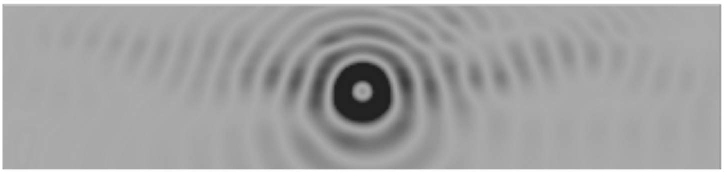
Advanced Solvers for Numerical Partial Differential Equations (PDEs)



Lecture 4

Nikolay Yavich
n.yavich@skoltech.ru

Part 1

Indefinite Helmholtz Equation

2

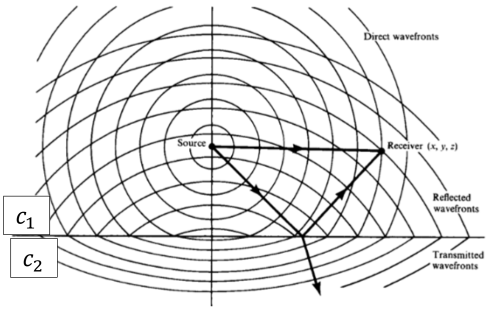
Physical Grounds

Helmholtz equation

$$-\Delta u - \frac{\omega^2}{c^2} u = f \text{ in } \Omega$$

+ boundary conditions (BC) on Γ

$f(x, y)$ time-harmonic source
$\omega = 2\pi\nu$ angular frequency
$c(x, y)$ speed of sound
$u(x, y)$ acoustic pressure



Aki & Richards, 2009

- Acoustic wave distribution
- Temporal Fourier transform of wave equation
- Propagation of energy rather than dissipation (as in Poisson and diffusion eq-s)

$k = \frac{\omega}{c}$ wavenumber
$\frac{c}{\nu}$ wavelength

3

Numerical Dispersion in 1D

General Solution
$-u'' - k^2 u = 0$ $\qquad u(x) = Ae^{ikx} + Be^{-ikx}$

uniform grid, $x_j = jh$
FD method

$u(x_j) = Ae^{ikjh} + Be^{-ikjh}$

$$-\frac{u_{j-1} - 2u_j + u_{j+1}}{h^2} - k^2 u_j = 0$$

$$-u_{j-1} + (2 - k^2h^2)u_j - u_{j+1} = 0$$

Exercise
Solve the recurrence equation assuming $0 < kh < 2$

$$\gamma = 2\arcsin(\frac{kh}{2})$$
$$u_j = Ae^{i\gamma j} + Be^{-i\gamma j}$$

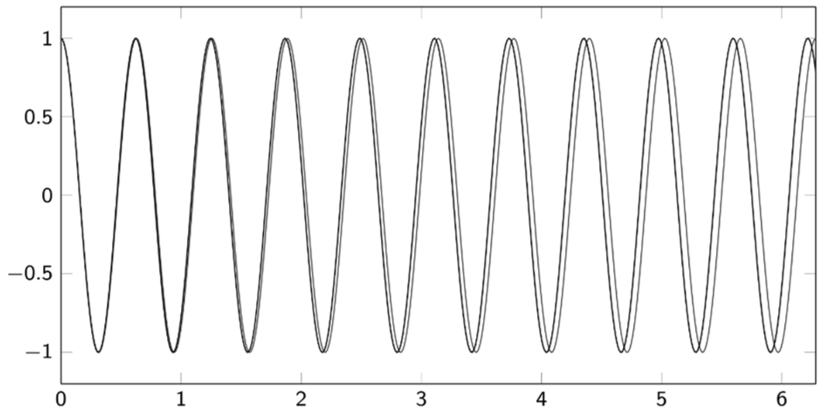
Equating the exponents, $i\gamma j = ik_h jh$

$$k_h = \frac{2}{h}\arcsin(\frac{kh}{2})$$

- k_h controls velocity of the modelled waves
- h typically picked such that $kh = 0.03 \cdots 0.10$, i.e. 30 $\cdots$ 10 steps per wavelength

4

Numerical Dispersion in 1D



— Re $\mathbf{u}_{\text{exact}} = \cos(\kappa x)$; — Re $u_h = \cos(\kappa_h x)$

$\kappa = 10$, $h = 0.05$, $\kappa_h = \frac{2}{h}\arcsin(\frac{\kappa h}{2}) \sim 10.10721$.

