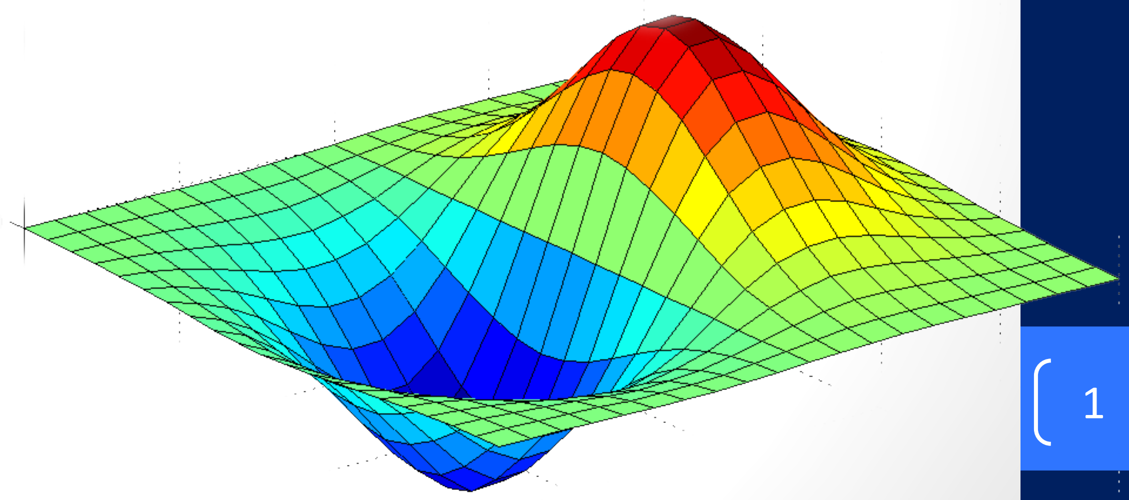
Frequency-dependent attenuation in fractional Helmholtz wave equations

Progress Report

By: Samuel Rouah-Sayag
Advisor: Prof. Michael Berman



Content

- What is medical ultra-sound
- The Physics of the problem
- Available methods to solve the problem numerically
- Study of the Finite Difference Method
- Exact Finite Difference Scheme for the Helmholtz equation
- Higher Order Schemes
- Conclusion
  - Progress in the project
  - Next Steps

Medical ultrasound imaging

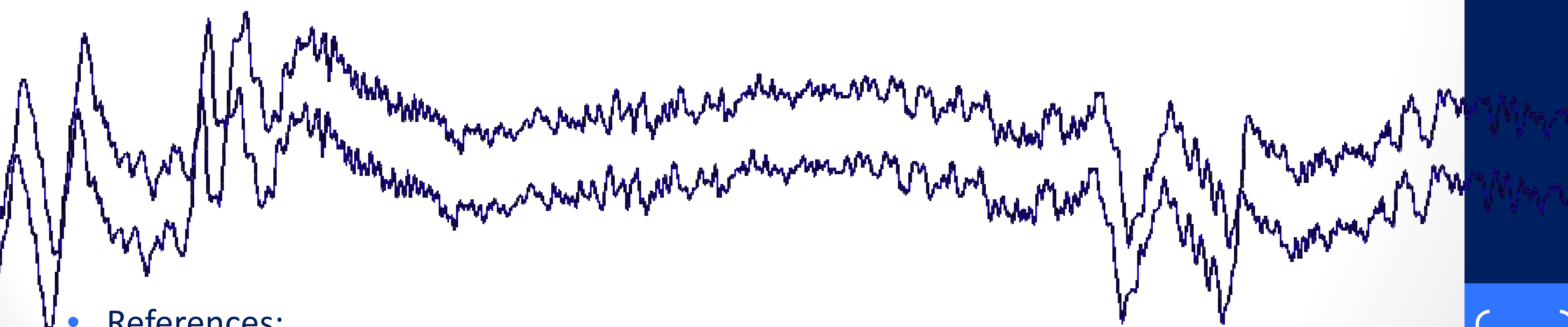
- Modes (A, B, C, M, Doppler, 3D, Tomography…)

- B-Mode (Brightness mode)
  - back-reflection only

2D (B-Mode) and 3D picture
(Mirpaha Meuhedet, Jerusalem, 2014)

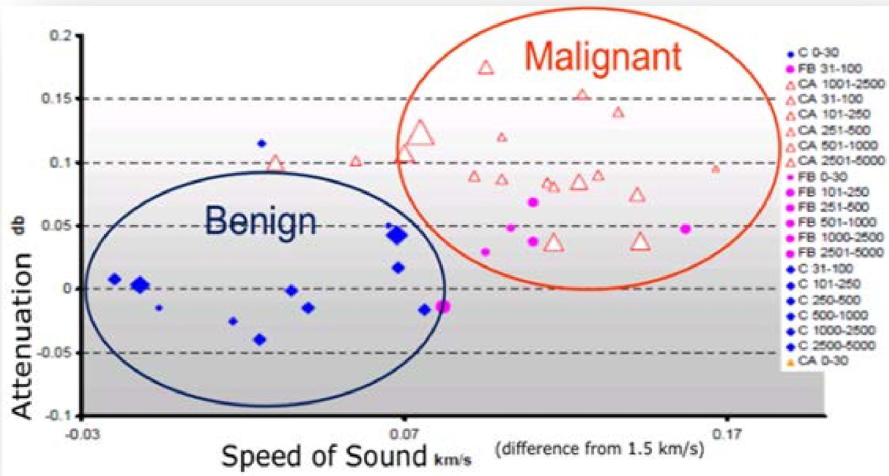
- Tomography
  - Transmission & scattering in many directions
  - Enables the measurement of speed of sound and attenuation

The physics of the problem



- References:
- Olof Runborg, "Helmholtz Equation and High Frequency Approximations," in http://www.csc.kth.se/utbildning/kth/kurser/DN2255/ndiff12/.
- Burden R L. and Faire J D, ***Numerical Analysis***, 9th ed.: Brooks/Cole, 2010.

Speed of sound and power law attenuation in Tomography



Breast masses as a function of **attenuation** and **speed of sound**

- Resolution ⇔ *high f*
- Depth ⇔ *low f*
- *Resolution* and *depth* ⇔ a trade off
- Speed of sound & Attenuation - *not* measureable in B-Mode

f = frequency

Speed of sound and power law attenuation

$$\alpha(\mathbf{x},\omega) = \alpha(\mathbf{x})|\omega|^{\gamma(\mathbf{x})}$$

- Attenuation coefficient $\alpha(\mathbf{x},\omega)$, where:
  - $\mathbf{x}$ a vector position (1D, 2D, 3D)
  - ω is the angular frequency ($\omega = 2\pi f$)
  - $\alpha(\mathbf{x})$ and $\gamma(\mathbf{x})$ are non negative location dependent parameters.
- If the material is homogeneous $\alpha(\mathbf{x}) = \alpha_0$
- γ typically ranges between 1-2

Fractional wave equation

$$\nabla^2 p(\mathbf{x},t) - \frac{1}{c^2(\mathbf{x})}\frac{\partial^2 p(\mathbf{x},t)}{\partial t^2} - \frac{2\alpha(\mathbf{x})}{c(\mathbf{x})\cos\left(\frac{\pi\gamma(\mathbf{x})}{2}\right)}\frac{\partial^{\gamma(\mathbf{x})+1} p(\mathbf{x},t)}{\partial t^{\gamma(\mathbf{x})+1}} - \frac{\alpha^2(\mathbf{x})}{\cos^2\left(\frac{\pi\gamma(\mathbf{x})}{2}\right)}\frac{\partial^{2\gamma(\mathbf{x})} p(\mathbf{x},t)}{\partial t^{2\gamma(\mathbf{x})}} = 0$$

- Describes a pressure wave
- $p(\mathbf{x},t)$ - pressure as a function of location $\mathbf{x}$ and time t.
- $c(\mathbf{x})$ - speed of sound.
- $\alpha(\mathbf{x})$ - attenuation.
- $\gamma(\mathbf{x})$ -exponent parameter.

$for\ \alpha(\mathbf{x})$ = 0: $\nabla^2 p(\mathbf{x},t) - \frac{1}{c^2(\mathbf{x})}\frac{\partial^2 p(\mathbf{x},t)}{\partial t^2} = 0$ (standard wave equation)

Fractional wave equation in the frequency domain

- Fractional derivative equations are rather complicated numerically.
- We change the *space-time* domain to *space-frequency* domain by Fourier transformation

$$\hat{p}(\mathbf{x},\omega)=\int_{-\infty}^{\infty}p(\mathbf{x},t)\,e^{i\omega t}dt$$

- $\hat{p}(\mathbf{x},\omega)$ - location and frequency dependent pressure wave.
- ω - angular frequency ($\omega=2\pi f$)

Helmholtz equation

$$\nabla^2 \hat{p}(\mathbf{x},\omega) + n^2(\mathbf{x},\omega)\omega^2 \hat{p}(\mathbf{x},\omega) = 0, \qquad \mathbf{x} \in \mathbb{R}^d$$

Where the refraction index $n\,(\mathbf{x},\omega)$ is defined through:

$$n^2(\mathbf{x},\omega) \equiv \frac{1}{c^2(\mathbf{x})} - \frac{2\alpha(\mathbf{x})}{c(\mathbf{x})\, cos\left(\frac{\pi\gamma(\mathbf{x})}{2}\right)}(-i)^{\gamma(\mathbf{x})+1}\omega^{\gamma(\mathbf{x})-1} - \frac{\alpha^2(\mathbf{x})}{cos^2\left(\frac{\pi\gamma(\mathbf{x})}{2}\right)}(-i)^{2\gamma(\mathbf{x})}\omega^{2(\gamma(\mathbf{x})-1)}$$

- The time derivative terms are now powers of $(-\mathrm{i}\omega)$.
- For $\alpha(\mathbf{x}) = 0 \Rightarrow \nabla^2 \hat{p}(\mathbf{x},\omega) + \frac{\omega^2}{c^2(\mathbf{x})}\hat{p}(\mathbf{x},\omega) = 0,$

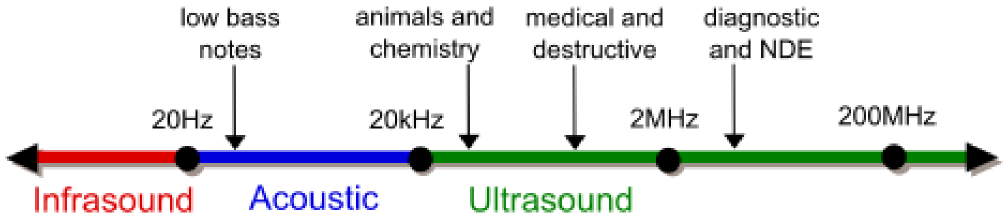
  the standard Helmholtz eq.

