Content

Content

3. Statistical and many-body physics

3.1 Classical Monte Carlo and phase transitions (Ising model and critical phenomena)

3.2 Quantum Monte Carlo methods (Path-integral and cluster update)

4. Machine learning in physics and High performance computation

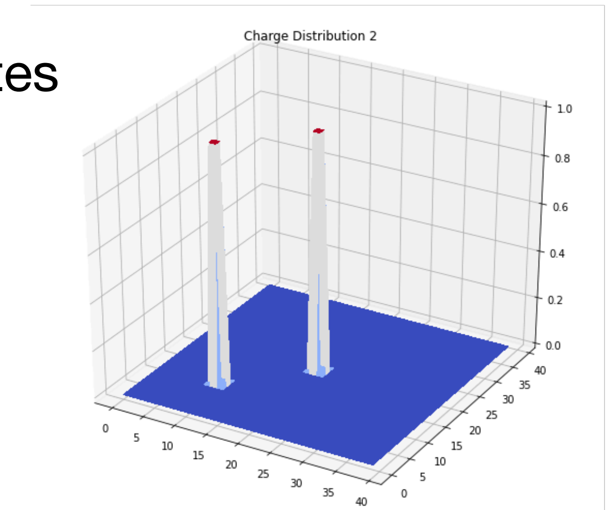
4.1 AI in quantum physics

4.2 HPC and parallelism

4.3 …

Partial differential equations

1st, 2nd derivatives of spatial and time coordinates



- Poisson equation

$$\Delta\phi(\vec{r}) = -\frac{1}{\epsilon_0}\rho(\vec{r})$$

Laplacian $\Delta = \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2}$

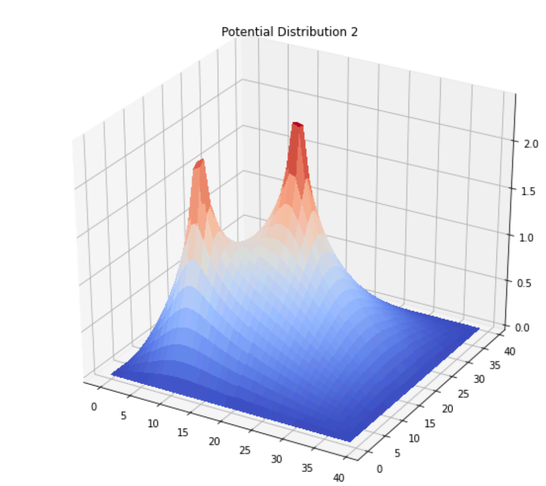
$\rho(\vec{r})$ charge density inside domain V

$\phi(\vec{r})$ electrostatic potential

elliptic PDE

Dirichlet boundary condition $\phi(\vec{r}), \vec{r} \in \partial V$

Neumann boundary condition $(\vec{n} \cdot \vec{\nabla})\phi(\vec{r}), \vec{r} \in \partial V$

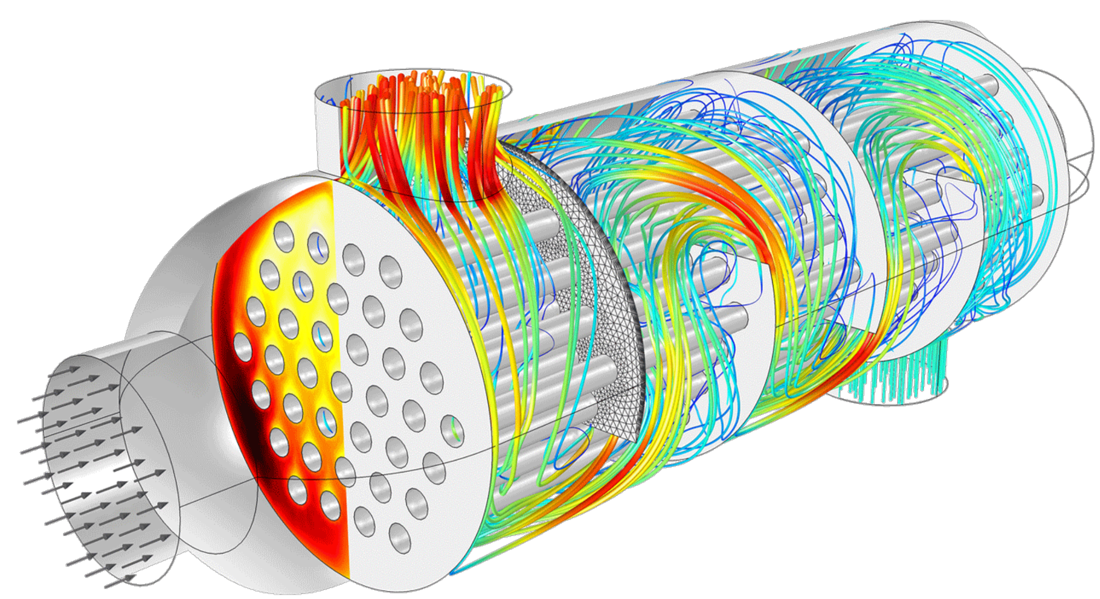


- Diffusion equation

$$\frac{\partial u(\vec{r},t)}{\partial t} - D\Delta u(\vec{r},t) = S(\vec{r},t)$$

$u(\vec{r},t)$ concentration of a substance at position $\vec{r}$ and time t

$S(\vec{r},t)$ source/drain D diffusion coefficient



parabolic PDE asymmetrical under time-reversal $t \to -t$

Cauchy initial value problem $u(\vec{r}, t=0)$ on domain V

Neumann boundary condition $(\vec{n} \cdot \vec{\nabla})u(\vec{r}) = 0, \vec{r} \in \partial V$

Initial configuration must be consistent with the boundary condition

- Wave equation $$\frac{1}{c^2}\frac{\partial^2 u(\vec{r},t)}{\partial t^2} - \Delta u(\vec{r},t) = S(\vec{r},t)$$

c wave velocity

Hyperbolic PDE symmetrical under time-reversal, oscillations $t \to -t$

initial value problem $u(\vec{r}, t=0), \frac{\partial u(\vec{r},t)}{\partial t}|_{t=0}$