- What if k varies?
- Problematic for multigrid

5

BC for Scattering Problems

Sommerfeld radiation condition (2D)
- ensures uniqueness of the solution
- scattering of an incoming wave only produces outgoing not incoming waves from infinity

$$\lim_{r\to\infty} \sqrt{r}\left(\frac{\partial u}{\partial r} - iku\right) = 0.$$

Bounded domains used in numerical modeling…

1. First-order and higher-order non-reflecting boundary condition,

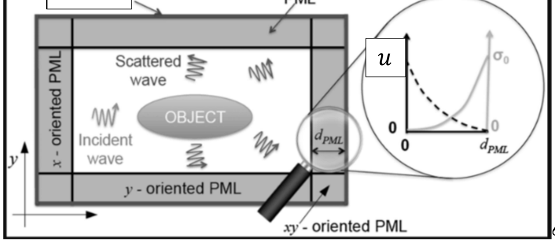
$$\frac{\partial u}{\partial v} - iku = 0$$

2. Perfectly matched layers

$$\frac{\partial^2 u}{\partial x^2} \to \frac{1}{S_x(x)}\frac{\partial}{\partial x}\left(\frac{1}{S_x(x)}\frac{\partial u}{\partial x}\right)$$

$$S_x(x) = 1 - \frac{i\alpha}{\omega}\frac{(x - x_0)^2}{\delta^2}$$

$u = 0$



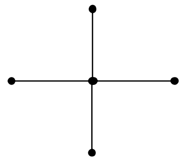
6

Sankaran,2019

System Matrix Properties

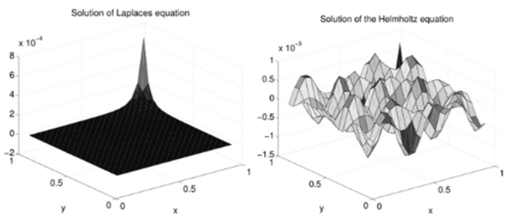
$$-\Delta u - k^2 u = f$$
$$u = 0$$

$$A_h u_h = f_h$$



- The matrix is sparse and symmetric
- For small k , the matrix is still positive-definite
- For larger k / higher frequencies, the matrix becomes indefinite

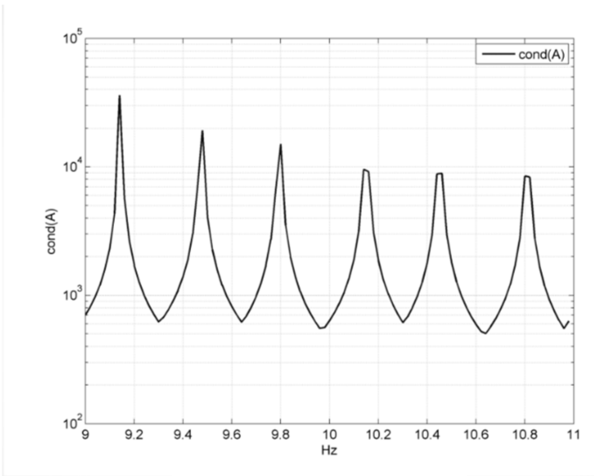
$$k = \frac{\omega}{c}$$



Solution due to point source

7

Condition number



8

Solvers for Indefinite Matrices

For an indefinite A_h the CG is not applicable.
We can multiply the system by the adjoint,

$$A_h^* A_h u_h = A_h^* f_h$$

and apply the CG to the normal system

Exercise: Estimate $\mathrm{cond}(A_h^* A_h)$ via $\mathrm{cond}(A_h)$

GMRes (Generalized minimum residual method)
- Convergence for any nonsingular matrix
- Memory consuming – typically stores 20 extra vectors

