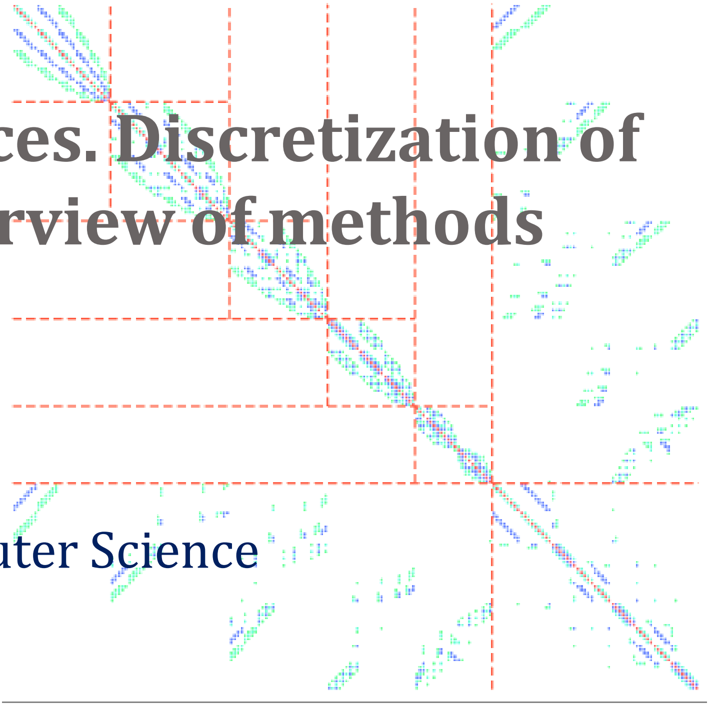
Numerical Methods of Linear Algebra for Sparse Matrices

Lecture 8

Storage schemes for sparse matrices. Discretization of partial differential equations: overview of methods



Anna Nasedkina

Department of Mathematical Modeling

Institute of Mathematics, Mechanics and Computer Science

Southern Federal University

Storage schemes and algorithms of matrix-by-vector multiplication for sparse matrices

Coordinate format

Compressed sparse row format (CRS)

Compressed sparse column format (CRC)

Modified sparse row format (MSR)

Modified sparse column format (MSC)

Diagonal format (DIAG)

Ellpack-Itpack

Coordinate format

$$A = \begin{pmatrix} 1. & 0. & 0. & 2. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 6. & 0. & 7. & 8. & 9. \\ 0. & 0. & 10. & 11. & 0. \\ 0. & 0. & 0. & 0. & 12. \end{pmatrix}$$

$$A \in C^{n \times m}$$

Nz=12
n=5
m=5

AA	12.	9.	7.	5.	1.	2.	11.	3.	6.	4.	8.	10.	Nz
JR	5	3	3	2	1	1	4	2	3	2	3	4	Nz
JC	5	5	3	4	1	4	4	1	1	2	4	3	Nz