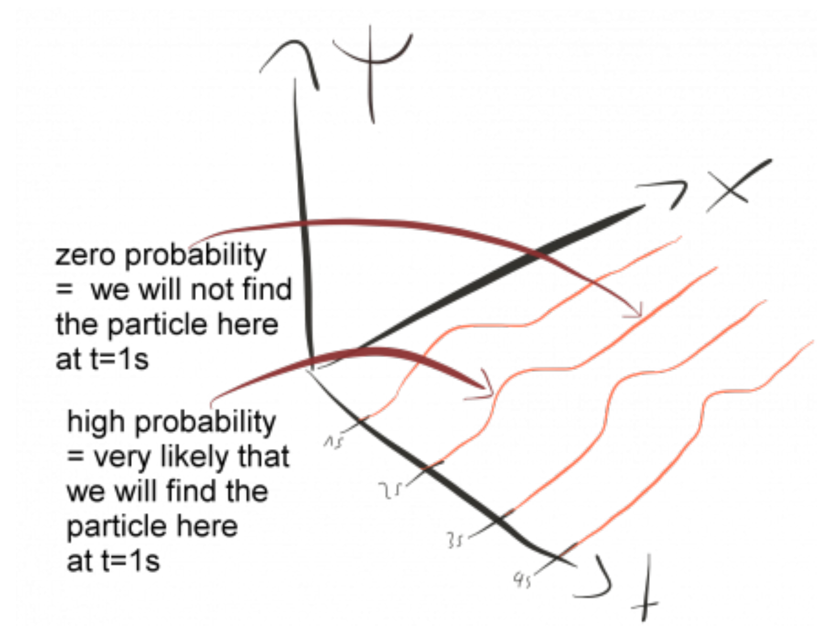
Initial configuration must be consistent with the boundary condition

浮世绘，葛饰北斋，神奈川冲浪里

- Schrödinger equation $$i\hbar\frac{\partial \Psi(\vec{r},t)}{\partial t} = H\Psi(\vec{r},t)$$

For free particle $H = -\frac{\hbar^2}{2m}\Delta$

diffusion equation in imaginary time



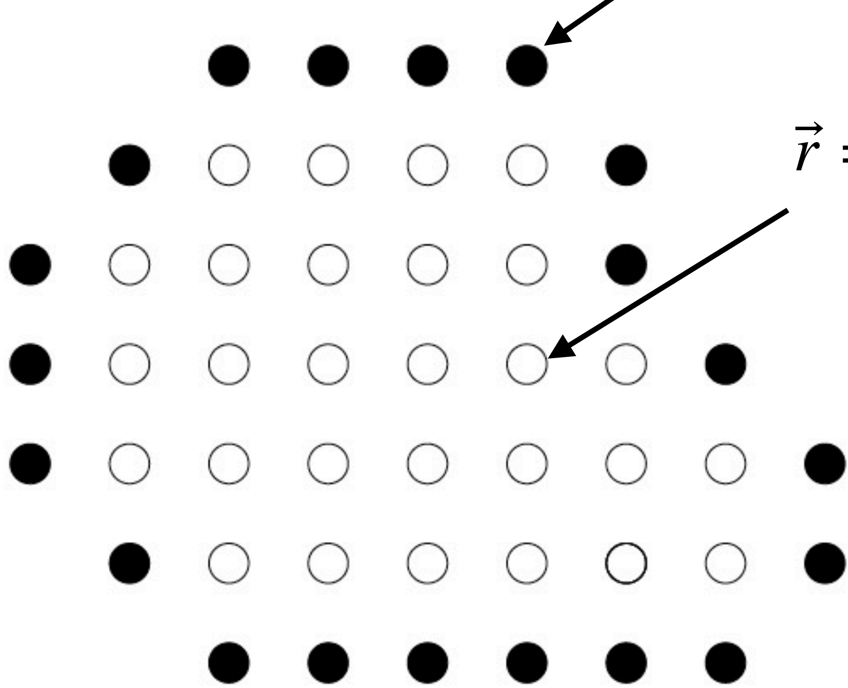
Animating Schrödinger's Equation

https://www.youtube.com/watch?v=Xj9PdeY64rA

- Fluid Dynamics, Navier-Stokes equation

Discretization

boundary ∂V



$\vec{e}_i, i = 1,\cdots, d$ unit vectors of a d-dimensional hyper cubic lattice

$$\vec{r} = \sum_{i=1}^{d=2} h_i r_i \vec{e}_i = \sum_{i=1}^{d=2} r_i \vec{e}_i$$