Helmholtz equation revisited

- Postulate a solution of the form: $\hat{p}(\mathbf{x},\omega) = A_\omega(\mathbf{x})e^{i\omega T_\omega(\mathbf{x})}$
- A_ω is the *amplitude,* T_ω is the *travel time*
- Develop and obtain the two way coupled equations:

$$-A_\omega(\mathbf{x})\nabla T_\omega(\mathbf{x})\cdot\nabla T_\omega(\mathbf{x}) + \frac{\nabla^2 A_\omega(\mathbf{x})}{\omega^2} + Re[\,n^2(\mathbf{x},\omega)]A_\omega(\mathbf{x}) = 0$$

$$A_\omega(\mathbf{x})\nabla^2 T_\omega(\mathbf{x}) + 2\nabla A_\omega(\mathbf{x})\cdot\nabla T_\omega(\mathbf{x}) + Im\,[n^2(\mathbf{x},\omega)]\omega A_\omega(\mathbf{x}) = 0$$



- High frequency ⇔ *highly* oscillating system, the ray approximation
- *low to medium frequency ⇔ full wave description*

Highly oscillating system

- Approx. $\cancel{\frac{\nabla^2 A_\omega(\mathbf{x})}{\omega^2}}$ at high-frequency (ω is large)

$\Rightarrow$one way coupled equations:

$$|\nabla T_\omega(\mathbf{x})|^2 = Re[\,n^2(\mathbf{x},\omega)]\ \ \textbf{(Eikonal)}$$

$$2\nabla A_\omega(\mathbf{x})\cdot\nabla T_\omega(\mathbf{x}) + A_\omega(\mathbf{x})[Im\,[n^2(\mathbf{x},\omega)]\omega + \nabla^2 T_\omega(\mathbf{x})] = 0$$

(Stationary Transport Equation)

Low to medium oscillating system

$$-A_\omega(\mathbf{x})\nabla T_\omega(\mathbf{x})\cdot\nabla T_\omega(\mathbf{x})+\frac{\nabla^2 A_\omega(\mathbf{x})}{\omega^2}$$
$$+Re[\,n^2(\mathbf{x},\omega)]A_\omega(\mathbf{x})=0$$

$$A_\omega(\mathbf{x})\nabla^2 T_\omega(\mathbf{x})+2\nabla A_\omega(\mathbf{x})\cdot\nabla T_\omega(\mathbf{x})$$
$$+Im\,[n^2(\mathbf{x},\omega)]\omega A_\omega(\mathbf{x})=0$$

$\Leftrightarrow$

$$\nabla^2\hat{p}(\mathbf{x},\omega)+n^2(\mathbf{x},\omega)\omega^2\hat{p}(\mathbf{x},\omega)=\ 0$$
$$,\qquad\qquad \mathbf{x}\in\mathbb{R}^d$$

Boundary Value Closed Problem

- Dirichlet

$$u(\mathbf{x}) = \alpha; \quad \mathbf{x} \in \boldsymbol{\Gamma}$$

- Neumann

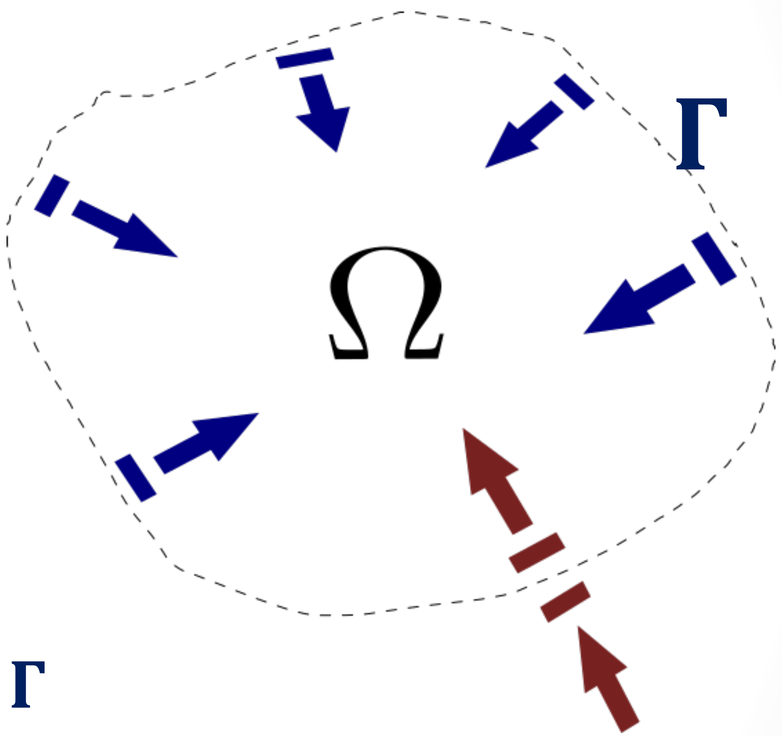
$$\frac{\partial u(\mathbf{x})}{\partial \mathrm{n}} = \alpha; \quad \mathbf{x} \in \boldsymbol{\Gamma}$$

- Cauchy

$$u(\mathbf{x}) = \alpha, \frac{\partial u(\mathbf{x})}{\partial \mathrm{n}} = \beta \; ; \; \mathbf{x} \in \boldsymbol{\Gamma}$$

- Robin

$$a\, u(\mathbf{x}) + b\, \frac{\partial u(\mathbf{x})}{\partial \mathrm{n}} = \alpha \; ; \quad \mathbf{x} \in \boldsymbol{\Gamma}$$



Boundary Value Open Problem

- Source is Dirichlet

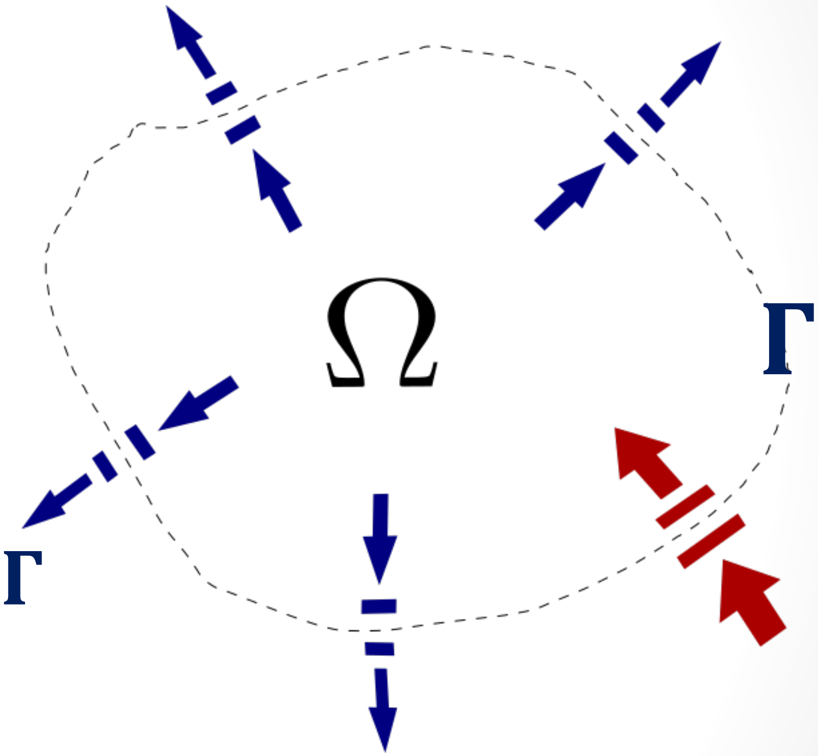
$$u(\mathbf{x}) = \alpha; \qquad \mathbf{x} \in \boldsymbol{\Gamma}$$

- Sommerfeld

$$\frac{\partial u(\mathbf{x})}{\partial \mathrm{n}} + i\beta u(\mathbf{x}) = 0; \quad \mathbf{x} \in \boldsymbol{\Gamma}$$

- Damping technique

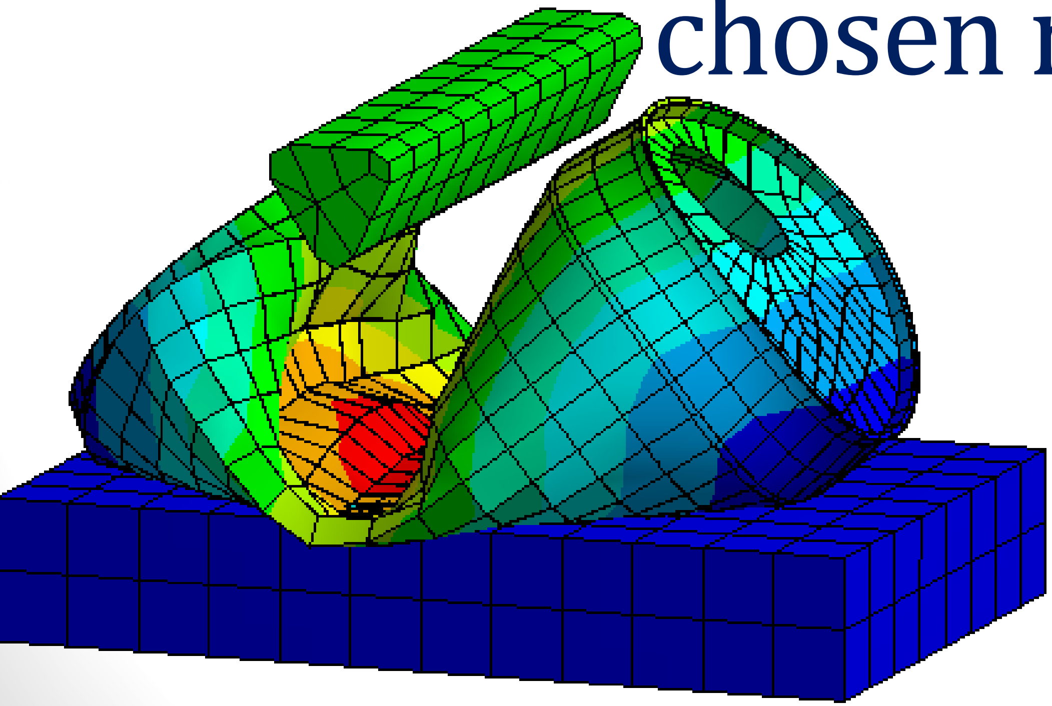
$$i\alpha\beta u(\mathbf{x}) + (\nabla + n)u = 0; \qquad \mathbf{x} \in \boldsymbol{\Gamma}$$



Project goals: solution for the low and intermediate frequency range

- Solve the ***forward problem***
- High frequency hypothesis: ray approximation to the wave front propagation
- **The ultimate goal of the project is the solution of the full-wave Helmholtz equation**
- Build a tool
- Conduct numerical simulations to evaluate the results

Available methods and the chosen method



Finite Differences Method (FDM)

We seek an *approximation* of the equation by "*replacing the derivatives in the differential equation by finite difference approximation*" (Leveques, 2007).

- For each approximation there exists an *error.*
- Formulas are found according to the desired order of precision thanks to Taylor theorem.
- Methods exist that are suitable for various types of partial differential equations : parabolic, hyperbolic, elliptic etc.

Other Approximation Methods

- Finite elements method (FEM)
- Spectral Method (SM)
- Multi-Grid
- Finite Volume

Study of the FDM (Finite Difference Method)

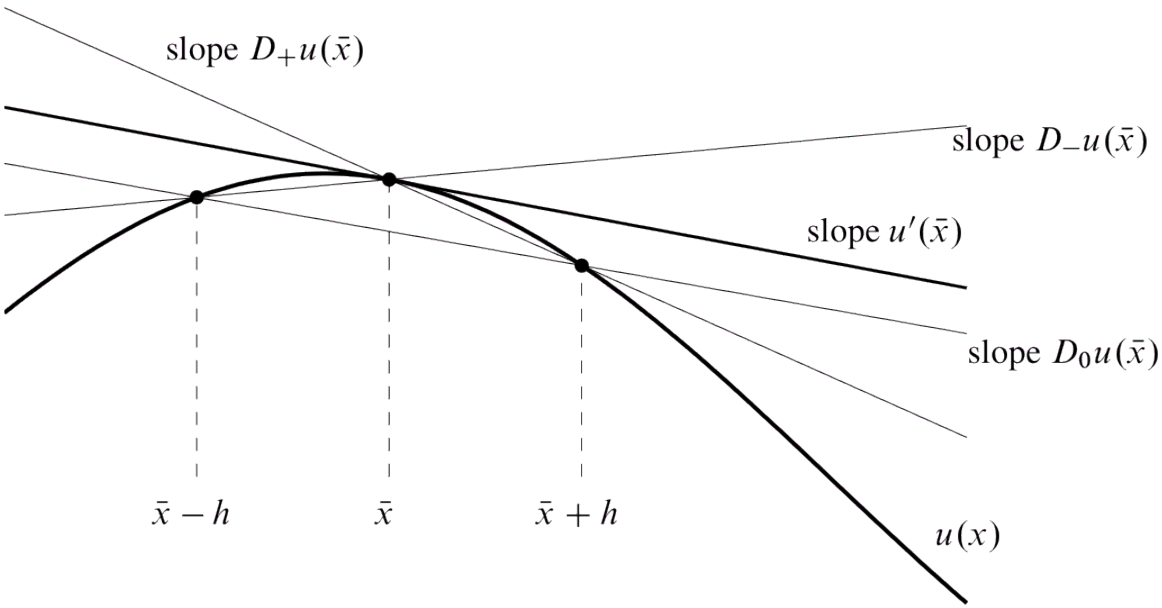
References:

LeVeque R J., ***Finite Difference Methods for Ordinary and Partial Differential Equations***. Philadelphia: Society for Industrial and Applied Mathematics (SIAM), 2007.

Burden R L. and Faire J D, ***Numerical Analysis***, 9th ed.: Brooks/Cole, 2010.

S C Chapra and R P Canale, ***Numerical Methods For Engineers***, Sixth Edition ed., Mac Graw Hill, Ed., 2010

From the derivative definition



$$\lim_{x \to 0} \frac{u(x+h) - u(x)}{h} = u'(x)$$

$$D_-u(\bar{x}) \equiv \frac{u(\bar{x}) - u(\bar{x}-h)}{h}$$

$$D_+u(\bar{x}) \equiv \frac{u(\bar{x}+h) - u(\bar{x})}{h}$$

$$D_0u(\bar{x}) \equiv \frac{u(\bar{x}+h) - u(\bar{x}-h)}{2h} = \frac{1}{2}(D_+u(\bar{x}) + D_-u(\bar{x}))$$

Illustration: Finite Difference Methods for Ordinary and Partial Differential Equations, Leveque, 2007.