Nz – number of nonzero elements, n – number of rows

AA – nonzero entries

JR – row indices

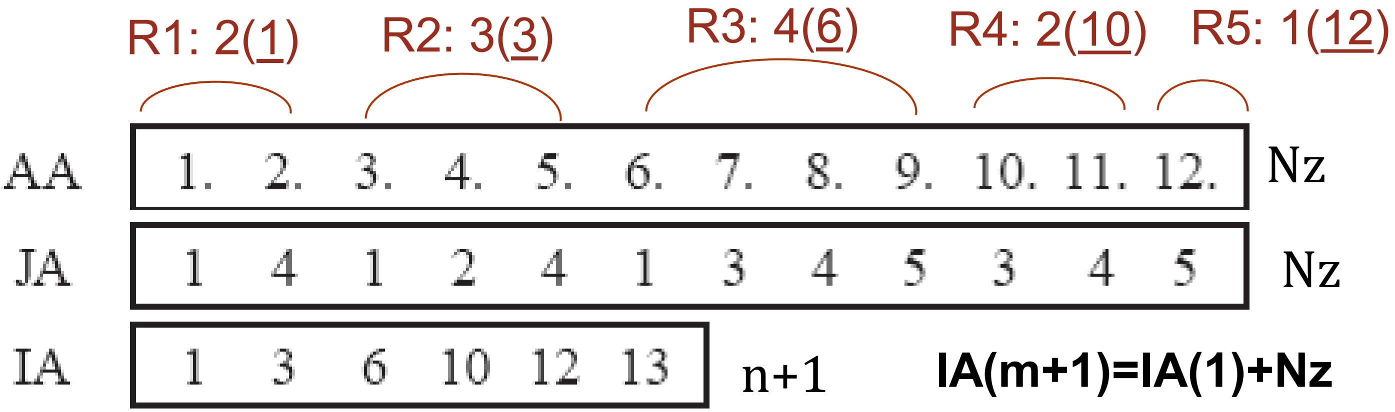
JC – column indices

Compressed Sparse Row (CSR)

$$A = \begin{pmatrix} 1. & 0. & 0. & 2. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 6. & 0. & 7. & 8. & 9. \\ 0. & 0. & 10. & 11. & 0. \\ 0. & 0. & 0. & 0. & 12. \end{pmatrix}$$

$$A \in C^{n \times m}$$

Nz=12
n=5
m=5



AA	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Nz
JA	1	4	1	2	4	1	3	4	5	3	4	5	Nz
IA	1	3	6	10	12	13							n+1

IA(m+1)=IA(1)+Nz

Nz – number of nonzero elements, n – number of rows

AA – nonzero entries by rows,

JA – column indices; **IA** – pointers to the descriptions of rows

Description of i-th row: from IA(i) to IA(i+1)−1

Number of nonzero elements in i-th row: IA(i+1)−IA(i)

Compressed Sparse Column (CSC)

$$A = \begin{pmatrix} 1. & 0. & 0. & 2. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 6. & 0. & 7. & 8. & 9. \\ 0. & 0. & 10. & 11. & 0. \\ 0. & 0. & 0. & 0. & 12. \end{pmatrix} \qquad A \in C^{n \times m}$$

Nz=12
n=5
m=5

$$\text{AA} \quad \boxed{\overbrace{1.\;\;3.\;\;6.}^{\text{C1: }3(\underline{1})}\;\;\overbrace{4.}^{\text{C2: }1(\underline{4})}\;\;\overbrace{7.\;\;10.}^{\text{C3: }2(\underline{5})}\;\;\overbrace{2.\;\;5.\;\;8.\;\;11.}^{\text{C4: }4(\underline{7})}\;\;\overbrace{9.\;\;12.}^{\text{C5: }2(\underline{11})}} \quad \text{Nz}$$

$$\text{JA} \quad \boxed{1\;\;2\;\;3\;\;2\;\;3\;\;4\;\;1\;\;2\;\;3\;\;4\;\;3\;\;5} \quad \text{Nz}$$