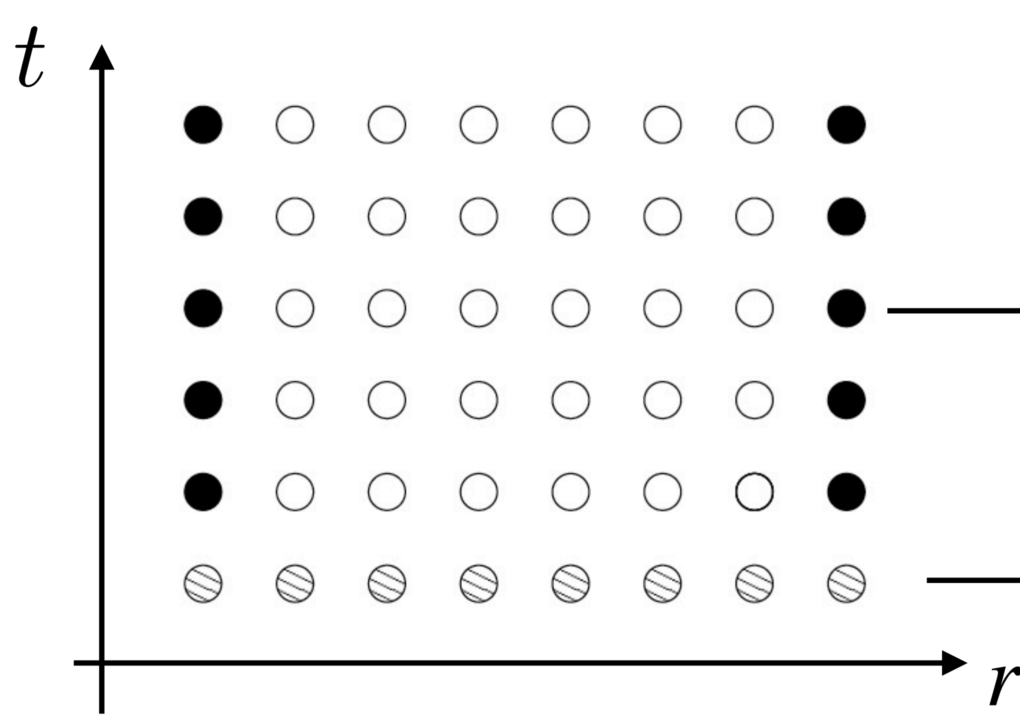
$h_i = h = 1$ lattice spacing

r_i defines the domain V

$$t_n = n\tau, n = 0, 1, 2, \cdots$$

$$u(\vec{r}, t) \rightarrow u(\vec{r}_i, t_n)$$

Discretisation of space-time domain



Compatible boundary conditions

Initial values

$$u(\vec{r}, t = 0), \frac{\partial u(\vec{r}, t)}{\partial t}\Big|_{t=0}$$

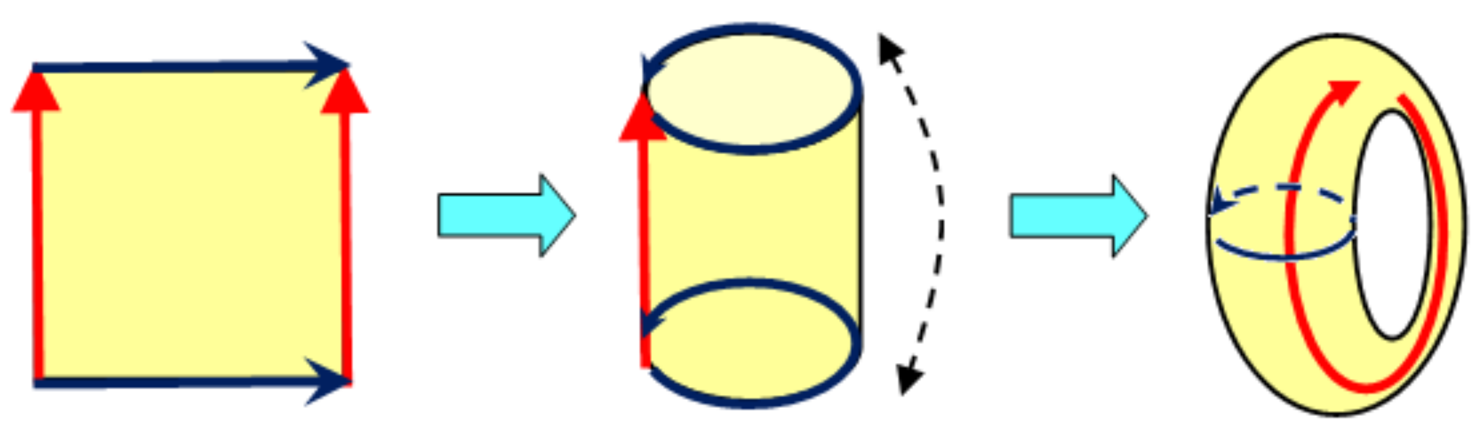
Forward time (FT) discretisation $$\frac{\partial u(\vec{r},t_n)}{\partial t} \to \frac{u(\vec{r},t_{n+1}) - u(\vec{r},t_n)}{\tau} + O(\tau)$$

Centered space (CS) discretisation $$\frac{\partial u(\vec{r},t_n)}{\partial x_i} \to \frac{u(\vec{r}+h_i\overrightarrow{e}_i,t_n) - u(\vec{r}-h_i\overrightarrow{e}_i,t_n)}{2h_i} + O(h^2)$$

$$\frac{\partial^2 u(\vec{r},t_n)}{\partial x_i^2} \to \frac{u(\vec{r}+h_i\overrightarrow{e}_i,t_n) + u(\vec{r}-h_i\overrightarrow{e}_i,t_n) - 2u(\vec{r},t_n)}{h_i^2} + O(h^2)$$

Boundary conditions

Periodic boundary conditions (PBC) $u(\vec{r}+N_ih_i\overrightarrow{e}_i) = u(\vec{r})$



1d ring
2d torus (donut)
higher-d tori

Poisson equation (Potential Problems)

$$\Delta\phi(\vec{r}) = -\frac{1}{\epsilon_0}\rho(\vec{r}) \qquad \vec{E} = -\nabla\phi \qquad \Delta\phi(\vec{r}) = 4\pi G\rho(\vec{r}) \qquad \vec{g} = -\nabla\phi$$

$$\phi(\vec{r}) = \frac{Q}{4\pi\epsilon_0|\vec{r}|} \qquad \phi(\vec{r}) = -\frac{Gm}{|\vec{r}|}$$

1781-1840
French mathematician, physicist

Centered three-point formula in each direction

$$\Delta\phi(\vec{r}) \to \frac{1}{h^2}(-2d\phi(\vec{r}) + \sum_{i=1}^{d}(\phi(\vec{r}+h\vec{e}_i) + \phi(\vec{r}-h\vec{e}_i))) + O(h^2)$$

$$\phi(\vec{r}) = \frac{1}{2d}\sum_{i=1}^{d}(\phi(\vec{r}+h\vec{e}_i) + \phi(\vec{r}-h\vec{e}_i)) + \frac{h^2}{2d\epsilon_0}\rho(\vec{r})$$

Jacobi Relaxation

1. Initial configuration for $\phi(\vec{r})$ (an educated guess) with boundary conditions

2. Calculation a new configuration $\phi^{new}(\vec{r}) = \frac{1}{2d}\sum_{i=1}^{d}[\phi(\vec{r}+h\vec{e}_i) + \phi(\vec{r}-h\vec{e}_i)] + \frac{h^2}{2d\epsilon_0}\rho(\vec{r})$

   Use the boundary condition to determine $\phi^{new}(\vec{r}), \vec{r} \in \partial V$

3. Calculate the deviation $\delta\phi = \max_{\vec{r}}|\phi^{new}(\vec{r}) - \phi(\vec{r})|$

4. Replace ϕ by ϕ^{new}. If $\delta\phi$ smaller than the bound, stop; otherwise, repeat from step 2.

Relax towards the actual solution from boundary to the interior of V

Computational complexity $\# \text{ iterations } \approx \frac{1}{2} p N_s^{2/d}$ to improve the error by 10^{-p}

$$\text{for } d = 2, \sim N_s$$

- Gauss-Seidel Relaxation — no need to work with two arrays ϕ and ϕ^{new}

Change the 2nd step in Jacobi

$$\phi(\vec{r}) = \frac{1}{2d} \sum_{i=1}^{d} (\phi(\vec{r} + h\vec{e}_i) + \phi(\vec{r} - h\vec{e}_i)) + \frac{h^2}{2d\epsilon_0} \rho(\vec{r})$$

As one systematically carry out the relaxation

$\phi(\vec{r})$ contains both old and new values during the iteration through the lattice