From Taylor expansion

Taylor Theorem (1712 !)

$$u(a+h) = u(a) + \frac{u^{(1)}(a)}{1!}h + \frac{u^{(2)}(a)}{2!}h^2 + \frac{u^{(3)}(a)}{3!}h^3 + O(h^4) \qquad (1)$$

$$u(a-h) = u(a) - u^{(1)}(a)h + \frac{1}{2}u^{(2)}(a)h^2 - \frac{1}{6}u^{(3)}(a)h^3 + O(h^4) \qquad (2)$$

$$u(a-2h) = u(a) - u^{(1)}(a)2h + 2u^{(2)}(a)h^2 - \frac{4}{3}u^{(3)}(a)h^3 + O(h^4) \qquad (3)$$

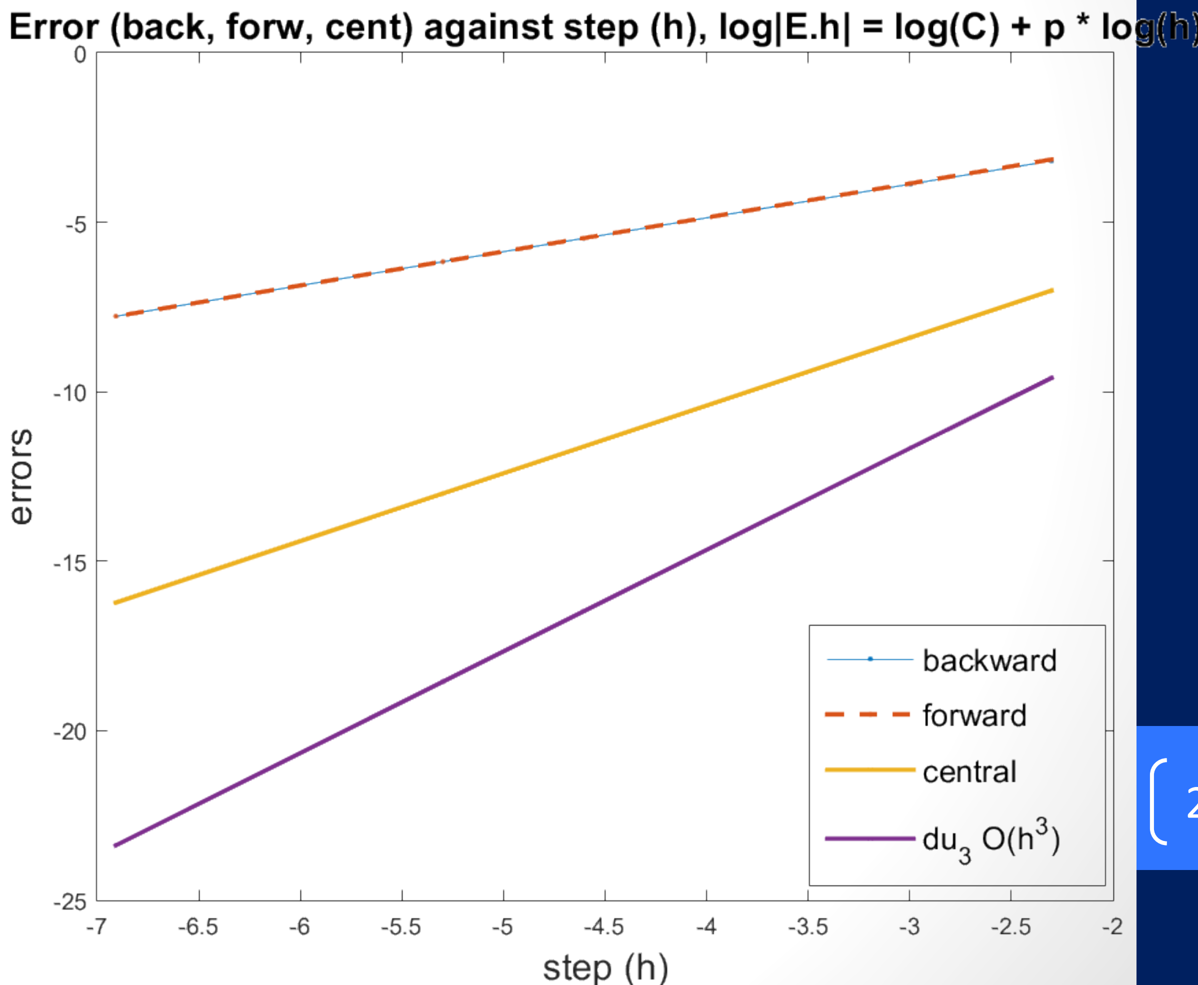
Derived	Scheme	Type	Order
(1)	$u^{(1)} \approx \frac{u(a+h)-u(a)}{h}$	Forward	$O(h)$
(2)	$u^{(1)} \approx \frac{u(a)-u(a-h)}{h}$	Backward	$O(h)$
(1)-(2)	$u^{(1)} \approx \frac{u(a+h)-u(a-h)}{2h}$	Central	$O(h^2)$
2(1)-6(2)+(3)	$u^{(1)} \approx \frac{1}{6h}[2u(x+h)+3u(x)-6u(x-h) + u(x-2h)]$	No name	$O(h^3)$
(1)+(2)	$u^{(2)} \approx \frac{u(a+h)-2u(a)+u(a+h)}{h^2}$	Symmetric	$O(h^2)$

Error

Example: $u(x) = \sin(x)$, and we want an approximation a $x = 1$ of the first derivative ($u'(1) = \cos(1) = 0.54030230586814$).

'h'	*'Back Err'*	*'Forw Err'*
[1.0000e-**01**]	[-4.1138e-**02**]	[4.2939e-**02**]
[5.0000e-**02**]	[-2.0807e-**02**]	[2.1257e-**02**]
[1.0000e-**02**]	[-4.1983e-**03**]	[4.2163e-**03**]
[5.0000e-**03**]	[-2.1014e-**03**]	[2.1059e-**03**]
[1.0000e-**03**]	[-4.2065e-**04**]	[4.2083e-**04**]

'Cent Err'	*'du_3_err'*
[9.0005e-**04**]	[-6.8207e-**05**]
[2.2510e-**04**]	[-8.6491e-**06**]
[9.0050e-**06**]	[-6.9941e-**08**]
[2.2513e-**06**]	[-8.7540e-**09**]
[9.0050e-**08**]	[-6.9979e-**11**]



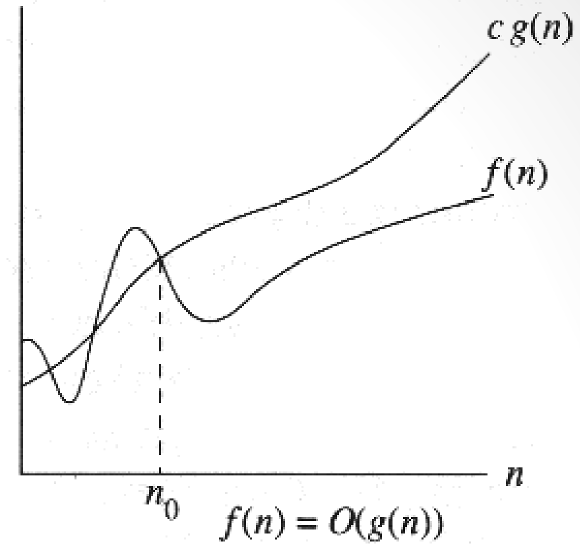
Errors: truncation

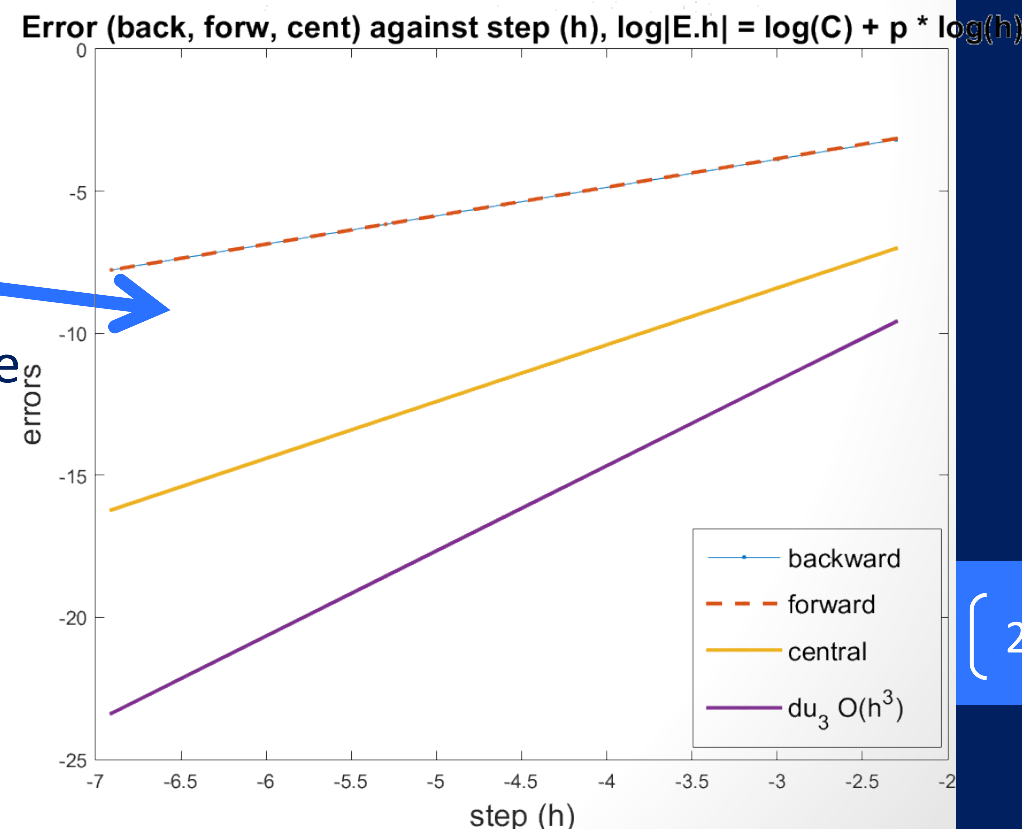
- Big-O

  Error $\in O(h^p) \Leftrightarrow \exists C, h_0\ \forall h < h_0, \dfrac{|error|}{|h^p|} < C$

- $Error(h) = Ch^p \Leftrightarrow \log\big(Err(h)\big) = \log(C) + p\log(h)$

- This may be a representation we can use to compare schemes
- Leveque (2007) demonstrates that the error of the scheme is the same as the cumulative error.