$$\text{IA} \quad \boxed{1\;\;4\;\;5\;\;7\;\;11\;\;13} \quad m+1$$

IA(n+1)=IA(1)+Nz

Nz – number of nonzero elements, m – number of columns

AA – nonzero entries by columns,

JA – row indices; **IA** – pointers to the descriptions of columns

Description of i-th column: from IA(i) to IA(i+1)−1

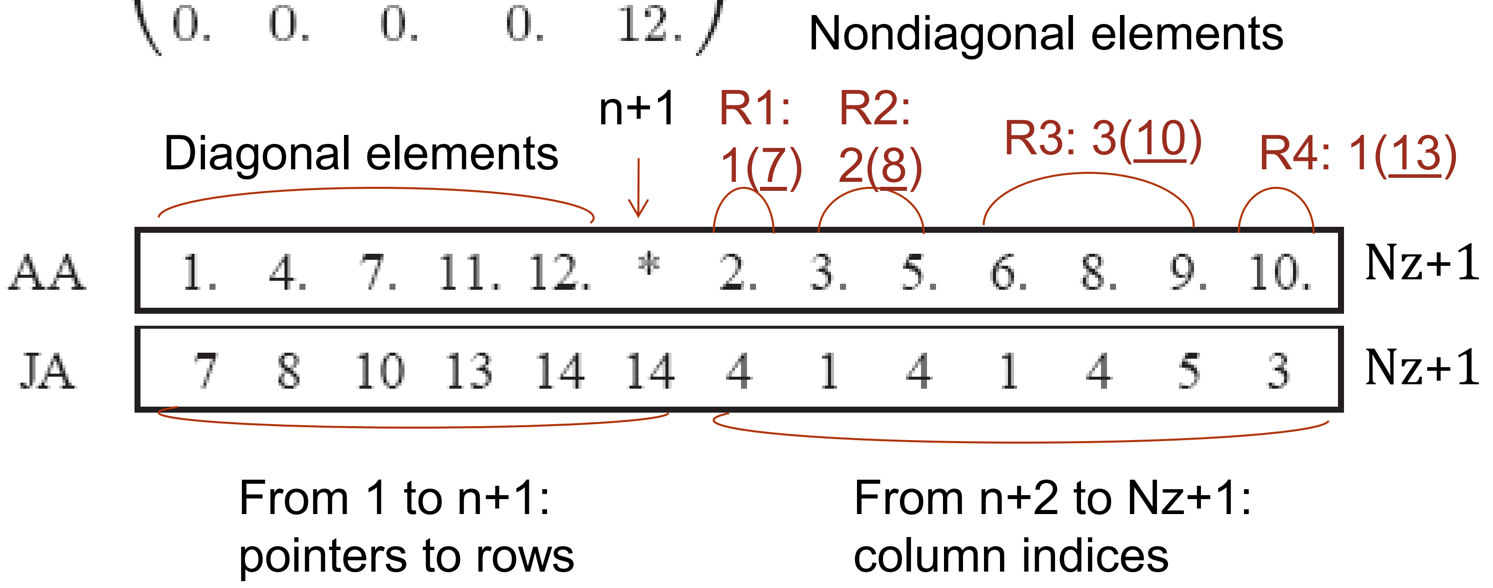
Number of nonzero elements in i-th column: IA(i+1)−IA(i)

Modified Sparse Row (MSR)

$$A = \begin{pmatrix} 1. & 0. & 0. & 2. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 6. & 0. & 7. & 8. & 9. \\ 0. & 0. & 10. & 11. & 0. \\ 0. & 0. & 0. & 0. & 12. \end{pmatrix}$$

$$A \in C^{n\times n}$$

For square matrix
Nz=12
n=5



Nondiagonal elements

Diagonal elements

n+1

R1: 1(7)

R2: 2(8)

R3: 3(10)

R4: 1(13)

AA	1.	4.	7.	11.	12.	*	2.	3.	5.	6.	8.	9.	10.	Nz+1
JA	7	8	10	13	14	14	4	1	4	1	4	5	3	Nz+1