Save the memory

Speed-up $\# \text{ iterations } \approx \frac{1}{4} p N_s^{2/d}$

Successive Overrelaxation (SOR)

$$\phi(\vec{r}) \to (1-\omega)\phi(\vec{r}) + \omega\left(\frac{1}{2d}\sum_{i=1}^{d}(\phi(\vec{r}+h\vec{e}_i) + \phi(\vec{r}-h\vec{e}_i)) + \frac{h^2}{2d\epsilon_0}\rho(\vec{r})\right)$$

relaxation parameter ω Gauss-Seidel $\omega = 1$

Underrelaxation $0 < \omega < 1$ overrelaxation $1 < \omega < 2$

Put more weight to the new value than in Gauss-Seidel

- Stable for $\omega < 2$
- Converges faster than Gauss-Seidel
- Optimal value of the ω_{opt} close to 2
- $\#\text{ iterations} \approx \frac{1}{3}pN_s^{1/d}$

$$\omega_{opt} = \frac{2}{1+\sqrt{1-\frac{1}{4}(\cos(\frac{\pi}{N_x})+\cos(\frac{\pi}{N_y}))^2}} \approx 1.939$$

with N=100

$N_x \times N_y$ square lattice with Dirichlet boundary condition

$$d = 2, \quad N_s^{2/d} \sim N_s$$

$$d = 2, \quad N_s^{1/d} \sim \sqrt{N_s}$$

Our bible

Chap. 20.5.

http://numerical.recipes/book.html

Matrix-Formulation

$$\Delta\phi(\vec{r}) = -\frac{1}{\epsilon_0}\rho(\vec{r})$$

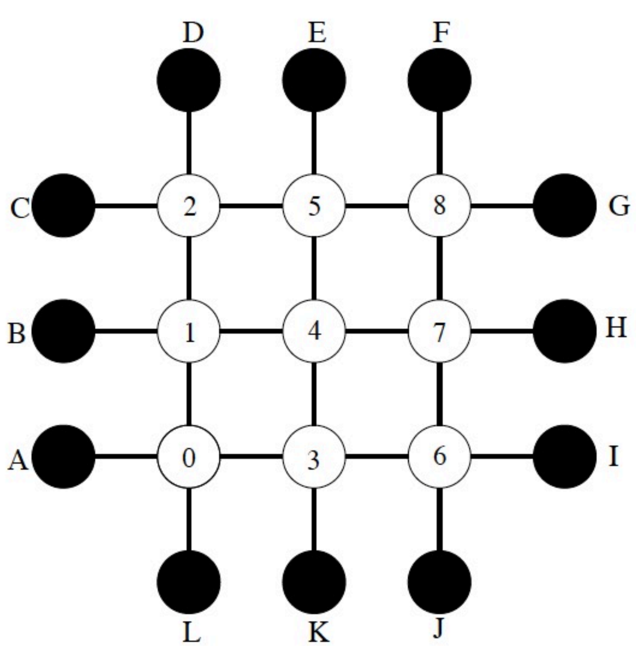
$$\phi(\vec{r}) = \frac{1}{2d}\sum_{i=1}^{d}\left(\phi(\vec{r}+h\vec{e}_i)+\phi(\vec{r}-h\vec{e}_i)\right)+\frac{h^2}{2d\epsilon_0}\rho(\vec{r})$$

A set of linear equations $$A\vec{\phi} = \vec{b}$$

$$\vec{\phi} = \begin{pmatrix} \phi(\vec{x}_0) \\ \phi(\vec{x}_{\vec{r}_1}) \\ \vdots \\ \phi(\vec{x}_{\vec{r}_{N-1}}) \end{pmatrix}$$

$\vec{b}$ boundary condition

A Laplace operator $-\Delta$



$$\text{on site } 0: -\phi_3 - \phi_A - \phi_1 - \phi_L + 4\phi_0 = \frac{h^2}{\epsilon_0}\rho_0$$

$$\text{on site } 1: -\phi_4 - \phi_B - \phi_2 - \phi_0 + 4\phi_1 = \frac{h^2}{\epsilon_0}\rho_1$$

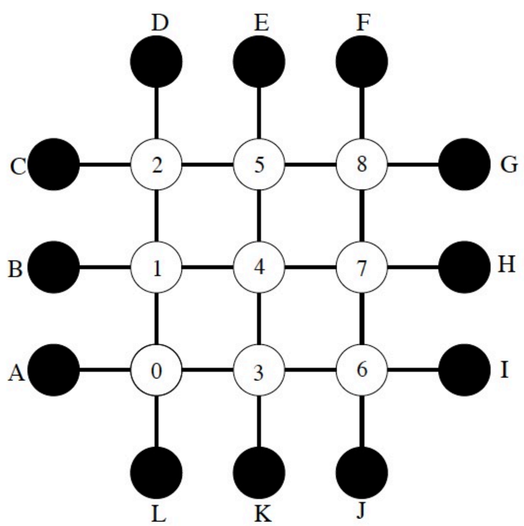
$$\vdots$$

$$\text{on site } 4: -\phi_7 - \phi_1 - \phi_5 - \phi_3 + 4\phi_4 = \frac{h^2}{\epsilon_0}\rho_4$$

$$\vdots$$

$$\text{on site } 8: -\phi_G - \phi_5 - \phi_F - \phi_7 + 4\phi_8 = \frac{h^2}{\epsilon_0}\rho_8$$

Matrix-Formulation



$$
\begin{aligned}
-\phi_3 - \phi_1 + 4\phi_0 &= \frac{h^2}{\epsilon_0}\rho_0 + \phi_A + \phi_L =: b_0 \\
-\phi_4 - \phi_2 - \phi_0 + 4\phi_1 &= \frac{h^2}{\epsilon_0}\rho_1 + \phi_B =: b_1 \\
&\vdots \\
-\phi_7 - \phi_1 - \phi_5 - \phi_3 + 4\phi_4 &= \frac{h^2}{\epsilon_0}\rho_4 =: b_4 \\
&\vdots \\
-\phi_5 - \phi_7 + 4\phi_8 &= \frac{h^2}{\epsilon_0}\rho_8 + \phi_G + \phi_F =: b_8
\end{aligned}
$$

$$A\vec{\phi} = \vec{b}$$

$$
\begin{pmatrix}
4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4
\end{pmatrix}
\begin{pmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \\ \phi_6 \\ \phi_7 \\ \phi_8 \end{pmatrix}
=
\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \end{pmatrix}
$$

- A is symmetric $A^T = A$
- A is sparse $A_{ii} = 2d, A_{ij} = -1 \text{ only if } \langle i, j\rangle$
- A is positive definite $\vec{x}^T A \vec{x} > 0, \text{ for } \vec{x} \neq 0$

$$\int d^d x f^*(\vec{x})(-\Delta) f(\vec{x}) = \int d^d x |\nabla f(\vec{x})|^2 > 0,$$