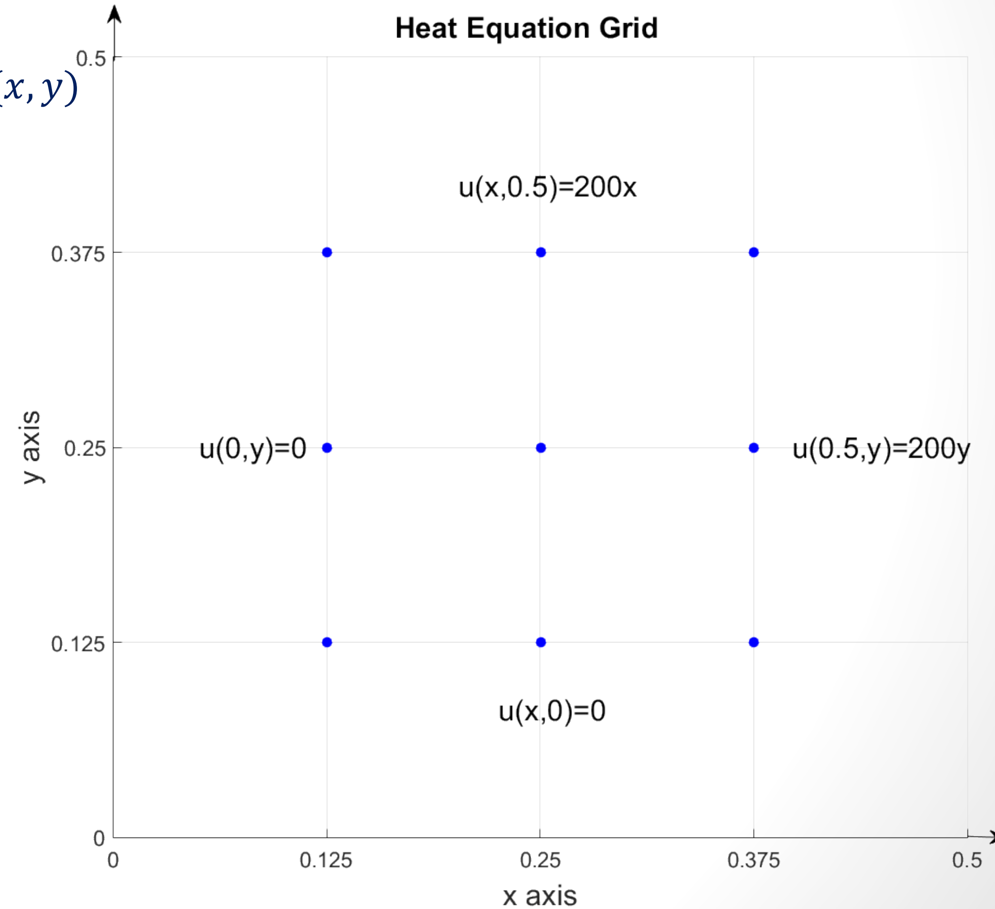
PDE Poisson – to test my code

- **Elliptic** PDE (stationary) so is Helmholtz $B^2 - 4AC < 0$
- $\Omega = [0.125, 0.375] \times [0.125\ 0.375]$ with $h = 0{,}125$

$$\begin{cases} \nabla^2 u(x,y) \equiv \dfrac{\partial^2 u(x,y)}{\partial x^2} + \dfrac{\partial^2 u(x,y)}{\partial y^2} = f(x,y) \\ u(x,y) = g(x,y) \end{cases}$$

$$\frac{u(x_{i+1},y_j) - 2u(x_i,y_j) + u(x_{i-1},y_j)}{h^2} + \frac{u(x_i,y_{j+1}) - 2u(x_i,y_j) + u(x_i,y_{j+1})}{h^2} = f(x_i,y_j) + \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4} + \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}$$

$$\begin{cases} 4u_{i,j} - u_{i+1,j} + u_{i-1,j} - u_{i,j+1} + u_{i,j-1} \\ \quad = -h^2 f(x_i,y_j) \\ u_{0j} = g(x_0,y_j), u_{nj} = g(x_n,y_j), \\ u_{i0} = g(x_i,y_0), u_{im} = g(x_i,y_m) \end{cases}$$



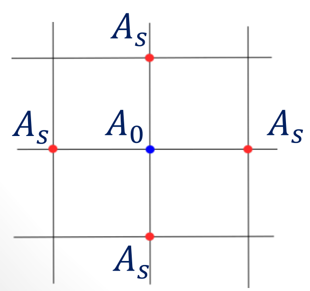
Poisson Results

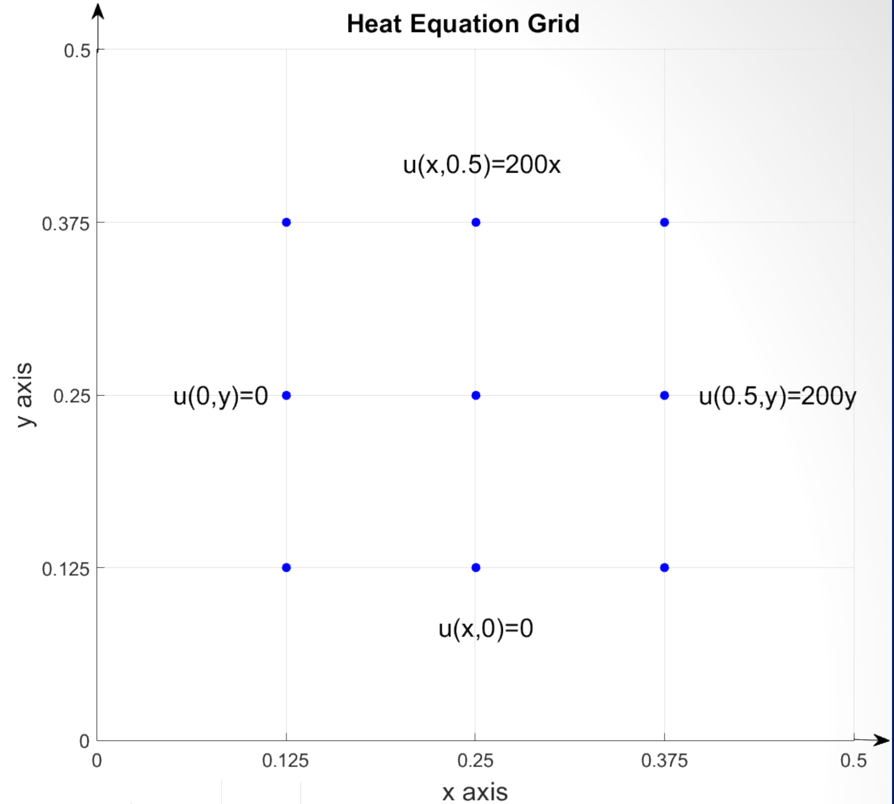
Use of the *five point stencil* with the preceding described *scheme*

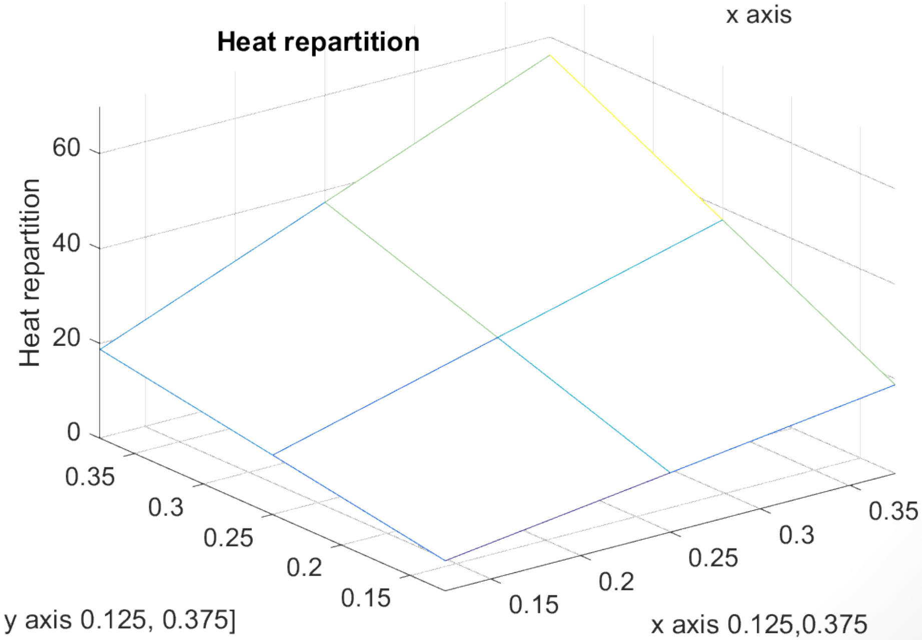
```
L = i + (m-j) * n;

L1:  4 -1  0 -1  0  0  0  0  0      25
L2: -1  4 -1  0 -1  0  0  0  0      50
L3:  0 -1  4  0  0 -1  0  0  0     150
L4: -1  0  0  4 -1  0 -1  0  0       0
L5:  0 -1  0 -1  4 -1  0 -1  0  =    0
L6:  0  0 -1  0 -1  4  0  0 -1      50
L7:  0  0  0 -1  0  0  4 -1  0       0
L8:  0  0  0  0 -1  0 -1  4 -1       0
L9:  0  0  0  0  0 -1  0 -1  4      25
```

$$A\,.\,sol = b \Leftrightarrow sol = A^{-1}b$$

```
sol=
 18.75   37.50   56.25
 12.50   25.00   37.50
  6.25   12.50   18.75
```

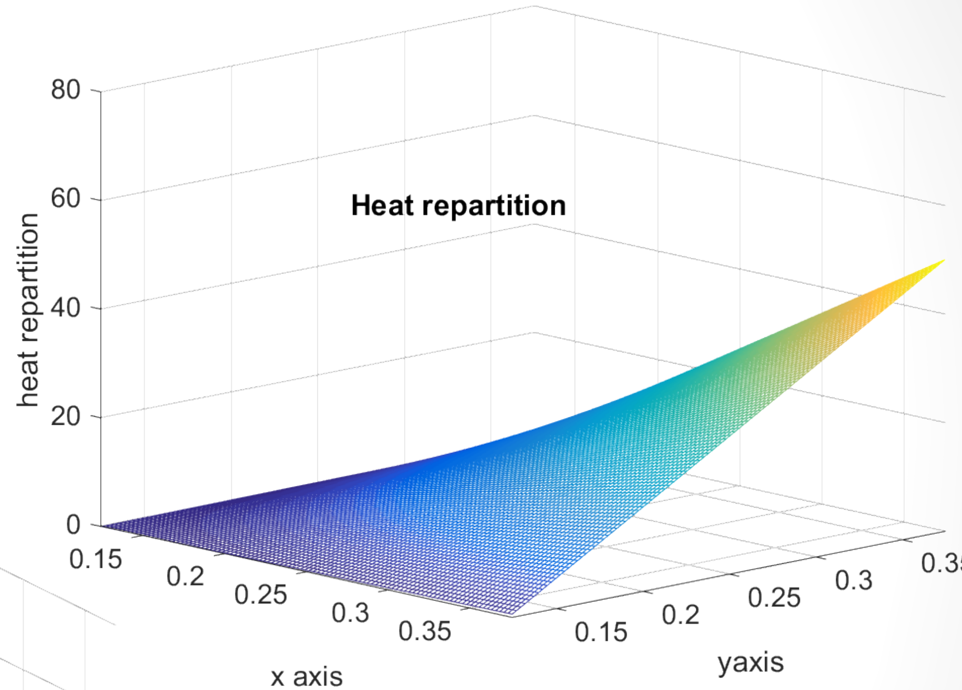


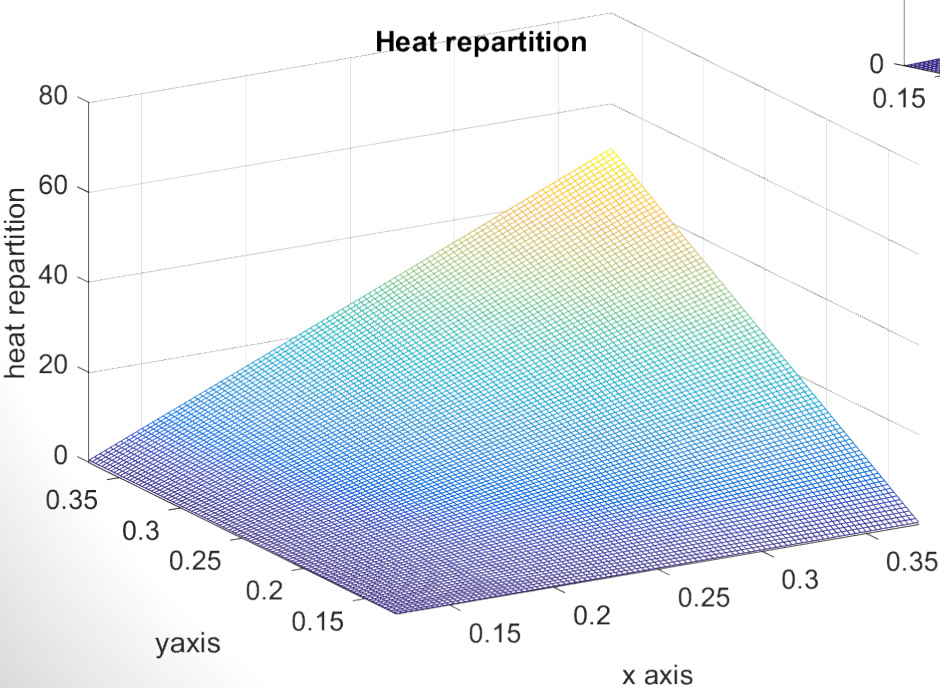




Poisson (suite)

- If $h = 0.01$ the result is more precise…





Exact finite difference scheme for the **Helmholtz** equation

Study of the article:

Wong Yau Shu and Li Guangrui, ***Exact Finite Difference Schemes for Solving Helmholtz Equation at Any Wavenumber***, Institute for Scientific Computing and Information, Ed.: International Journal of Numerical Analysis and Modeling, 2001, vol. 2.

Hegedus G. and Kuczmann M., ***Calculation of the Solution of Two-Dimensional Helmholtz Equation***.: Acta Technica Jaurinensis, 2010, vol. 1.