From 1 to n+1:
pointers to rows

From n+2 to Nz+1:
column indices

Nz – number of nonzero elements, n – size of matrix

AA – nonzero entries: main diagonal and nondiagonal elements by rows

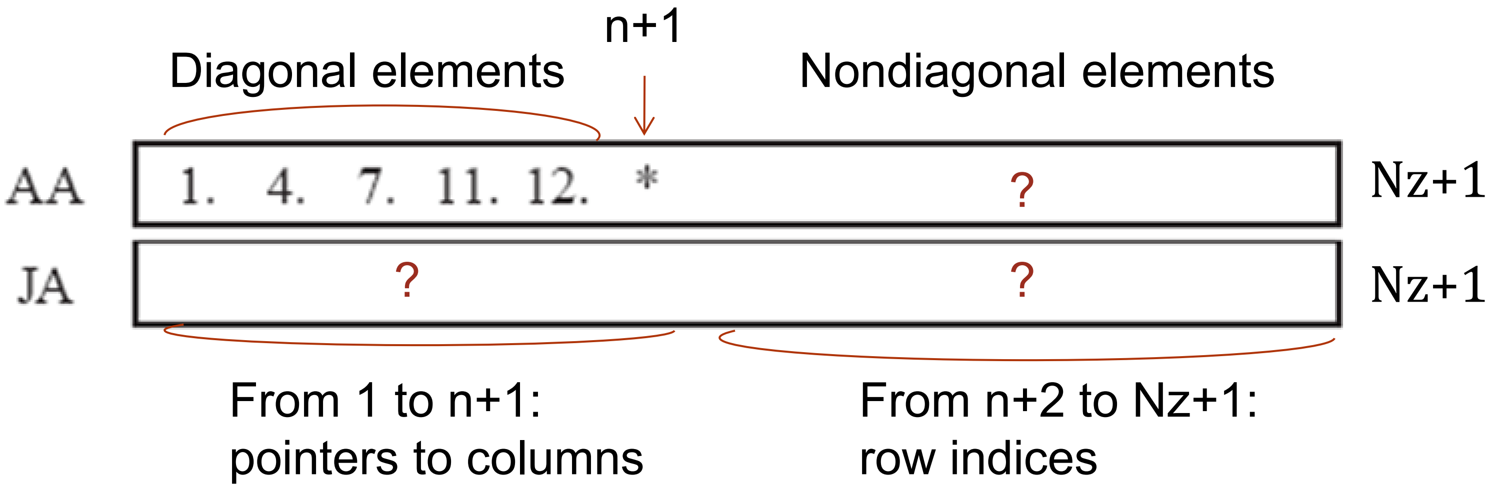
JA – pointers to rows and column indices

Modified Sparse Column (MSC)

$$A = \begin{pmatrix} 1. & 0. & 0. & 2. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 6. & 0. & 7. & 8. & 9. \\ 0. & 0. & 10. & 11. & 0. \\ 0. & 0. & 0. & 0. & 12. \end{pmatrix}$$

