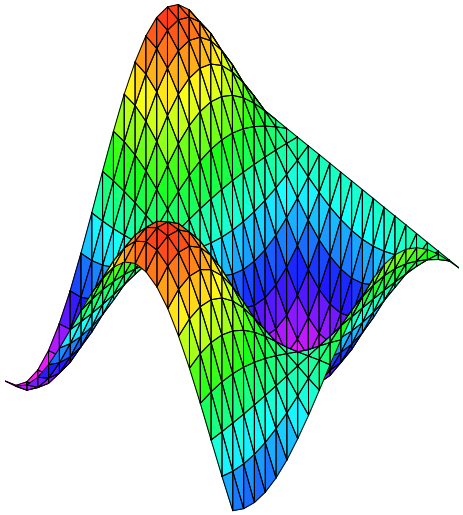
Lecture Notes on Numerical Analysis

Manuel Julio García
Department of Mechanical Engineering
EAFIT University Medellin, Colombia



August 4, 2006

Contents

Preface

The aim of *Engineering Analysis* is to model the physical behaviour of structures under the interaction of external loads such as forces and temperatures, in order to verify that they comply with design specifications.

The course Advanced Methods in Numerical Analysis is a postgraduate course of the Mechanical Engineering Department at EAFIT University in Medellin, Colombia. The aim of the course is to study the numerical (computational) solutions to mathematically formulated physical problems. The course covers an introduction to state-of-the-art linear algebra matrix methods as well as an introduction to the numerical formulation of continuum mechanics problems such as: heat transfer, potential flow, diffusion, and others that can be modelled by a second-order differential equation. The method of solution studied during this part is the Finite Element Method.

These notes grew from my undergrad notes while attending professor Jose Rafael Toro's course, Numerical Analysis, at Los Andes University. From there they evolved after several years of teaching the field while at los Andes University and then at EAFIT University. They do not pretend to be a mathematical theory of numerical methods or Finite Elements. Instead, they intend to introduce the students to powerful mathematical and numerical techniques and to motivate them to further study in the topic. Numerical modelling can be summarised as follows: Start by establishing the nature of the problem and the mathematical equations that model its physics, transform the mathematical formulation into a suitable form to be solve by a computer and finally implement the solution into a computer code. That covers the whole cycle. The course is not all inclusive in topics but tries to cover and understand all the single steps involved in the process of solving a basic continuum problem by computer. There is no preference for computer languages but Matlab, Maple, and C++ are used extensively. However, the algorithms presented in the text are generic and

do not compromise with any of these languages.

Care must be taken. These notes are not in the final version and may have some typos and missing parts. I apologise for that. They are intended as a guide during the course to help the students in the understanding of the concepts and algorithms given during the course. New examples will be added continuously. I hope you find this work to be of help.

A preliminary course in calculus of several variables and a computer programing course are required for success in the course.

Manuel García
Medellín, September 2002

$\binom{2}{3}$ $\quad$ $4/5$

Notation

A	matrix
a_{ij}	component in row i column j of the A matrix
a_j	*jth* column of A
a_i^*	*ith* row of A
$\mathbf{x}$	a vector of n elements
$\mathbf{x}^T$	the traspouse of x, row vector
$\alpha,\ \beta$	scalar quantities
E_i	the *ith* equation of a system of equations
(E1)	a property of a vector space
$N(\mathbf{x})$	a norm of $\mathbf{x}$
W_i	the *ith* base function
Ω	a domain
Γ	the boundary of a domain Ω
∂_k	$\frac{\partial}{\partial x_k}$
$a(u, v)$	a bilinear operator
$\ell(v)$	a linear operator
$\langle , \rangle$	inner product
A^D	Dirichlet stiffness matrix. Does not contain rows or columns in corrresponding to the dirichlet boundary.
∇v	gradient of v
$\nabla^2 v$	Laplacian of v

Chapter 1

Linear Systems

1.1 Linear Systems of Equations

This section deals with finding the value of the variables $x_1, x_2, x_3 \ldots, x_n$ that simultaneously satisfy a set of linear equations which are present in the solution of many physical problems. In general, a system of n linear equations can be represented as

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n &= b_n \end{aligned}$$

where 'a' represents real coefficients and 'b' independent real constants. A system of equations have a geometrical interpretation that helps to to understand the meaning of a system of equations, also the mechanics of the the method of solution, and the meaning of singular systems. Next section, we introduce these concepts starting with a system of two equations and later we extend the concept to the general case.

1.1.1 Geometric interpretation

Let's have the following system of two equations:

$$2x_1 + 4x_2 = 2 \tag{1.1}$$

$$4x_1 + 11x_2 = 1. \tag{1.2}$$

This system can be written in matrix form as:

$$\begin{bmatrix} 2 & 4 \\ 4 & 11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

or in a compact notation:

$$A\,\mathbf{x} = \mathbf{b}$$

or

$$[A]\,\mathbf{x} = \mathbf{b}$$

with

$$A = \begin{bmatrix} 2 & 4 \\ 4 & 11 \end{bmatrix}, \qquad \text{and} \qquad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}.$$

We refer to x_i as the ith component of vector $\mathbf{x}$. Also in the literature it is common to refer to the ith component of vector $\mathbf{x}$ as $[x]_i$ and $x(i)$. In the same way a_{ij} represent the ith row, jth column component of the matrix A. Unless it is explicitly stated, a_{ij} and x_i are real numbers. That is x_i and $a_{ij} \in \mathbb{R}$. Rows and columns are denoted using Matlab colon notation: thus $A[i,:]$ represents the ith row and $A[:,i]$ represents the ith column.