1D problem

$$\begin{cases} \nabla^2 u(x) + k^2 u(x) = 0, x \epsilon [a, b] \\ \dfrac{\partial u(x)}{\partial x} = iku(x)\ (Sommerfeld) \\ U(a) = \alpha\ (Dirichlet) \end{cases}$$

- Standard scheme (Taylor)

- Central point:

$[2 - (kh)^2]u_i - u_{i+1} - u_{i-1} = 0$

- Sommerfeld (right):

$[2 - (kh)^2 - 2ikh]u_i - 2u_{i-1} = 0$

- Sommerfeld (left):

$[2 - (kh)^2 + 2ikh]u_i - 2u_{i-1} = 0$

- New scheme(Taylor and $\cos(x), \sin(x)$ series)

- $\begin{cases} -k^2 u_i = \frac{1}{h^2}[u_{i+1} - \omega u_i + u_{i-1}] \\ \omega = 2\cos(kh) + (kh)^2 \\ u_{i+1} - 2isin(kh)u_i - u_{i-1} = 0 \end{cases}$

- The scheme is exact i.e. it does not have any truncation error.

Program requirements

- Build a first generic tool to handle the problem according to:
  - The type of scheme
  - Boundary Conditions
    - Sommerfeld Boundary Condition (left or right)
    - Dirichlet Boundary Condition
  - Definition of the range (a, b) for example $(a = 0, b = 1)$
  - h, k values
- The error is the infinite norm:

$$E_\infty = \max_{i=1\ldots n_x} \max_{j=1\ldots n_y} \left| u_{i,j} - \overline{u_{i,j}} \right|$$

- Closed form solution of the 1D problem: $u(x) = e^{ikx}$

Results example of the 1D scheme – compared to literature

- It is possible to reproduce results of the article. $x \in (0,1), u(0) = 1$

Mine

```
h = 0.01
error1 =
    'k'       'SFD'       'NFD'       'SFD'       'NFD'
    [ 10]     [0.0048]    [0.0017]    [0.0040]    [4.5521e-14]
    [ 30]     [0.1106]    [0.0150]    [0.1137]    [1.2969e-14]
    [ 50]     [0.5371]    [0.0431]    [0.5267]    [8.2523e-15]
    [ 70]     [1.3487]    [0.0841]    [1.3859]    [1.5922e-14]
    [100]     [1.9998]    [0.1814]    [1.9998]    [1.0716e-14]
    [150]     [2.4932]    [0.3963]    [2.1038]    [1.4453e-14]
```

TABLE 1. E_∞ for SFD and NFD with h=0.01

From Literature

		SBC		NBC	
kh	k	SFD	NFD	SFD	NFD
0.1	10	0.0048	0.0017	0.0040	4.29e-14
0.3	30	0.1106	0.0149	0.1148	1.26e-14
0.5	50	0.5371	0.0428	0.5274	9.55e-15
0.7	70	1.3487	0.0823	1.3856	1.28e-14
1	100	1.9998	0.1792	2.0216	5.60e-15
1.5	150	2.4932	0.3928	2.0043	5.66e-16

2D problem

$$\begin{cases} \nabla^2 u(x,y) + k^2 u(x,y) = 0, x\epsilon[a,b] \\ \dfrac{\partial u(x,y)}{\partial n} = iku(x,y)\ (Sommerfeld) \\ u(x,y) = \alpha\ (Dirichlet) \end{cases}$$

- Standard scheme (Taylor)
- Central point:

$$4-(kh)^2]u_{i,j} - u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1} = 0$$

- Side point (4 schemes):

$$[4-(kh)^2 - 2ikh]u_{i,j} - 2u_{i-1,j} - u_{i,j+1} - u_{i,j-1} = 0$$

- Corner point (4 schemes):

$$[2-\frac{1}{2}(kh)^2 - i\sqrt{2}kh]u_{i,j} - u_{i-1,j} - u_{i,j-1} = 0$$

- New scheme(Taylor and $\cos(x), \sin(x)$ series)

$$\begin{cases} 4J_0(kh)u_{i,j} - u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1} = 0, \\ i,j \in [1,n] \\ u_{n+1,j} - 2isin(k_1 h)u_{n,j} - u_{n-1,j} = 0, \\ j \in [1,n] \\ u_{j,n+1} - 2isin(k_2 h)u_{n,j} - u_{j,n-1} = 0, \\ j \in [1,n] \end{cases}$$

- The scheme use a Bessel function of the first kind:

$$J_0(kh) = \frac{1}{\pi}\int_0^{\pi} \cos(kh\sin(\theta)\, d\theta$$

- There is one scheme for the central point, 4 scheme for the side and 4 scheme for the corners to integrate Sommerfeld.

2D Program

- Building of a second generic program to handle this problem parameterised by:
  - The type of scheme (new, standard)
  - **The Bessel function and its derivations (integral, exact theta)**
  - Boundary condition (left and/or right) Dirichlet or Sommerfeld
  - the coordinate of the grid (a, b, c, d) for example $(a = 0, b = 1, c = 0, d = 1)$
  - h, k, θ
- The error is the infinite norm:

$$E_\infty = \max_{i=1\ldots n_x} \max_{j=1\ldots n_y} \left| u_{i,j} - \overline{u_{i,j}} \right|$$

- **Closed form solution:**

$$u(x, y) = e^{i(k_1 x + k_2 y)}, \qquad (k_1, k_2) = (k\cos(\theta), k\sin(\theta))$$

2D Results

- $\Omega = [0,1] \times [0,1], Dirichlet\ (South, West), Sommerfeld\ (North, East),$
- Analytical fonction: $u(x) = e^{i(k_1 x + k_2 y)}, (k_1, k_2) = (k cos(\theta), \sin(\theta))$

Mine

```
h: 0.0200
res_tab =
    ''            ''            'E inf'       'E inf'
    'kh'          'k'           'SFD'         'NFD'
    [0.8485]      [42.4264]     [25.0648]     [12.1595]
    [0.7071]      [35.3553]     [14.3569]     [ 7.0017]
    [0.5657]      [28.2843]     [ 7.2401]     [ 3.5638]
    [0.4243]      [21.2132]     [ 3.0220]     [ 1.4996]
    [0.2828]      [14.1421]     [ 0.8713]     [ 0.4450]
    [0.1414]      [ 7.0711]     [ 0.1053]     [ 0.0543]
```

From Literature

TABLE 6. E_∞ and $J_0(kh)$ for $h = 0.02$

		E_∞	
kh	k	SFD	NFD
0.8485	$30\sqrt{2}$	1.70661	3.21431
0.7071	$25\sqrt{2}$	2.60665	0.79162
0.5657	$20\sqrt{2}$	0.71042	0.25167
0.4243	$15\sqrt{2}$	0.20008	0.10524
0.2828	$10\sqrt{2}$	0.13488	0.07627
0.1414	$5\sqrt{2}$	0.04299	0.00349

Higher order finite difference schemes

References:

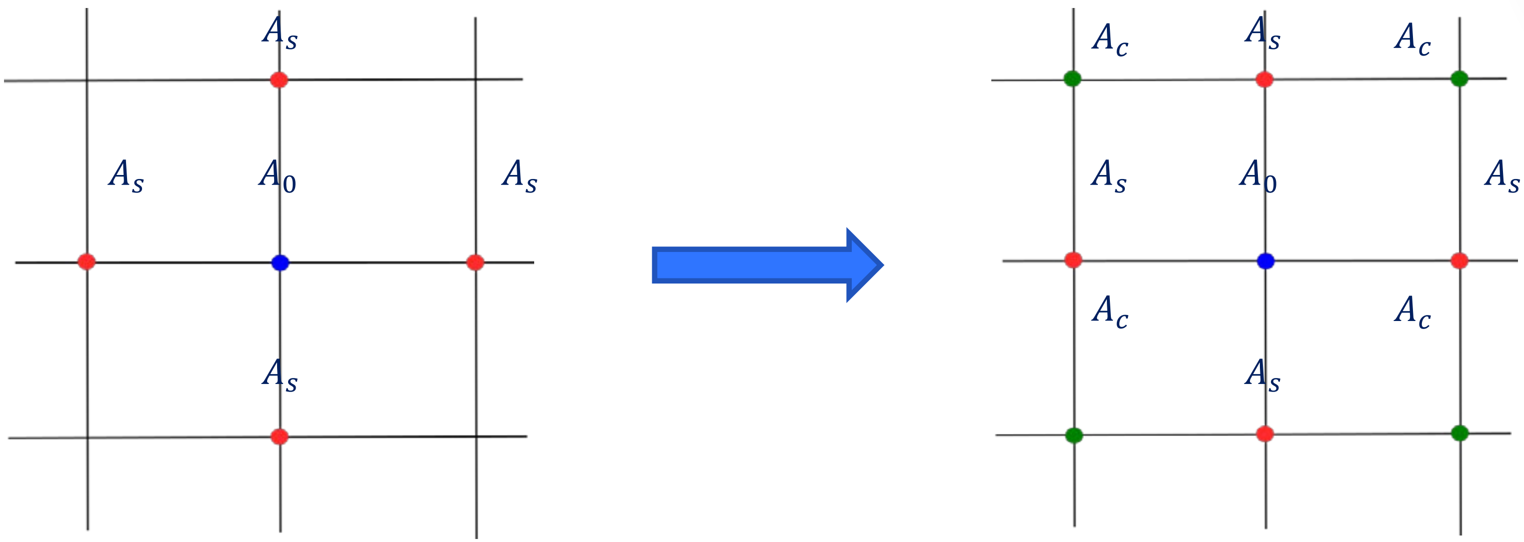
Isaac Harari and Eli Turkel, "**Accurate Finite Difference Methods for Time-Harmonic Wave Propagation**," *Journal of Computational Physics*, vol. 119, 252-270 (1995), 1994.

Yogi Erlangga and Eli Turkel, "**ITERATIVE SCHEMES FOR HIGH ORDER COMPACT DISCRETIZATIONS TO THE EXTERIOR HELMHOLTZ EQUATION**," 2012.

Dan Gordon and Rachel Gordon, "**Parallel solution of high frequency Helmholtz equations using high order finite difference schemes**," *Applied Mathematics and Computation*, vol. 218 (2012) 10737–10754, 2012

Eli Turkel, Dan Gordon, Rachel Gordon, and Semyon Tsynkov, "**Compact 2D and 3D sixth order schemes for the Helmholtz equation with variable wave number**," *Journal of Computational Physics*, vol. 232 (2013) 272–287, 2012

5-points to 9-points stencil



$$A_0\phi_{i,j} + \mathrm{A_s}\sigma_\mathrm{s} + \mathrm{A_c}\sigma_\mathrm{c} = 0$$