$$A \in C^{n \times n}$$

For square matrix
Nz=12
n=5



n+1

Diagonal elements

Nondiagonal elements

AA 1. 4. 7. 11. 12. * ? Nz+1

JA ? ? Nz+1

From 1 to n+1:
pointers to columns

From n+2 to Nz+1:
row indices

Nz – number of nonzero elements, n – size of matrix

AA – nonzero entries: main diagonal and nondiagonal elements by columns

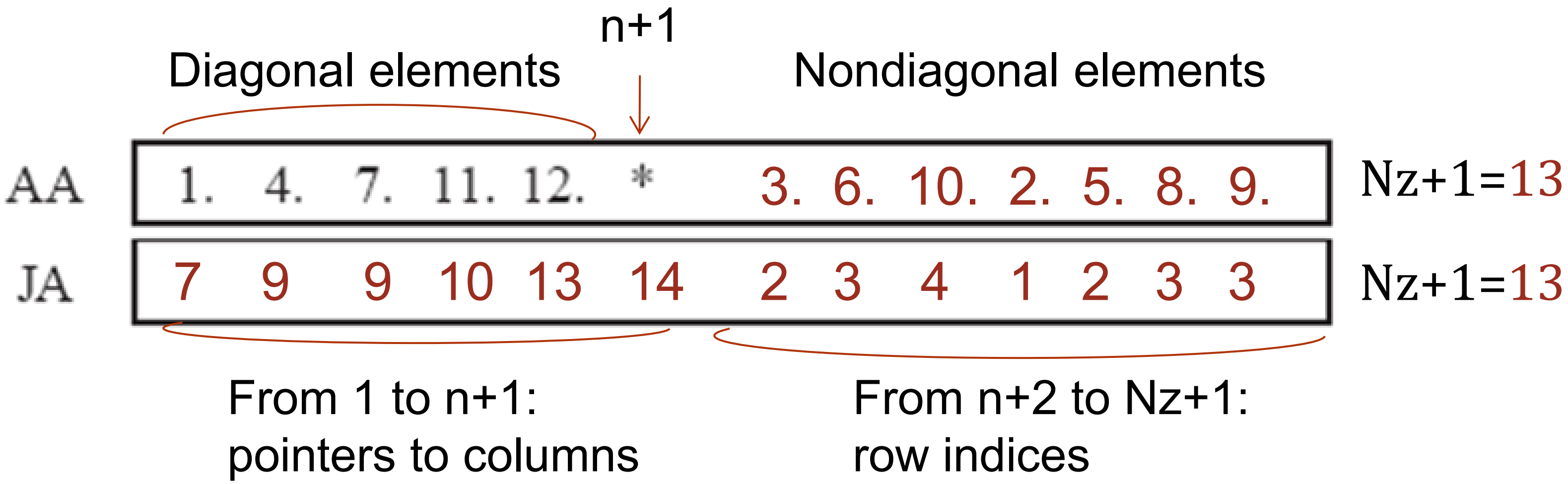
JA – pointers to columns and row indices

Modified Sparse Column (MSC): answer

$$A = \begin{pmatrix} 1. & 0. & 0. & 2. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 6. & 0. & 7. & 8. & 9. \\ 0. & 0. & 10. & 11. & 0. \\ 0. & 0. & 0. & 0. & 12. \end{pmatrix}$$

$$A \in C^{n\times n}$$

For square matrix
Nz=12
n=5



n+1

Diagonal elements

Nondiagonal elements

AA	1.	4.	7.	11.	12.	*	3.	6.	10.	2.	5.	8.	9.	Nz+1=13
JA	7	9	9	10	13	14	2	3	4	1	2	3	3	Nz+1=13

From 1 to n+1:
pointers to columns

From n+2 to Nz+1:
row indices

Nz – number of nonzero elements, n – size of matrix
AA – nonzero entries: main diagonal and nondiagonal elements by columns
JA – pointers to columns and row indices

Diagonal format (DIAG)

$$A = \begin{pmatrix} 1. & 0. & 2. & 0. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 0. & 6. & 7. & 0. & 8. \\ 0. & 0. & 9. & 10. & 0. \\ 0. & 0. & 0. & 11. & 12. \end{pmatrix}$$

$$A \in C^{n \times n}$$

For square matrix
Nd=3
Nz=12
n=5

$$\mathrm{DIAG} = \boxed{\begin{array}{ccc} * & 1. & 2. \\ 3. & 4. & 5. \\ 6. & 7. & 8. \\ 9. & 10. & * \\ 11 & 12. & * \end{array}}$$

$$\mathrm{IOFF} = \boxed{-1 \quad \overset{\text{Main diagonal}}{\overset{\downarrow}{0}} \quad 2}$$

DIAG(i,j)←a(i,i+IOFF(j))