Partial integration over an infinite domain

Positive semi-definite $v^T Q v = v^T X^T X v = (Xv)^T (Xv) = u^T u \geq 0$

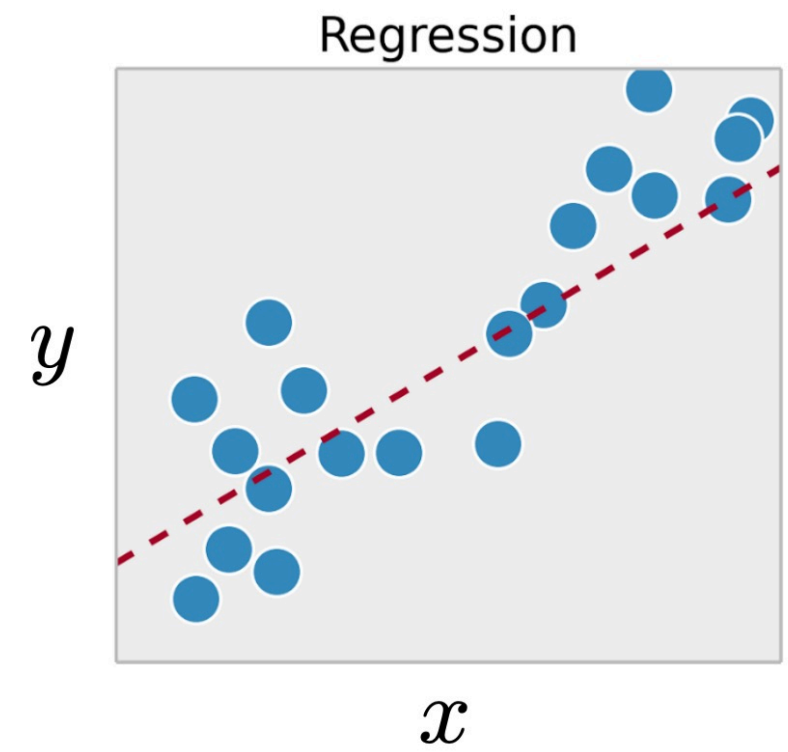
Remember what we learnt in the machine learning lecture ?

https://quantummc.xyz/wp-content/uploads/2023/03/lecture1_1.pdf

$$\{(x_j^{(i)}, y^{(i)}), \theta_j\};\ j = 1, 2, \cdots, N;\ i = 1, 2, \cdots, M; N < M$$

$$y^{(i)} = \theta_0 + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} + \cdots + \theta_N x_N^{(i)}$$

$$\mathbb{R}^{M\times(N+1)} \quad \begin{bmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_N^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_N^{(2)} \\ & \vdots & & \vdots & \\ 1 & x_1^{(M)} & x_2^{(M)} & \cdots & x_N^{(M)} \end{bmatrix} \cdot \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_N \end{bmatrix} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(M)} \end{bmatrix}$$



$$\underline{\underline{X}} \cdot \underline{\Theta} = \underline{Y}$$

Paraboloid and positive-definite

$$f(x) = \frac{1}{2}x^T A x - b^T x + c \qquad \nabla_x f(x) = Ax - b = 0 \qquad Ax = b$$

$$A = \frac{1}{M} X^T X \qquad \mathbb{R}^{(N+1)\times(N+1)}$$

$$b = X^T Y \qquad \mathbb{R}^{(N+1)\times 1}$$

$$A = \underbrace{\begin{bmatrix} \langle 1 \rangle & \langle x_1 \rangle & \langle x_2 \rangle & \cdots & \langle x_N \rangle \\ \langle x_1 \rangle & \langle (x_1)^2 \rangle & \langle x_1 x_2 \rangle & \cdots & \langle x_1 x_N \rangle \\ \langle x_2 \rangle & \langle x_2 x_1 \rangle & \langle (x_2)^2 \rangle & \cdots & \langle x_2 x_N \rangle \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \langle x_N \rangle & \langle x_N x_1 \rangle & \langle x_N x_2 \rangle & \cdots & \langle (x_N)^2 \rangle \end{bmatrix}}_{\mathbb{R}^{(N+1)\times(N+1)}}$$

Gauss Elimination

$$\begin{array}{rcrcrcr} x_0 & + & x_1 & + & x_2 & = & 6 \\ -x_0 & + & 2x_1 & & & = & 3 \\ 2x_0 & & & + & x_2 & = & 5 \end{array}$$

$$A = \begin{pmatrix} 1 & 1 & 1 \\ -1 & 2 & 0 \\ 2 & 0 & 1 \end{pmatrix}, \quad \vec{b} = \begin{pmatrix} 6 \\ 3 \\ 5 \end{pmatrix}, \quad \text{and } \vec{x} = \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix}$$

Eliminate x0 from 2nd and 3rd equation

$$\begin{array}{rcrcrcr} x_0 & + & x_1 & + & x_2 & = & 6 \\ & & 3x_1 & + & x_2 & = & 9 \\ & & -2x_1 & - & x_2 & = & -7 \end{array}$$

Forward elimination $O(n^3)$

For $j = 0, ..., n-2$ do:
For $i = j+1, ..., n-1$ do:

Test with 3×3 matrix

$$\begin{array}{rcl} f & := & -A_{ij}/A_{jj} \\ A_{ik} & \rightarrow & A_{ik} + f A_{jk}, \quad k = j, ..., n-1 \\ b_i & \rightarrow & b_i + f b_j. \end{array}$$

Eliminate x1 from 3rd equation

$$\begin{array}{rcrcrcr} x_0 & + & x_1 & + & x_2 & = & 6 \\ & & 3x_1 & + & x_2 & = & 9 \\ & & & - & \frac{1}{3}x_2 & = & -1 \end{array}$$

Trigonal system $(\tilde{A}, \vec{\tilde{b}})$

$$\tilde{A} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 3 & 1 \\ 0 & 0 & -\frac{1}{3} \end{pmatrix}, \quad \vec{\tilde{b}} = \begin{pmatrix} 6 \\ 9 \\ -1 \end{pmatrix}$$

backsubstitution $O(n^2)$

$$x_{n-1} = \frac{\tilde{b}_{n-1}}{\tilde{A}_{n-1,n-1}}, \qquad x_i = \frac{1}{\tilde{A}_{ii}} \left[\tilde{b}_i - \sum_{j=i+1}^{n-1} \tilde{A}_{ij} x_j \right], \quad i = n-2, ..., 0.$$

Problem: after a few iterations, A_{ii} (almost) zero   Divide very small numbers, leads to breakdown