$$\sigma_s = \phi_{i,j+1} + \phi_{i+1,j} + \phi_{i,j-1} + \phi_{i-1,j}$$

$$\sigma_c = \phi_{i+1,j+1} + \phi_{i+1,j-1} + \phi_{i-1,j-1} + \phi_{i-1,j+1}$$

Central & Sommerfeld Schemes

Central Scheme

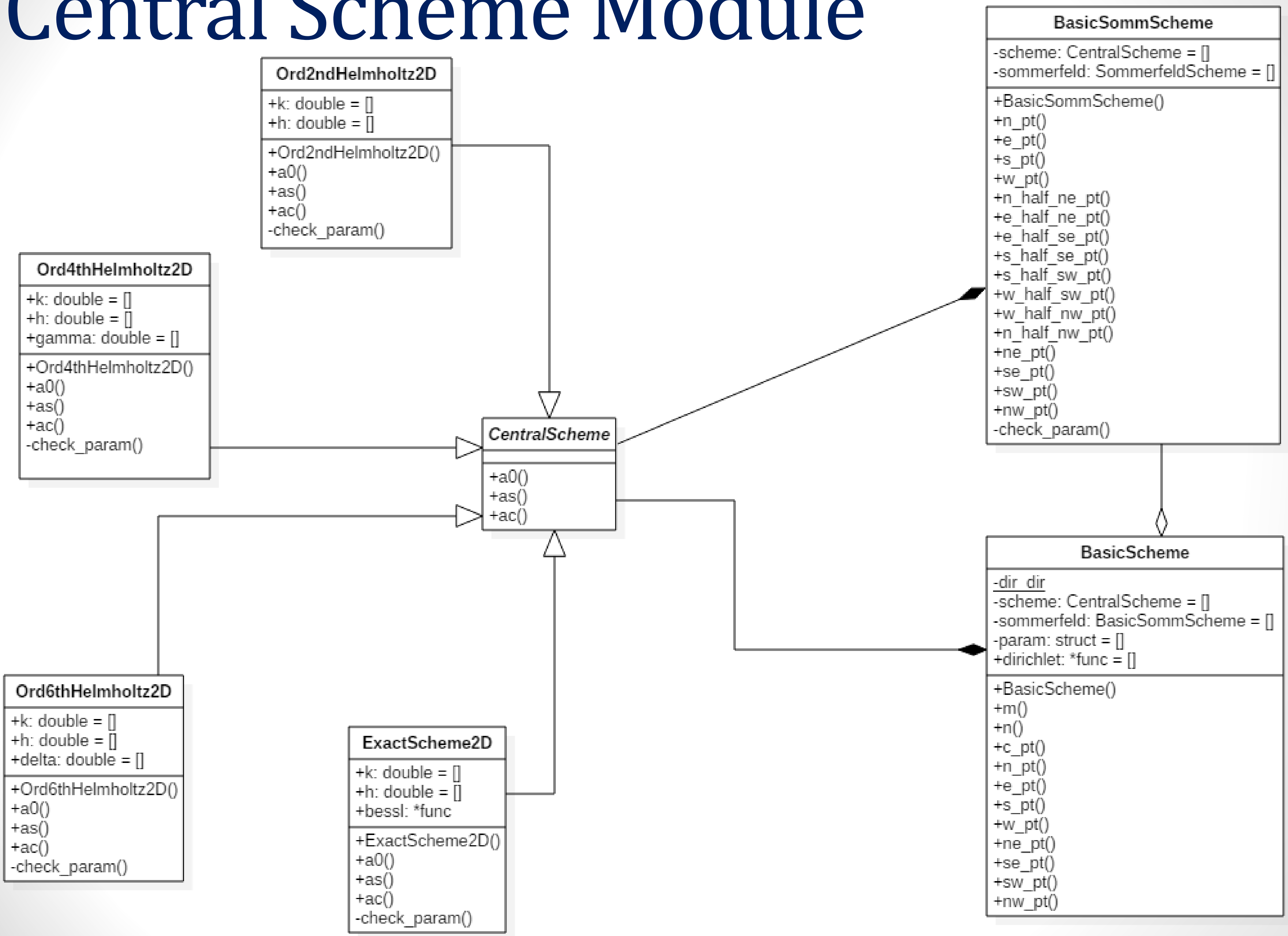
Order	A_0	A_s	A_c
2nd	$-4+(kh)^2$	1	0
4th	$-\frac{10}{3}+(kh)^2(\frac{2}{3}+\frac{\gamma}{36})$	$\frac{2}{3}+(kh)^2(\frac{1}{12}-\frac{\gamma}{72})$	$\frac{1}{6}+(kh)^2\,\gamma/144$
6th	$-\frac{10}{3}+\frac{67}{90}(kh)^2+\frac{\delta-3}{180}(kh)^4$	$\frac{2}{3}+\frac{2}{45}(kh)^2+\frac{3-2\delta}{720}(kh)^4$	$\frac{1}{6}+\frac{7}{360}(kh)^2+\frac{\delta}{720}(kh)^4$
exact	$4J_0(kh)$	-1	0

Order	Sommerfeld scheme
2nd	$u_{n+1}+2ikh\,u_n-u_{n-1}=0$
6th	$u_{n+1}+2i\beta h\left(1-\frac{\beta^2h^2}{6}+\frac{\beta^4h^4}{120}\right)u_n-u_{n-1}=0$
exact	$u_{n+1,j}-2isin(k_1h)u_{n,j}-u_{n-1,j}=0,\ u_{j,n+1}-2isin(k_2h)u_{n,j}-u_{j,n-1}=0$

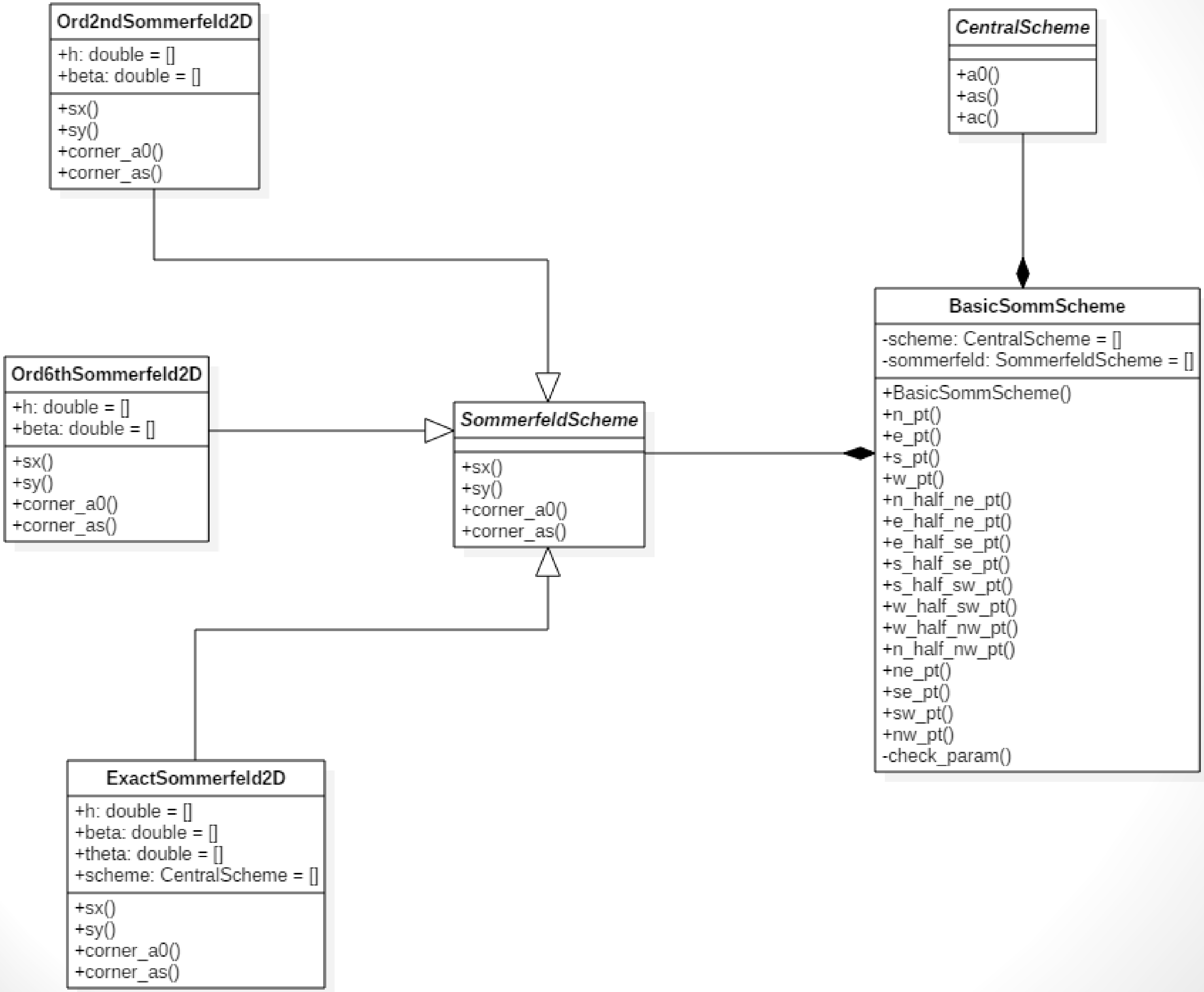
Redesign of the code in OOP

- A strong need for a more flexible type of program that would take into account:
  - Various Central schemes (two types of stencils)
  - Various Sommerfeld schemes
  - A need to easily use a variety of Central Scheme / Boundary Condition combinations
  - A more testable (and tested!) framework
  - Easy to use and to read from a user (and programmer) point of view
  - Clear computation of error
  - Easy and quick graphical representation
  - Export of resulting data

