sigma_j \vec{u}_j \vec{v}_j^\dagger$$

with a residual error given by $\epsilon = \sum_{j=\chi+1}^{r} \sigma_j^2.$

Matrix decompositions

This result has many applications. Here is an example of image compression:



The full uncompressed image formally has a rank of $r = 512$, however we can severely truncate the SVD and still retain a good image.

Later we will use the same technique to compress quantum states …

Computational complexity

In the forthcoming lectures will be very interested in how expensive operations are. For this we will need to know the computational complexity of the linear algebra operations just discussed:

Dense matrix multiplication $\mathbf{A} \in \mathbb{C}^{n \times n}$ $\mathbf{B} \in \mathbb{C}^{n \times n}$

$\mathbf{AB}$ naively $\mathcal{O}(n^3)$ more optimally $\mathcal{O}(n^{2.81})$

Eigenvalue decomposition assuming $\mathbf{A}$ is diagonalisable

$$\mathbf{A} = \mathbf{X}\mathbf{\Lambda}\mathbf{X}^{-1} \searrow$$

Singular value multiplication $\mathcal{O}(n^3)$

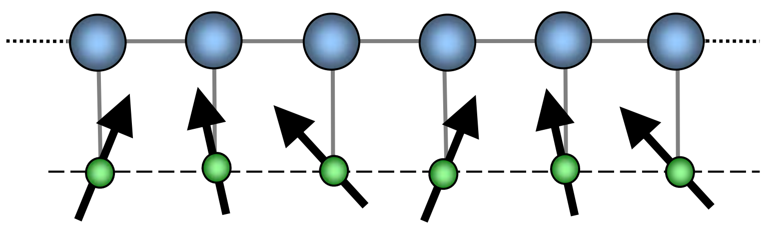
$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\dagger} \nearrow$$

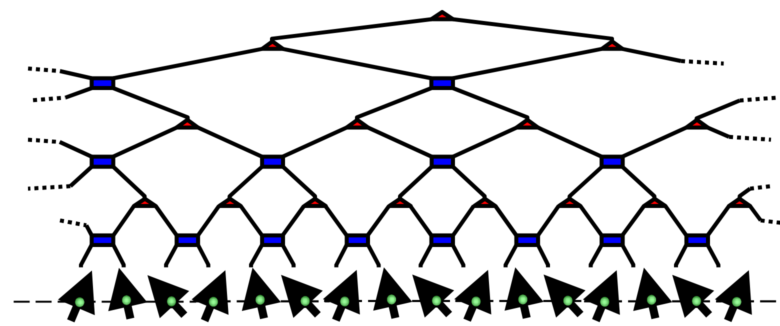
The “Big-O” notation hides constants crucial for practical calculations – using good linear algebra libraries is essential.

Next lecture …

Over the next 3 lectures we will introduce tensors and explore tensor network like matrix product states, and discuss others …

1D





2D