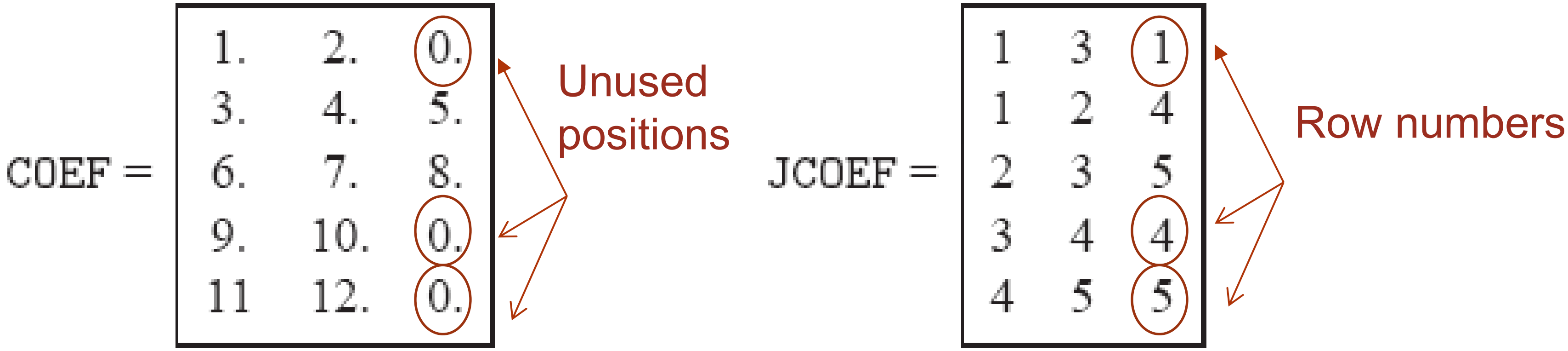
Nd – number of diagonals, n – size of matrix
DIAG – 2D array [1..n,1..Nd], its columns contain diagonals of the matrix
IOFF – array [1..Nd], contains offsets of diagonals with respect to the main diagonal

Ellpack-Itpack format

$$A = \begin{pmatrix} 1. & 0. & 2. & 0. & 0. \\ 3. & 4. & 0. & 5. & 0. \\ 0. & 6. & 7. & 0. & 8. \\ 0. & 0. & 9. & 10. & 0. \\ 0. & 0. & 0. & 11. & 12. \end{pmatrix}$$

$$A \in C^{n \times m}$$

Nmax=3
n=5
m=5



$$\mathrm{COEF} = \begin{bmatrix} 1. & 2. & \bigcirc\!\!\!\!\!\!\;0. \\ 3. & 4. & 5. \\ 6. & 7. & 8. \\ 9. & 10. & \bigcirc\!\!\!\!\!\!\;0. \\ 11 & 12. & \bigcirc\!\!\!\!\!\!\;0. \end{bmatrix}$$

$$\mathrm{JCOEF} = \begin{bmatrix} 1 & 3 & \bigcirc\!\!\!\!\!\!\;1 \\ 1 & 2 & 4 \\ 2 & 3 & 5 \\ 3 & 4 & \bigcirc\!\!\!\!\!\!\;4 \\ 4 & 5 & \bigcirc\!\!\!\!\!\!\;5 \end{bmatrix}$$

Nmax – maximal number of nozero elements per row, n – number of rows

COEFF – 2D array [1..n,1..Nmax], its rows contain nonzero entries by rows

JCOEFF – 2D array [1..n,1..Nmax], its rows contain column positions of nonzero entries

Algorithms of matrix-by-vector multiplication for CSR

- **CSR format**

  N is the number of rows, Ax=z

  IA are the pointers of rows, JA are the column indices

```
for i=1:N
z(i)=0
for j=IA(i):IA(i+1)-1
z(i)=z(i)+x(JA(j))*AA(j)
end
end
```

Algorithms of matrix-by-vector multiplication for CSC

- **CSC format**

  N is the number of rows, M is the number of columns, Ax=z

  IA are the pointers of columns, JA are the row indices

```
for i=1:N
z(i)=0
end
for j=1:M
for i=IA(j):IA(j+1)-1
z(JA(i))=z(JA(i))+x(j)*AA(i)
end
end
```

Discretization of PDEs

Discretization methods

Discretization of Partial Differential Equations

- Partial differential equations (PDEs) are the main source of linear systems with sparse matrices of large and extra-large size

- Discretization of PDE is the approximation of a boundary-value problem by a linear system of equations with *finite* number of unknowns

Discretization methods

- Finite difference method (FDM): the derivatives are approximated using the finite difference formulas based on low order Taylor series expansions
- Finite element method (FEM): unknown functions are approximated by piecewise-polynomial functions that are continuous on small elements of simple shape
- Finite volume method (FVM): volume integrals that contain a divergence term are converted to surface integrals, using the divergence theorem
- other methods