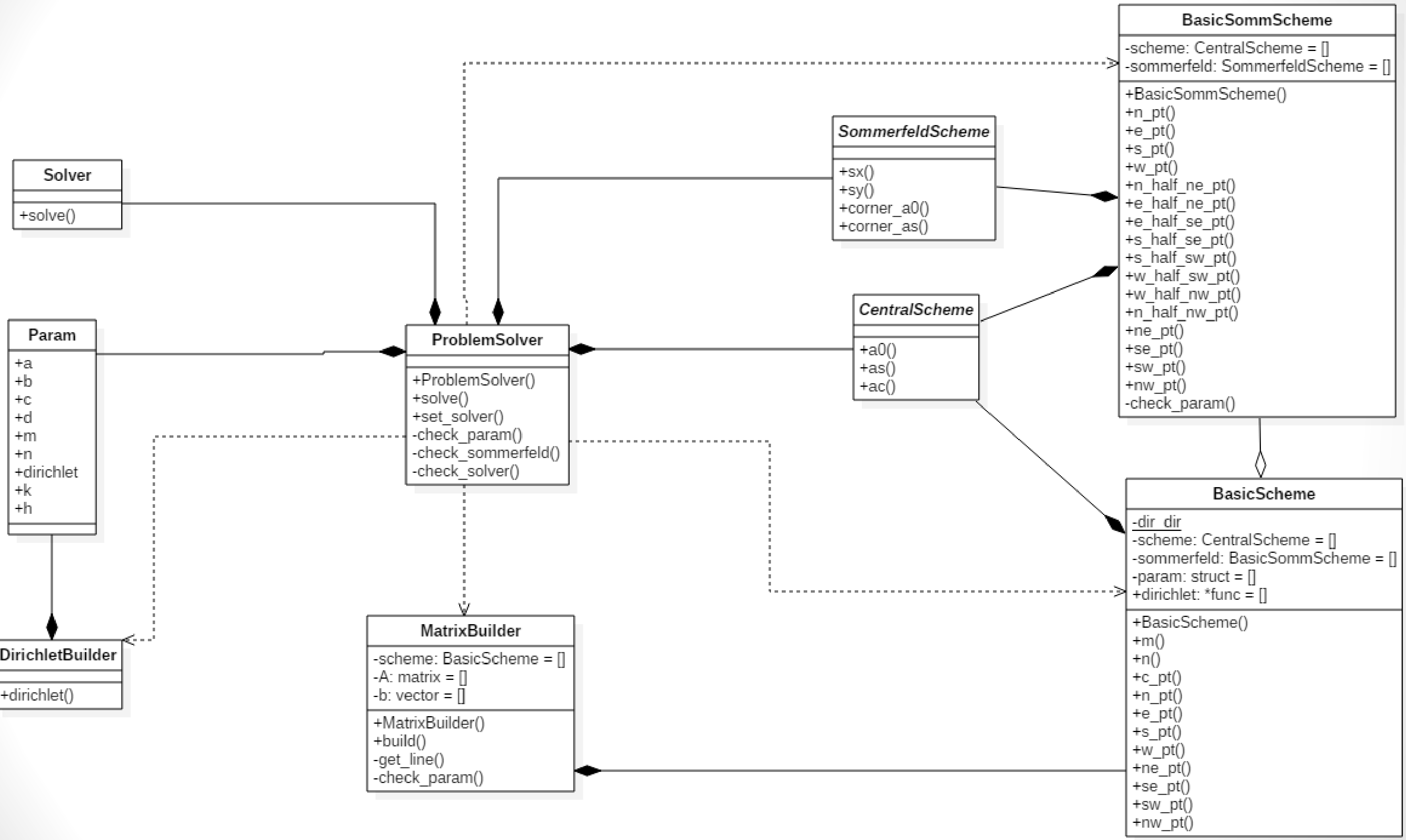
Central Scheme Module



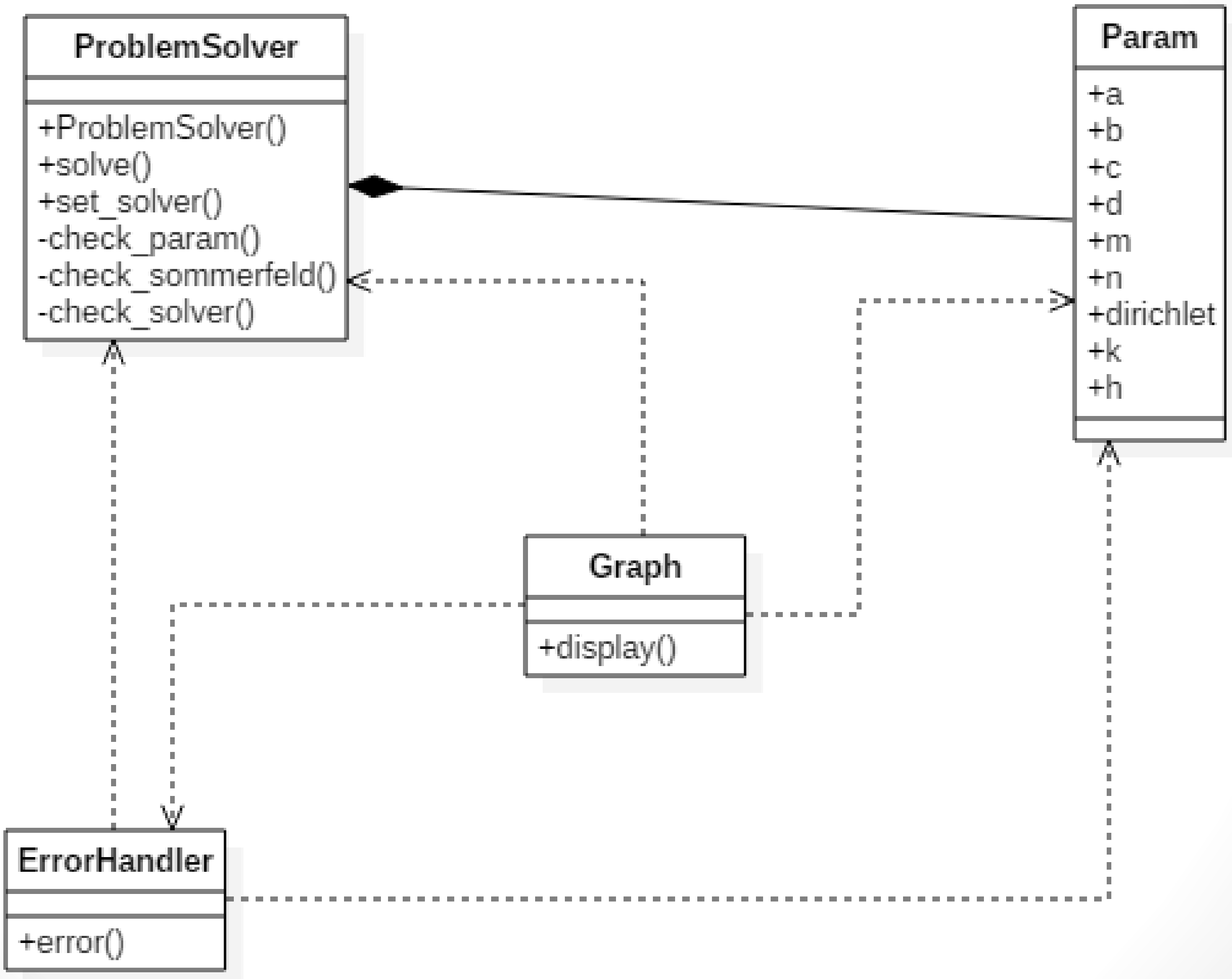
Sommerfeld Scheme Module



Simulation Module

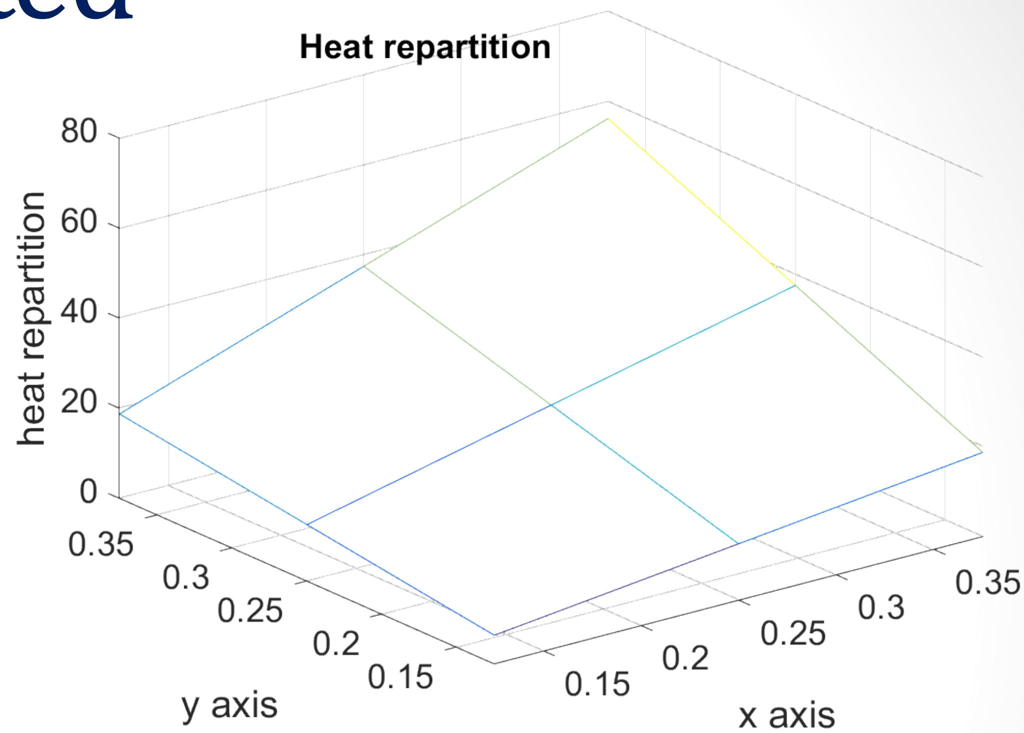


Reporting Module



Poisson revisited

```
% definition of the area we
% simulate in it
param.h = 0.125;%3 pt
param.a = 0.125;
param.b = 0.375;
param.c = 0.125;
param.d = 0.375;
param.m = ...
(param.d - param.c)/param.h + 1;
param.n = ...
(param.b - param.a)/param.h + 1;

% dirichlet function
param.dirichlet = …
@(x,y) poisson_dirichlet( x, y);

scheme = Poisson2D();

% define the solver
solver = @(A, b) A\b;

% solve the problem
ps = …
 ProblemSolver(param, scheme, solver);
[ A, b, sol ] = ps.solve();
```

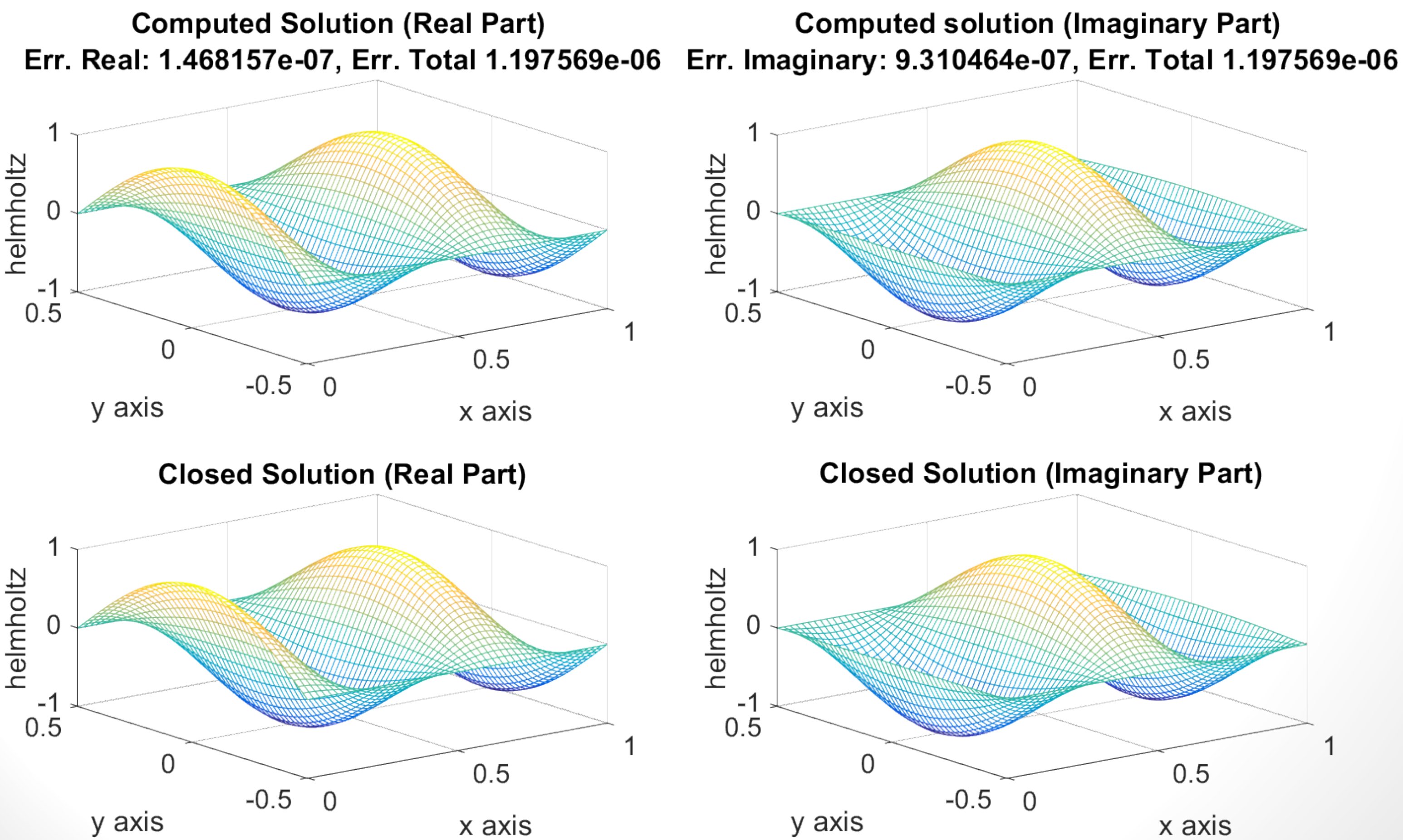



```
>> full(sol)

ans =

   18.7500   37.5000   56.2500
   12.5000   25.0000   37.5000
    6.2500   12.5000   18.7500
```

Example: 6th order scheme

$$\nabla^2 u + k^2 u = 0 \; in \; \Omega = [0,1] \times [-\frac{1}{2}, \frac{1}{2}]$$

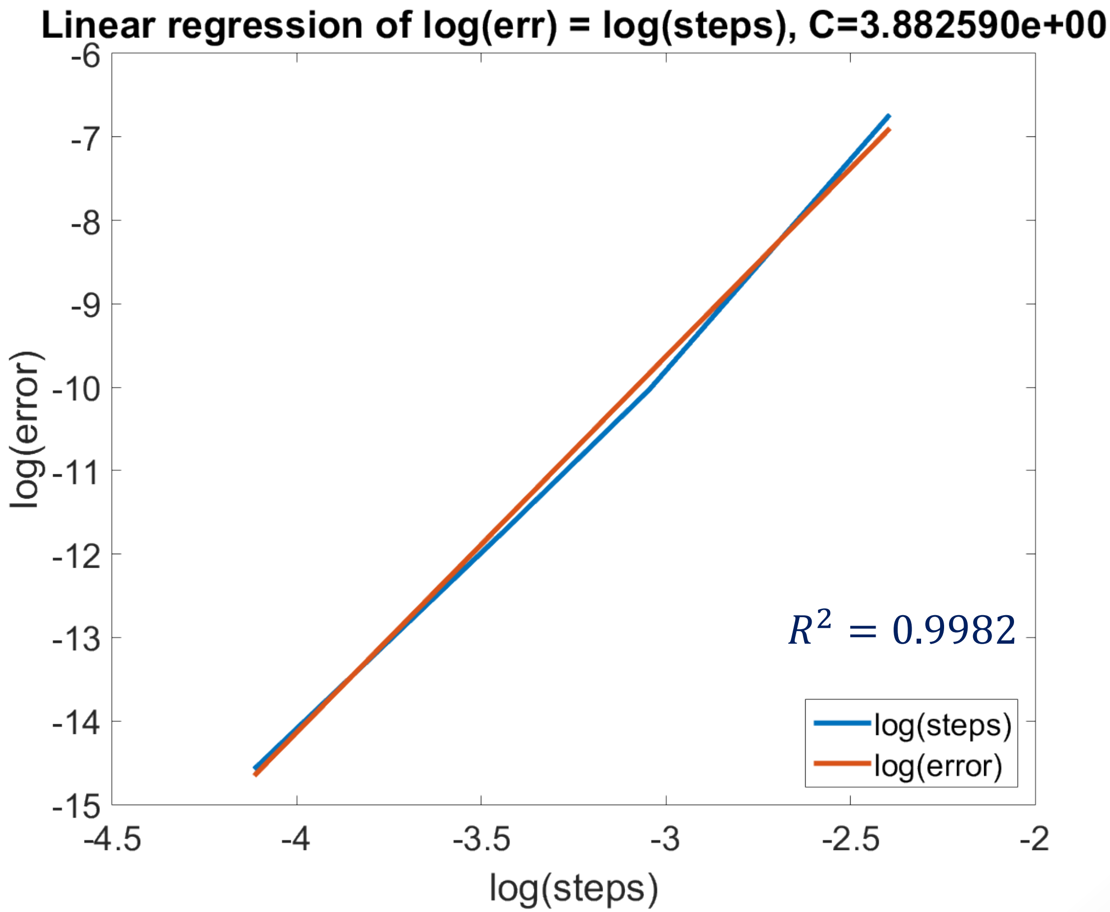
$$u\left(x, -\frac{1}{2}\right) = u\left(x, \frac{1}{2}\right) = 0; u(0,y) = \cos(\pi y)\,; \left.\frac{\partial u}{\partial x} + i\beta\right|_{x=1} = 0$$