BiCGStab (Bi-conjugate gradient stabilized)
- No convergence theorem / not smooth
- Memory economical

QMR (quasi-minimal residual method)
- Smother convergence than of BiCGStab
- Somewhat slower than BiCGStab

9

Standard Multigrid Performance

Multigrid iterations

GMRes + multigrid

k	Smoothing steps	2.5π	5π	10π	20π
Iterative	$\nu = 2$	7	div	div	div
Preconditioner	$\nu = 2$	6	12	41	127
Iterative	$\nu = 5$	7	stag	div	div
Preconditioner	$\nu = 5$	5	13	41	223
Iterative	$\nu = 10$	8	div	div	div
Preconditioner	$\nu = 10$	5	10	14	87

Ernst et al, 2012

* Helmholtz operator is not suitable to be solved with standard multigrid.

10

Approximate LU Factorization

$A \approx L \cdot U$

- ILU(0) an approximate LU factorization, retaining entries in the L and U factors only if the corresponding entry in the underlying matrix A is non-zero

- ILU(tol) elements in L and U are kept, provided they are bigger than the tolerance tol.

number of grid points per wavelength constant, i.e. kh = 10

	QMR		ILU('0')		ILU(1e-2)	
k	it	Mflops	it	Mflops	it	Mflops
5	197	120.1	60	60.4	22	28.3
10	737	1858.2	370	1489.3	80	421.4
15	1775	10185.2	> 2000	> 18133.2	220	2615.1
20	> 2000	> 20335.1	—	—	> 2000	> 42320.1

Ernst et al, 2012

ILU preconditioners are quite effective for small wave numbers, but their performance deteriorates when k becomes larger

11

Shifted-Laplacian Preconditioner

$$-\Delta u - k^2 u = f \qquad u = 0$$

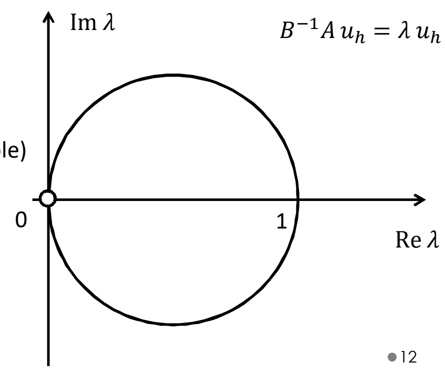
$$A\, u_h = f_h$$

$$-\Delta u - k^2(1 - i\beta)u = f \qquad u = 0$$

B $\quad\beta$ strong artificial attenuation

$$B^{-1} A\, u_h = \lambda\, u_h$$

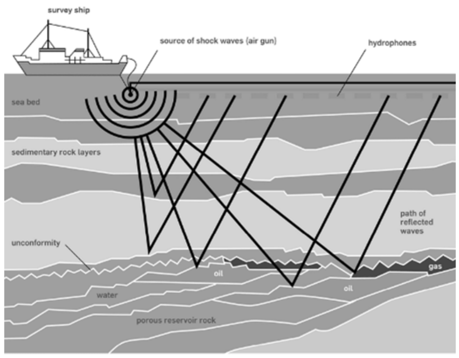
- Let us use B as a preconditioner to A
- Invert B with multigrid or FFT (if applicable)



12

Marine Oil/gas Exploration

thelundingroup.com



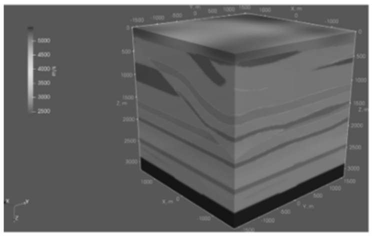
N. Kukreja

- Seismic waves are generated by an airgun and recorded by hydrophones
- With some approximation, seismic wave propagation can be modelled with Helmholtz eq-n