Gauss Elimination with pivoting

- Rescale each row of $(A, \vec{b}\,)$ so that the maximum element in each row of A equals 1.
- Initialize the index vector, $v(i) = i$, $i = 0, ..., n-1$. for bookkeeping
- For $j = 0, ..., n-2$ do:
  - Search for the pivot element: $|A_{v(p),j}| = \max\{|A_{v(i),j}|, i = j, ..., n-1\}$
    If $j \neq p$, exchange the j-th and p-th entries of v: $v(j) \leftrightarrow v(p)$
    This corresponds to exchanging the j-th and p-th row.
  - For $i = j+1, ..., n-1$ do:

$$\begin{aligned} f &:= -A_{v(i)j} \,/ A_{v(j)j} \\ A_{v(i)k} &\to A_{v(i)k} + f\,A_{v(j)k}, \quad k = j, ..., n-1 \\ b_{v(i)} &\to b_{v(i)} + f\,b_{v(j)}. \end{aligned}$$

Backsubstitution with trigonal system $(\tilde{A}, \vec{\tilde{b}})$

$$x_{n-1} = \frac{\tilde{b}_{v(n-1)}}{\tilde{A}_{v(n-1),n-1}}, \qquad x_i = \frac{1}{\tilde{A}_{v(i)i}} \left[\tilde{b}_{v(i)} - \sum_{j=i+1}^{n-1} \tilde{A}_{v(i)j} x_j \right], \quad i = n-2, ..., 0.$$

LU Decomposition

$\tilde{A} = MA$ where $M = M^{(n-2)} M^{(n-1)} \dots M^{(1)} M^{(0)}$

$\tilde{A}$ is upper triangular matrix

$M^{(j)}$ are lower triangular matrices

$$A = \underbrace{M^{-1}}_{L} \cdot \underbrace{\tilde{A}}_{U} = L \cdot U$$

$$M^{(j)}_{ii} = 1$$

$$M^{(j)}_{ij} = -\frac{A_{ij}}{A_{jj}}, i > j$$

$$M^{(j)}_{ij} = 0 \text{ otherwise}$$

Complexity still scales $O(N^3)$

$O(N^2)$ once the LU decomposition is known.

$$A\vec{x} = LU\vec{x} = L(U\vec{x}) = \vec{b}$$

Forward substitution $L\vec{y} = \vec{b}$

$$y_0 = \frac{b_0}{L_{00}}, \qquad y_i = \frac{1}{L_{ii}}\left[b_i - \sum_{j=0}^{i-1} L_{ij} y_j\right], \quad i = 1, ..., n-1$$

Backsubstitution $U\vec{x} = \vec{y}$

$$x_{n-1} = \frac{y_{n-1}}{U_{n-1,n-1}}, \qquad x_i = \frac{1}{U_{ii}}\left[y_i - \sum_{j=i+1}^{n-1} U_{ij} x_j\right], \quad i = n-2, ..., 0$$

Our bible

Chap.2.3.

http://numerical.recipes/book/book.html

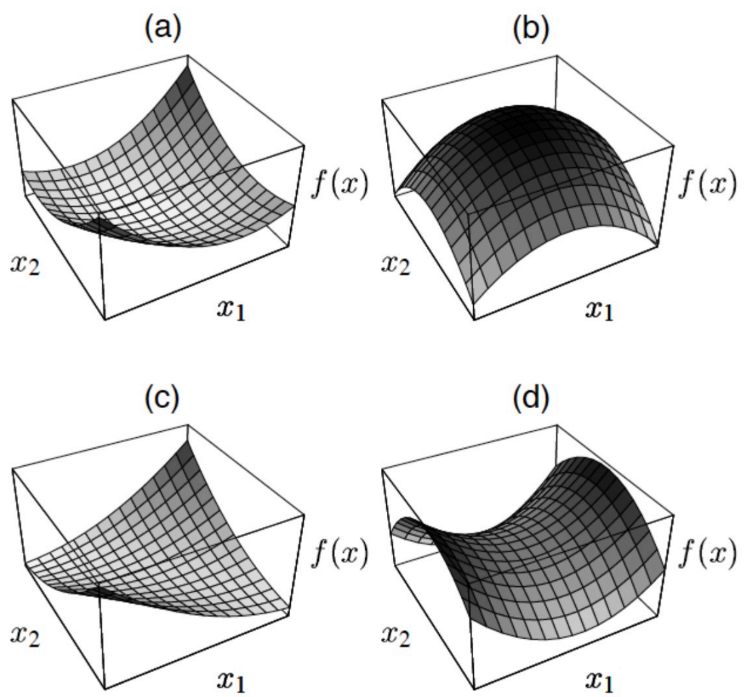
Steepest descent and Conjugate Gradient methods

$$f(x) = \frac{1}{2}x^T A x - b^T x + c \qquad \nabla_x f(x) = Ax - b = 0 \qquad Ax = b$$

$$A = \frac{1}{M}X^T X \quad \mathbb{R}^{(N+1)\times(N+1)}$$

$$b = X^T Y \quad \mathbb{R}^{(N+1)\times 1}$$

$$A = \begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \quad b = \begin{bmatrix} 2 \\ -8 \end{bmatrix} \qquad c = 0$$