Error computation: 6th order

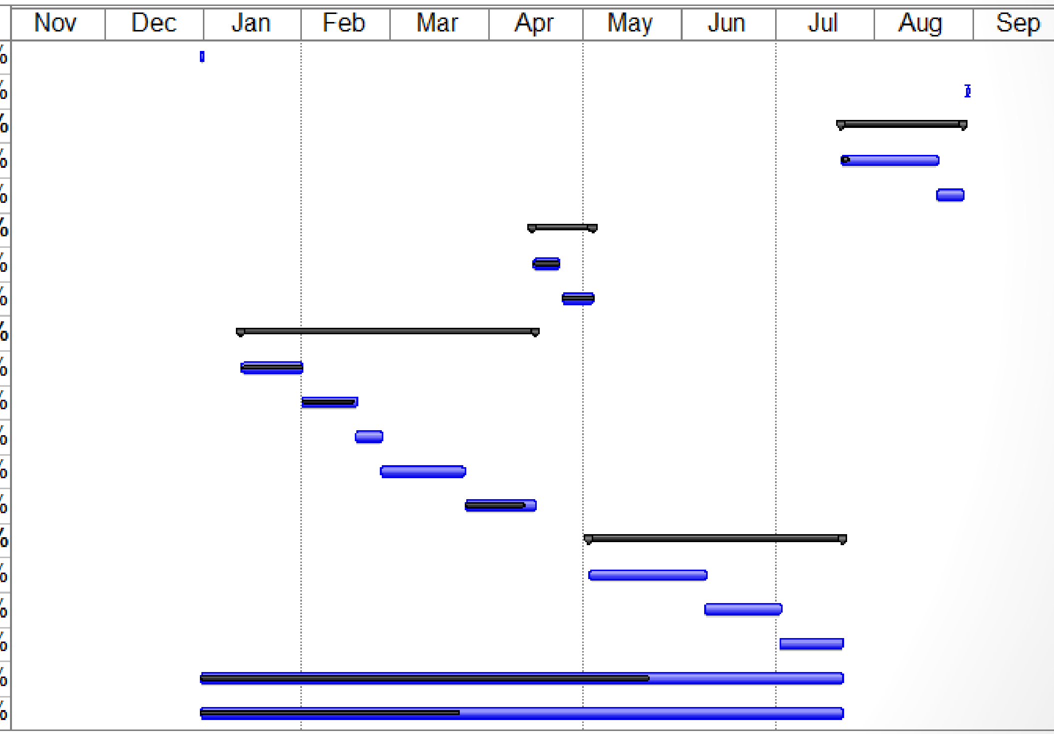
$$Error(h) = Ch^p \Leftrightarrow \log\big(Err(h)\big) = \log(C) + p\log(h)$$

```
'size' [     12         22         32         47         52         57         62]
'h'    [0.0909     0.0476     0.0323     0.0217     0.0196     0.0179     0.0164]
'error'[0.0012 4.4257e-05 8.2439e-06 1.5629e-06 1.0126e-06 6.8324e-07 4.7612e-07]

p = polyfit(log('h'), log('error'), 1); ⇒ [4.5045    3.8826]
```



Progress Report Gantt

ID	Task Name	% Complete
1	Project proposal submission	0%
2	Presentation of the project	0%
3	**Report Wrinting**	**8%**
4	Report Pre-submission	10%
5	Report Correction	0%
6	**Mid-term Presentation**	**100%**
7	Presentation Submission	100%
8	Presentation Correction	100%
9	**Finite Differences**	**56%**
10	Basics of Finite Differences	100%
11	Reproduce Literature Case	90%
12	Fast Marching Method	0%
13	Solve Eikonal/Transport Equation	0%
14	Solve Helmholtz	80%
15	**Finite Element**	**0%**
16	Basic of Finite Element	0%
17	Solve Eikonal/Transport	0%
18	Solve Helmholtz	0%
19	Tool Writing	70%
20	Analysis of Results	40%



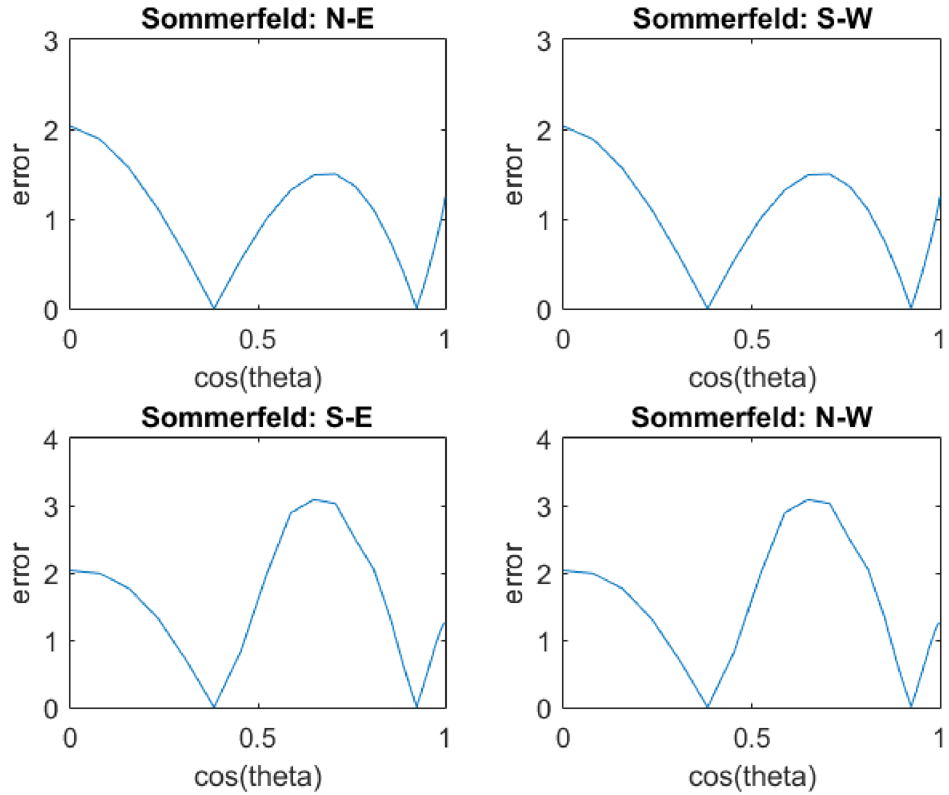
Next Steps

- Correct the actual framework to include correctly mixed and full Sommerfeld corners.
- Analyse errors more deeply and compare them between various combinations of schemes (add more report classes…)
- Compute results with variable k (k may be a function of the position and parametrically depend on the angular frequency).
- Analyse the ability of the scheme to handle position dependent refraction index.
- Solve a problem with a digital anatomy-phantom defined by the position dependent refraction index.
- ***Optional***, if time permits, compare the high-frequency ray approximation to the full solution of the Helmholtz equation.
- ***Wishes:***
  - try to solve a basic Helmholtz equation with Finite Element Method
  - Try to solve 3D case

END

2D Results (cont.)

- $\Omega = [0,1] \times [0,1], Dirichlet\ (South, West),\quad Sommerfeld\ (North, East),$
- Closed form function: $u(x) = e^{i(k_1 x + k_2 y)},\quad (k_1, k_2) = (k\cos(\theta), \mathrm{k}\sin(\theta))$
- Fix $k = \sqrt{2} \times 15$ but $\theta \in \left[0, \frac{\pi}{2}\right], 20\ values$



Search for theta

- Least-square algorithm
  1. Determine the coefficient of the linear system $Ax = b$ by calculating $J_0(kh)\ in\ \left[0, \frac{\pi}{2}\right]$ (is it θ_1 and θ_2 ? N.R).
  2. Solve the system by GMRES (or another solver)
  3. Take partial data from x_{temp} (we take the two lines besides the Dirichlet boundaries in this study) and form the least square function: $f(x,\theta) = \sum_{j=1}^{m}(A(j) - x(j,\theta))^2$ where $A(j)$ are the data from x_{temp} and $x(j,\theta)$ are the exact solution of plane wave $e^{ik(x\cos(\theta)+y\sin(\theta))}$with parameter θ.
  4. Estimate θ using a non-linear least square (algorithm N.R) such as the Levenberg-Marquardt algorithm. Using different approximation in Step 4, we determine θ_1 and θ_2.
  5. Update the coefficient of the system $Ax = b$ by re-computing $J_0(kh)\ in\ [\theta_1, \theta_2]$.
  6. Repeat 2-5 until θ converge.
- The implementation has been done with Dirichlet given the problems in the formulation and the impossibility to find a good explanation for the Sommerfeld boundary.
- It match $\pi/4$ within to iteration (that is in contradiction with the article). Other angles are much more problematic.

Result: Bessel

- Three possible version (from the article):
  - $J_0(kh) = \frac{1}{\pi}\int_0^{\pi} \cos(kh \sin(\theta)\, d\theta$
  - $bessel_{integral(kh)} = \frac{1}{|b-a|}\int_a^{b} \cos(kh \sin(\theta)\, d\theta$
  - $exact_{theta}(kh) = \cos(kh \sin(\theta)$

```
'kh'    'k'      'sum [0,pi]''exact theta''matlab'
[0.8485][42.4264][    0.8279][     0.8253][0.8279]
[0.7071][35.3553][    0.8789][     0.8776][0.8789]
[0.5657][28.2843][    0.9216][     0.9211][0.9216]
[0.4243][21.2132][    0.9555][     0.9553][0.9555]
[0.2828][14.1421][    0.9801][     0.9801][0.9801]
[0.1414][ 7.0711][    0.9950][     0.9950][0.9950]

'J0(kh)'    'J0(kh)'     'J0(kh)'
'sum [0,pi]''exact theta''matlab'
[    3.3118][     3.3013][3.3118]
[    3.5154][     3.5103][3.5154]
[    3.6863][     3.6842][3.6863]
[    3.8220][     3.8213][3.8220]
[    3.9204][     3.9203][3.9204]
[    3.9800][     3.9800][3.9800]
```

$J_0(kh)$	
$[0,\pi]$	Exact θ
3.645368	3.648019
3.752593	3.753879
3.841065	3.841593
3.910337	3.910505
3.960067	3.960099
3.990004	3.990006

2D Results (cont.)

- $\Omega = [0,1] \times [0,1], Dirichlet\ (South, West), Sommerfeld\ (North, East),$
- Analytical function: $u(x) = e^{i(k_1 x + k_2 y)}, (k_1, k_2) = (k cos(\theta), \sin(\theta))$

- Different Bessel functions:

'kh'	'SFD'	'NFD-J0[0,pi]'	'NFD-J0[pi/8,3pi/8]'	'NFD - exact theta'
[42.4264]	[25.0648]	[12.1595]	[4.4379]	[3.4574e-13]
[35.3553]	[14.3569]	[7.0017]	[2.5483]	[4.1767e-13]
[28.2843]	[7.2401]	[3.5638]	[1.2956]	[3.3647e-13]
[21.2132]	[3.0220]	[1.4996]	[0.5450]	[3.8106e-13]
[14.1421]	[0.8713]	[0.4450]	[0.1617]	[1.5676e-13]
[7.0711]	[0.1053]	[0.0543]	[0.0197]	[1.4720e-12]

- Full Dirichlet (all side, same problem)

'kh'	'SFD'	'NFD-J0[0,pi/2]'	'NFD-J0[pi/8,3pi/8]'	'NFD - exact theta'
[42.4264]	[72.8404]	[19.0699]	[9.8070]	[1.0542e-12]
[35.3553]	[22.1503]	[11.3565]	[3.1691]	[3.4689e-13]
[28.2843]	[39.6772]	[9.8812]	[3.1514]	[6.9322e-13]
[21.2132]	[40.1207]	[7.1648]	[3.0067]	[1.5089e-12]
[14.1421]	[1.0660]	[0.5276]	[0.1920]	[1.1014e-13]
[7.0711]	[0.2368]	[0.1198]	[0.0434]	[3.2334e-12]