Poisson’s equation

- Poisson’s equation is a classic case to study discretization methods

Consider 2D Poisson's equation

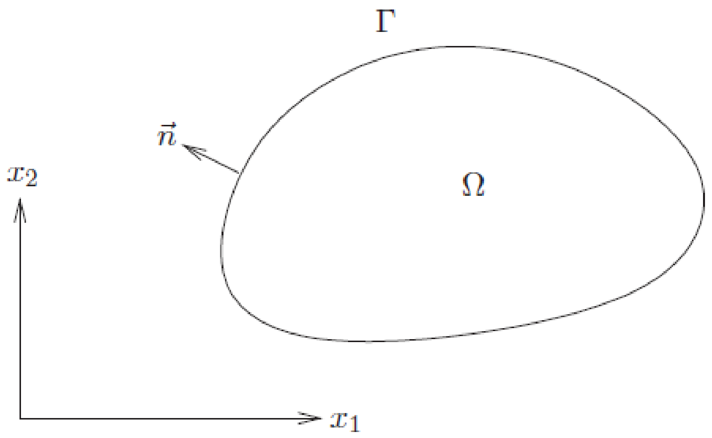
$-\Delta u(\underline{x}) = f(\underline{x})$ for $\underline{x} = (x_1, x_2)$ in a bounded open domain $\Omega \subset \mathbb{R}^2$,

$\Gamma = \partial\Omega$ is the boundary of Ω, $\overline{\Omega} = \Omega \cup \partial\Omega$ is the closed domain

$u(\underline{x}) = u(x_1, x_2)$ is a scalar **unknown function**, $f(\underline{x})$ is a known function

$\Delta = \dfrac{\partial^2}{\partial x_1^2} + \dfrac{\partial^2}{\partial x_2^2}$ is Laplacian

$\underline{n}$ is external unit vector normal to Γ (directed outwards)



Types of boundary conditions

- There are three common types of boundary conditions (BC) for PDEs
- Equation together with boundary conditions form a boundary-value problem

- Dirichlet (essential, BC of the 1st kind)

$$u(\underline{x}) = \varphi(\underline{x}) \text{ on } \Gamma_1$$

- Neumann (natural BC of the 2nd kind)

$$\frac{\partial u(\underline{x})}{\partial \underline{n}} = 0 \text{ on } \Gamma_2$$

- Robin (BC of the 3rd kind)

$$\alpha(\underline{x})u(\underline{x}) + \beta(\underline{x})\frac{\partial u(\underline{x})}{\partial n} = \gamma(\underline{x}) \quad \text{on } \Gamma_3$$

Poisson’s equation in 1D and 2D

1D Poisson's equation

$$-\frac{d^2u(x)}{dx^2} = f(x),\ x \in \Omega \subset \mathbb{R}$$

Consider domain as a segment on $\mathbb{R}$:

$$\Omega = (0,l),\ \overline{\Omega} = [0,l],\ \Gamma = \partial\Omega$$

Consider homogeneous boundary conditions: $u(x)\big|_{\Gamma} = 0$ or $u(0) = u(l) = 0$

2D Poisson's equation

$-\Delta u(x_1, x_2) = f(x_1, x_2)$ or

$$-\frac{\partial^2 u(x_1,x_2)}{\partial x_1^2} - \frac{\partial^2 u(x_1,x_2)}{\partial x_2^2} = f(x_1,x_2),\ \underline{x} = (x_1,x_2) \in \Omega \subset \mathbb{R}^2$$

Consider rectangular domain on $\mathbb{R}^2$:

$$\Omega = (0,l_1)\times(0,l_2),\ \overline{\Omega} = [0,l_1]\times[0,l_2],\ \Gamma = \partial\Omega$$

Consider homogeneous boundary conditions: $u(x_1,x_2)\big|_{\Gamma} = 0$