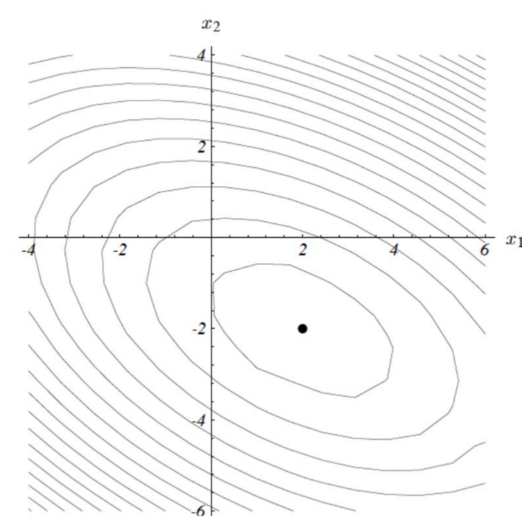


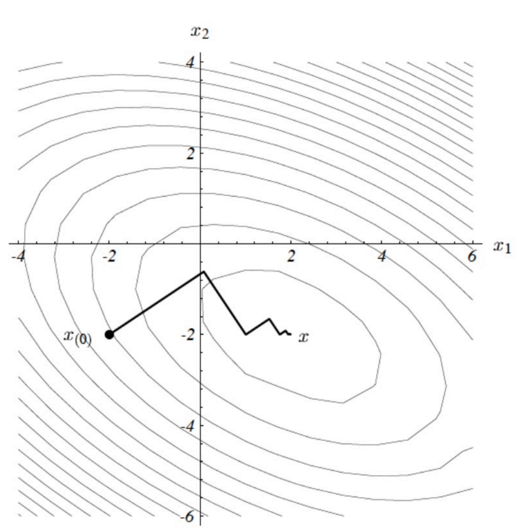
(a) positive-definite, symmetric

(b) negative-definite

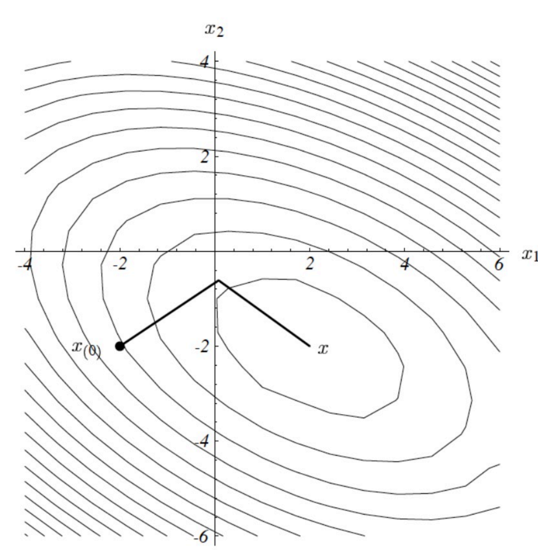
(c) singular, a set of solution

(d) saddle point





Many steps to find the solution



Only takes N-steps to find the solution

Remember what we learnt in the machine learning lecture ?

https://quantummc.xyz/wp-content/uploads/2023/03/lecture1_1.pdf

Iteration times $O(N)$

Our bible

Chap.10.8.

http://numerical.recipes/book/book.html

Parabolic PDEs

Time-dependent PDEs: diffusion equation

$$\frac{\partial u(t,\vec{r})}{\partial t} - D\Delta u(t,\vec{r}) = S(t,\vec{r})$$

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2}$$

$$t_n = n\cdot\tau,\quad n = 0,1,\cdots$$

$$\vec{r} = r\cdot h,\quad r = 0,1,\cdots,N-1$$

FTCS: Forward time (FT) Centered space (CS) scheme

$$\frac{u(n+1,r) - u(n,r)}{\tau} = D\frac{u(n,r+1) - 2u(n,r) + u(n,r-1)}{h^2}$$

Also called Explicit Euler method

$$u(n+1,r) = u(n,r) + \frac{D\tau}{h^2}(u(n,r+1) - 2u(n,r) + u(n,r-1))$$

Gaussian wave package as a solution

(can you see this ?)

$$u(t,r) = \frac{1}{\sqrt{2\pi}\sigma(t)}\exp[-\frac{r^2}{2\sigma(t)^2}] \qquad \sigma(t) = \sqrt{2Dt}$$

Characteristic diffusion-time $t_h = \dfrac{h^2}{2D}$ Width σ Increase from 0 to h

Stability $\tau << t_h$

Von Neumann Stability Analysis

Plane wave ansatz $u(n,r) = A^n e^{ikrh}$ $u(n+1,r) = u(n,r) + \frac{D\tau}{h^2}(u(n,r+1) - 2u(n,r) + u(n,r-1))$

$$AA^n e^{ikrh} = A^n e^{ikrh} + \frac{D\tau}{h^2}(A^n e^{ikrh} e^{ikh} - 2A^n e^{ikrh} + A^n e^{ikrh} e^{-ikh})$$

$$A = 1 + \frac{D\tau}{h^2}(e^{ikh} + e^{-ikh} - 2)$$

$$= 1 - 2\frac{D\tau}{h^2}(1 - \cos(kh))$$

$$= 1 - \frac{4D\tau}{h^2}\sin^2(\frac{kh}{2})$$

decrease h to h/2
decrease tau to tau/4

$d = 1$ $|A| \leq 1$ ⟶ $\frac{4D\tau}{h^2} \leq 2$ ⟶ $\tau \leq \frac{h^2}{2D}$ Higher dimension d $\tau \leq \frac{h^2}{2dD}$

With maximum $\tau = \frac{h^2}{2D}$ FTCS becomes $u(n+1,r) = \frac{1}{2}(u(n,r+1) + u(n,r-1))$

Similar with the Jacobi relaxation for Poisson equation when $d = 1$ and $\rho = 0$

https://en.wikipedia.org/wiki/Von_Neumann_stability_analysis

United Nations | **UN News**
Global perspective Human stories

Search
Advanced Search

Home | Topics | In depth | Secretary-General | Media

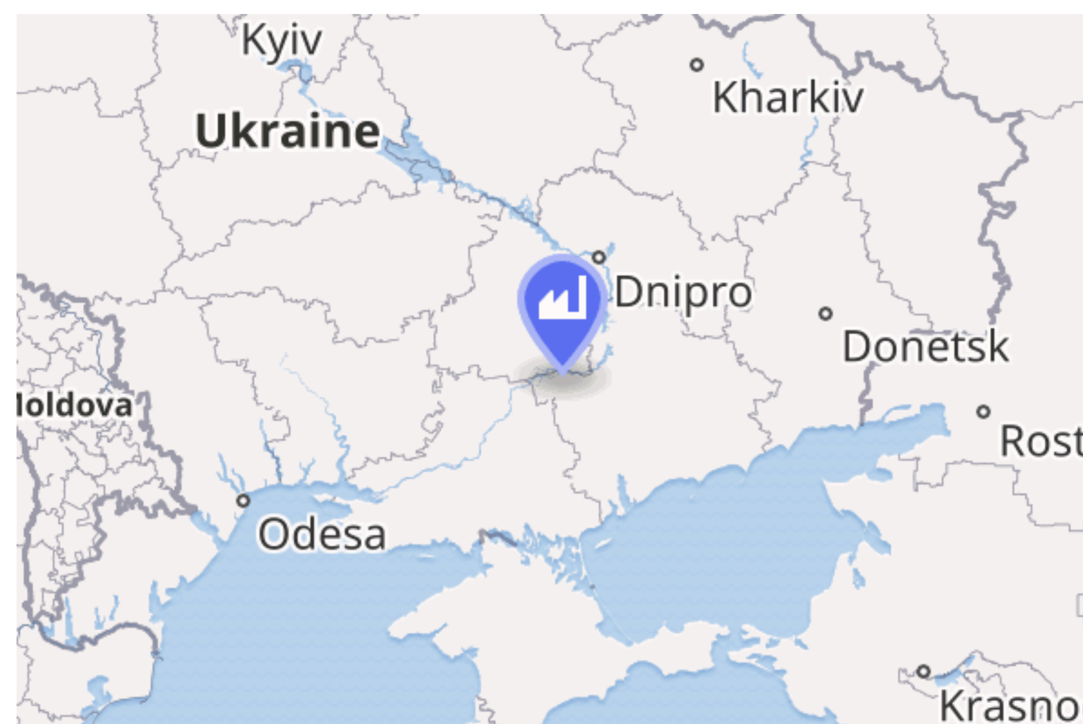
AUDIO HUB SUBSCRIBE

Ukraine: Zaporizhzhya nuclear plant initiates reactor shutdown following water leak, reports IAEA