13

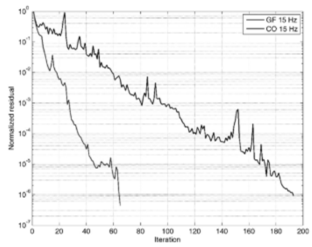
Example

Complex heterogeneous earth model used to test geophysical modeling software



Shifted-laplacian preconditioner combined with FFT (GF) and extra scaling (CO)

Frequency (Hz)	Step size (m)	Internal grid	Overall discrete problem size	Peak memory usage	GF solver Iteration count, CPU time (min)	CO solver Iteration count, CPU time (min)
10	15	210 × 210 × 130	12 M	4.5 Gb	81, 56 min	62, 43.6 min
15	10	320 × 320 × 200	33 M	13 Gb	192, 6.1 h	64, 2.1 h
20	8	400 × 400 × 250	59 M	24 Gb	n/a	71, 5.0 h
30	5	640 × 640 × 400	211 M	86 Gb	n/a	98, 27.6 h



Yavich et al, 2021

14

Resumé on Helmholtz Eq-n

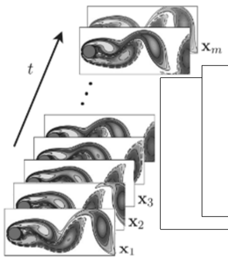
- Standard iterative solvers converge very slow/diverge for HE
- Solvers components has to be chosen differently
- Designing an optimal solver with performance independent of the wave number k is still open and important equation in Numerical PDEs

Part 2

Dynamic Mode Decomposition (DMD)

In collaboration with G.Ovchinnikov

Time Series Examples



Kutz et al, 2016

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{m-1} \\ | & | & & | \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} \cdots & \vdots & \cdots \\ \cdots & \text{'JPM'} & \cdots \\ \cdots & \text{'BOA'} & \cdots \\ \cdots & \text{'AMZN'} & \cdots \\ \cdots & \text{'CVX'} & \cdots \\ \cdots & \vdots & \cdots \end{bmatrix} \downarrow \text{companies}$$

$\longrightarrow$
daily stock prices

17

Linear Autonomous ODE System

$$\frac{dx}{dt} = \mathcal{A}x \qquad t = 0, \tau, \cdots m\tau \qquad x(t) \in R^n$$
$$x(0) = x_0 \qquad \mathcal{A} \in R^{n \times n}$$

Linear dynamical system $\qquad n \gg 1$

$$x(t) = \exp(\mathcal{A}t) \cdot x_0 \qquad x_k = x(k\tau)$$

$$x_{k+1} = \underbrace{\exp(\mathcal{A}\tau)}_{A} \cdot x_k \qquad x_{k+1} = A \cdot x_k$$

How to find the system (A or $\mathcal{A}$) from the given data?

18

DMD Idea

$x_k \in R^n$ measurements vector / snapshot, $k = 1 \cdots m$

$X = [x_1\ x_2 \cdots \mathrm{x}_{m-1}]\ \in R^{n \times m}$

$X' = [x_2\ \cdots\ x_{m-1} x_m]\ \in R^{n \times m}$ Note: $x_{k+1} = A \cdot x_k$

$X' = A \cdot X$

$\boxed{A = X' \cdot X^+}$

X^+ pseudo-inverse
Least-squares fit
Exact DMD

- Computing A explicitly would require to much CPU time/memory
- Knowing its eigenvalues/vectors would sufficient
- Since this is an estimate anyways, we might be happy with low-rank approximation

19

DMD Algorithm

Step 1. $X = U\,\Sigma\,V^*$ Reduced SVD, r truncation rank

Step 2. $A = X' \cdot X^+ = X'V\,\Sigma^{-1}\,U^*$
$\tilde{A} = U^* A U \in R^{r \times r}$
$\tilde{A} = U^* X' V\,\Sigma^{-1}$

