Polytechnique Montréal
Department of Mechanical Engineering
Course Number : **MEC6215**
Course Name : **Méthodes Numériques en Ingénierie**
Professor : **Sébastien Leclaire**
Homework : **Elliptic Equations (20%)**
Date of Delivery : **November 10th 2021 23:55**

Pedagogical Guidelines

- Submit a concise report of 10 pages maximum on Moodle (excluding first page, table of contents, and references), letter format, no narrow fonts, minimum 12pt, and margin 1.9cm;
- Submit your code that can reproduce your numerical results;
- This work must be done in a team of two people;
- All data (report and code) must be submitted in a single zip file "YOURMATRICULE1-YOURMATRICULE2.zip" (less than 100MB);
- Your report must be perfectly understandable when printed in black and white.
- Prepare a maximum of 6 "PowerPoint" slides (required for the final presentation).

Objective

- Learn the basics of programming solvers adapted to elliptic equations.

1 Two-dimensional Poisson equation and the method of manufactured solutions

Let's consider the two-dimensional elliptic Poisson equation:

$$u_{xx} + u_{yy} = f(x, y) \tag{1}$$

defined on the domain $(x, y) \in [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$. We will use the method of manufactured solutions (Roache [2001]) to generate an analytical solution and assess the code accuracy verification. First, the approach consists in choosing the analytical solution, here $u_a(x, y) = \sin(x) + \cos(y)$ is chosen and it inserted in the target equation (1). This imply that $f(x, y) = -\sin(x) - \cos(y)$. Besides, for any values $x_{\min}$, $x_{\max}$, $y_{\min}$, and $y_{\max}$ of the domain limits, the Dirichlet boundary conditions may be construct from the chosen analytical solution, such that:

$$u(x = x_{\min}, y) = u_a(x = x_{\min}, y) \equiv \sin(x_{\min}) + \cos(y) \tag{2}$$
$$u(x = x_{\max}, y) = u_a(x = x_{\max}, y) \equiv \sin(x_{\max}) + \cos(y) \tag{3}$$
$$u(x, y = y_{\min}) = u_a(x, y = y_{\min}) \equiv \sin(x) + \cos(y_{\min}) \tag{4}$$
$$u(x, y = y_{\max}) = u_a(x, y = y_{\max}) \equiv \sin(x) + \cos(y_{\max}) \tag{5}$$

Second order centered finite differences should be used to discretize the Poisson equation:

$$u_{xx} \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} \tag{6}$$

$$u_{yy} \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} \tag{7}$$

Also, let's construct the discrete version of the domain such that the boundary conditions pass over the computational nodes. If we define n_x and n_y the number of nodes in the x- and y-direction we have:

$$\Delta x = \frac{x_{\max} - x_{\min}}{n_x - 1} \tag{8}$$

$$\Delta y = \frac{y_{\max} - y_{\min}}{n_y - 1} \tag{9}$$

Since the value of $u(x,y)$ on the boundary nodes are known, there is $(n_x - 2)(n_y - 2)$ interior nodes where $u_{i,j}$ needs to be solved for. The errors between the chosen analytical solution u_a and numerical solution u_n should be evaluate with the L2 norm:

$$||e||_{l_2} = ||u_n - u_a||_{l_2} \equiv \left(\int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} ||u_n - u_a||^2 dxdy \right)^{\frac{1}{2}} \tag{10}$$

Exercises Use the MS Visual Studio solution **"MAIN_POISSON2D.sln"** already provided in the homework statement. You may use the information given in Golub and Ortega [1992] as a starting point. However, make sure eventually that your code is general enough such that you may easily change the following parameters n_x, n_y, $x_{\min}$, $x_{\max}$, $y_{\min}$, and $y_{\max}$ independently at the beginning of your program. Solve all the different cases presented in Table (1). **Complete the Excel file "poisson2D.xls"** to obtain the order of accuracy of the scheme. Insert this Excel table into your report document. Explain and add any relevant information that you may have needed to complete this homework. Compare and discuss your results against the analytical solution. Comment and analyze the results you present. Review the pros and cons of the approach and code.

Bonus and challenge You will automatically get 100 % for this assignment if you are able to redo this assignment to solve the Poisson equation, but by using the functionality of the "sparse" array of ArrayFire, while also programming an iterative linear solver. It will be necessary to demonstrate that your program is able to simulate much finer grids than those requested in Table (1). To ensure getting the 100 % score, the "sparse" solver will have to work and also your calculations will have to be correct. In addition, by achieving a final "A" rating for MEC6215, a successful completion of this challenge could lead to a rating increase towards "A*".

Table 1: Parameters for the homework.

Cases	n_x	n_y	$x_{\min}$	$x_{\max}$	$y_{\min}$	$y_{\max}$
1	5	5	0	1	0	1
2	10	10	0	1	0	1
3	20	20	0	1	0	1
4	40	40	0	1	0	1
5	80	80	0	1	0	1
6	7	6	-3	3π	3	4π
7	14	12	-3	3π	3	4π
8	28	24	-3	3π	3	4π
9	56	48	-3	3π	3	4π
10	112	96	-3	3π	3	4π
11	5	9	$-\pi$	2	-5π	3π
12	10	18	$-\pi$	2	-5π	3π
13	20	36	$-\pi$	2	-5π	3π
14	40	72	$-\pi$	2	-5π	3π
15	80	144	$-\pi$	2	-5π	3π

References

Gene H. Golub and James M. Ortega. Chapter 9 - the curse of dimensionality. In Gene H. Golub and James M. Ortega, editors, *Scientific Computing and Differential Equations*, pages 273 – 307. Academic Press, Boston, 1992. ISBN 978-0-08-051669-1. doi: https://doi.org/10.1016/B978-0-08-051669-1.50013-7. URL `http://www.sciencedirect.com/science/article/pii/B9780080516691500137`.

Patrick J. Roache. Code Verification by the Method of Manufactured Solutions . *Journal of Fluids Engineering*, 124(1):4–10, 11 2001. ISSN 0098-2202. doi: 10.1115/1.1436090. URL `https://doi.org/10.1115/1.1436090`.