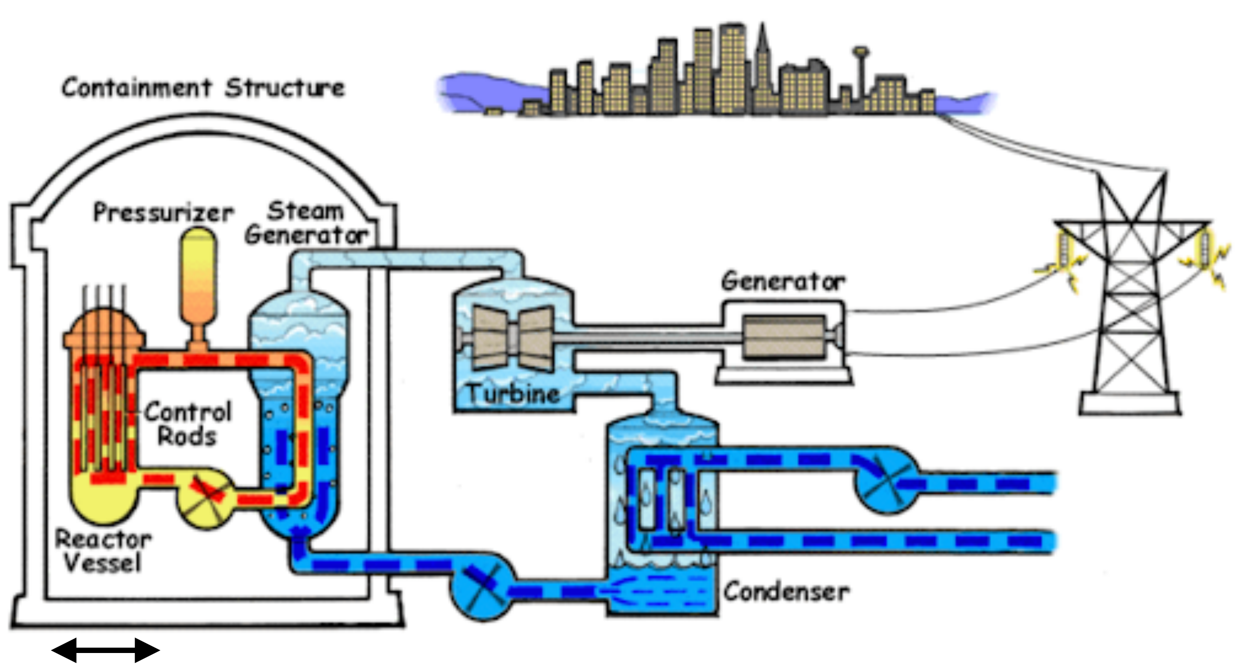
© **IAEA/Fredrik Dahl** | An IAEA expert mission team walks around the Zaporizhzhya Nuclear Power Plant and its surrounding area. (file)

10 August 2023 | Peace and Security

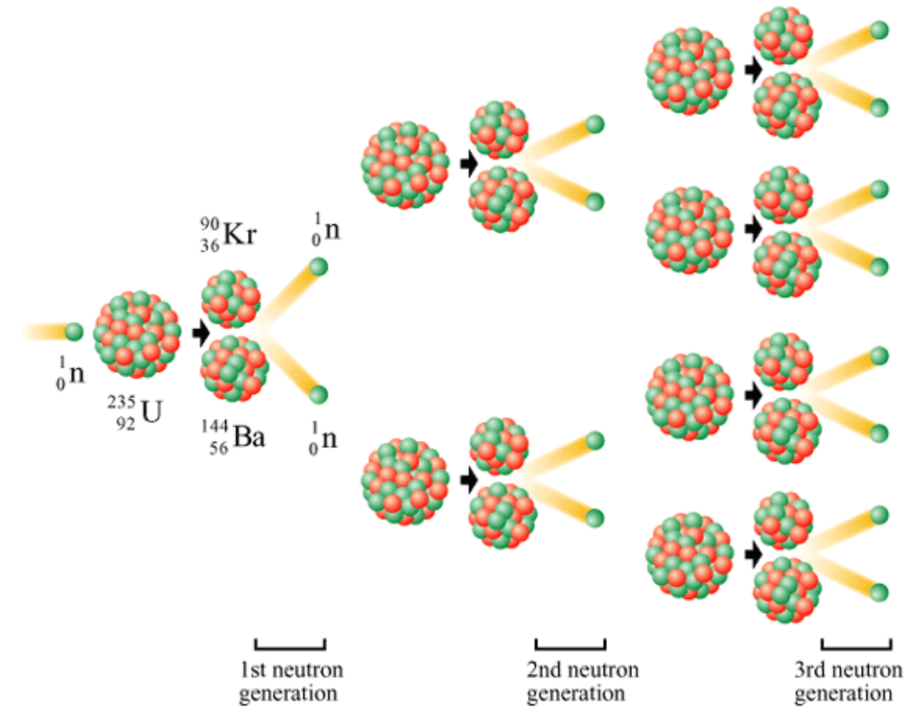
The **Zaporizhzhia Nuclear Power Station** (Ukrainian: Запорізька атомна електростанція, romanized: *Zaporiz'ka atomna elektrostantsiia*) in southeastern Ukraine is the largest nuclear power plant in Europe and among the 10 largest in the world.



Diffusion of neutrons from the chain reaction of Uranium



L



$$^{235}U + n \rightarrow {}^{139}Ba + {}^{95}Kr + 2n$$

For ^{235}U Diffusion constant $D \approx 10^5 m^2/s$ Creation rate $C \approx 10^8 s^{-1}$

$$\frac{\partial}{\partial t} n(t, \vec{r}) = D \Delta n(t, \vec{r}) + C n(t, \vec{r})$$

Dirichlet BC $n(\partial V) = 0 \quad V = L^d$

Critical length $L_c = \pi \sqrt{\frac{dD}{C}}$

FTCS
$$n(n+1, r) = (1 + C\tau) n(n, r) + \frac{D\tau}{h^2}(u(n, r+1) - 2u(n, r) + u(n, r-1))$$

Stability $\tau \leq \frac{h^2}{2dD} = \frac{h^2}{4D} \quad d = 2$

$$D = C = 1, L_c = \sqrt{2}\pi$$