U low-rank embedding space
$\tilde{A}$ low-dimensional linear model

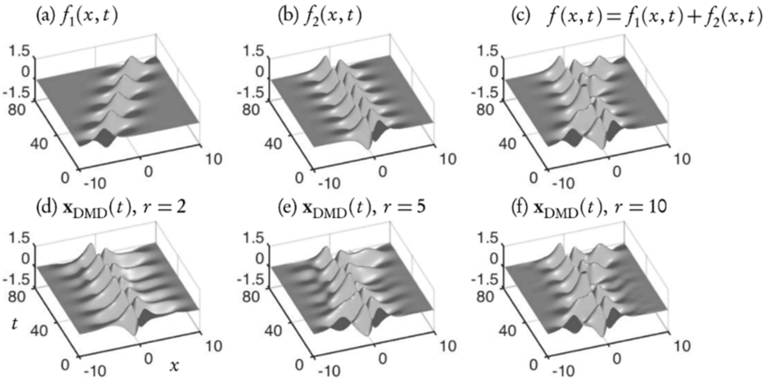
Step 3. $\tilde{A} = \mathrm{W}\Omega W^{-1}$ $\Omega = \mathrm{diag}(\omega_1 \cdots \omega_r)$ DMD eigenvalues

Step 4. $\Phi = U \cdot W$ DMD modes/eigenvectors
$x_k = \Phi \exp(\Omega k \tau)\, b$
$b = \Phi^{-1} \mathrm{x}_0$

- Very universal
- Short-time window into the future
- Problem if $\mathrm{Re}\,\omega_k > 0$

20

Examples



Kutz, 2016

21

Another look onto DMD

$X = [x_1\ x_2 \cdots \mathrm{x}_{m-1}] \ \in R^{n\times m}$

$X' = A \cdot X$

$X' = [x_2 \ \cdots \ x_{m-1} x_m] \ \in R^{n\times m}$

$c \in R^n$

$X' = X \cdot C$

$C = X^{+} \cdot A \cdot X$

$$C = \begin{pmatrix} 0 & 0 & 0 & & c_1 \\ 1 & 0 & 0 & & c_2 \\ 0 & 1 & 0 & & c_3 \\ & & \cdots & & \\ & & & 1 & c_n \end{pmatrix}$$

Companion matrix, upper Hessenberg

$x_m = X \cdot c$

$\boxed{c = R^{-1} Q^{*} x_m}$

$X = QR$ economy−size QR−decomposition

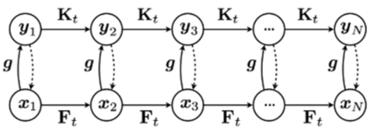
- not capable of extracting more than the first or first two dominant dynamic modes.
- no orthogonalization step is taken (as opposed to Arnoldi) instead, a smaller system matrix is generated from the snapshots directly.

22

Limitations

- Translational and/or rotational invariances of low-rank objects embedded in the data are not well captured.
- Transient phenomena difficult to predict
- What the system is non-autonomous?

Relation to Koopman operator



$$x_{k+1} = F(x_k) = Ax_k + \cdots$$

$y_k = g(x_k)$ – observables, e.g. $g(x_k) = x_k$

$$\boxed{Kg(x_k) = g(F(x_k)} \quad = g(x_{k+1})$$

Notice
- K is linear and infinite-dimensional
- If we figured K or its spectral decomposition (which is the Holy Grail of dynamical systems), we can advance observables avoiding non-linearity

$$K\phi_s(x) = \lambda_s \phi_s(x), \; s = 1,2, \cdots$$

$$g(x) = \sum_{s=1}^{\infty} v_s \phi_s(x)$$

$$g(x_{k+1}) = Kg(x_k) = \sum_{s=1}^{\infty} v_s \lambda_s \phi_s(x_k)$$

Example:
$F(x_k) = Ax_k$
$g(x_k) = x_k$ $\Longrightarrow$ $K = A$, i.e. the DMD estimates the Koopman operator

Summary

- Equation-free decomposition method
- Easy to code ☺
  Available in packages, e.g. `PyDMD`
- Provides reconstructions of spatiotemporal structures arising in nonlinear dynamical systems
- Allows to make predictions
- Many other algorithms based on this idea