A system of equations have a geometric interpretation that differs if we look at the rows or the columns of the system.

Row representation – (System of two equations)

Equations 1.1 and 1.2 are linear equations that can be plotted as lines in a two dimensional plane. The simultaneous solution represents the intersection of the two lines in the plane, see figure 1.1.

Column representation – (System of two equations)

Notice that equations 1.1 and 1.2 can also be written also in the following way:

$$x_1 \begin{pmatrix} 2 \\ 4 \end{pmatrix} + x_2 \begin{pmatrix} 4 \\ 11 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

That is the sum of two vectors which are multiplied by scalars x. Remember that when a number is multiplied by a scalar quantity, its magnitude is changed by this factor. Figure 1.2 shows the vectors represented by the

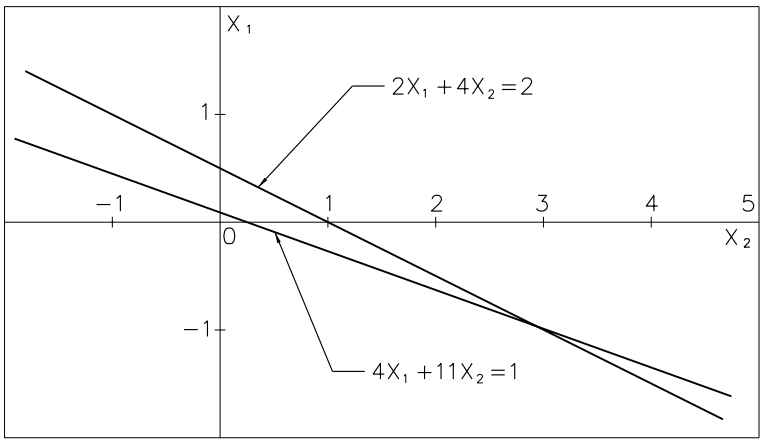


Figure 1.1: Geometric representation to the solution of a system of equations. Row representation.

columns of the system. If we multiply the first vector $(2, 4)$ by 3 and the second $(4, 11)$ by -1 and then add them, the result is vector $(2, 1)$ which is the solution to the system. This result is not a coincidence at all. As the columns are linearly independent they represent a base of the space which in this particular case is a base of $\mathbb{R}^2$. Therefore any vector in the space can be found as a linear combination of the elements of the base.

An important conclusion is that any vector in the plane can be reproduced by these two vectors if we choose x_1 and x_2 properly. Notice that this result can also be extended to any two vectors different from $(2, 4)$ and $(4, 11)$ if and only if they are not parallel to each other. Why?

Row representation – (System of three equations)

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Every row represents a plane in three dimensional space. That way, for example, (a_{11}, a_{12}, a_{13}) is the vector normal to the plane represented by the first row. It can also be written as $A[1, :]$ using the Matlab colon notation. In the same way $(a_{21}, a_{22}, a_{23}) = A[2, :]$ defines another plane and so forth.

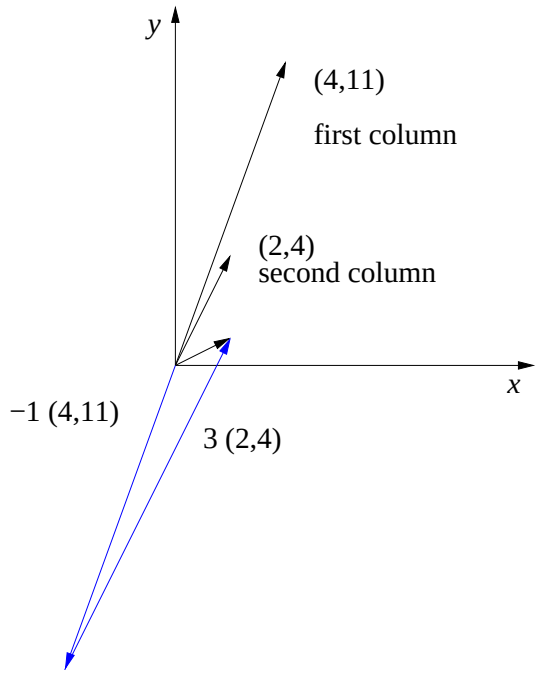


Figure 1.2: Column representation of a system of two equations.

The solution to the system of equations is given by the intersection of the planes. That is plane $A[1,:]$ intersects plane $A[2,:]$ in a line and that line intersects plane $A[3,:]$ in a point which is the solution to the system.

Column representation – (System of three equations)

In the same way as the $n = 2$ case, each matrix column represents a vector in three dimensional space. In Matlab colon notation the columns are written as

$$\begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \end{bmatrix} = A(:,1) \qquad \begin{bmatrix} a_{12} \\ a_{22} \\ a_{32} \end{bmatrix} = A(:,2) \qquad \begin{bmatrix} a_{13} \\ a_{23} \\ a_{33} \end{bmatrix} = A(:,3)$$

and the vector solution is obtained as a linear combination of these three vectors. That is, multiplying the column vectors $A\,[:,1]$, $A\,[:,2]$, and $A\,[:,3]$ by the scalar quantities x_1, x_2 and x_3 and then adding the result. See figure 1.3.

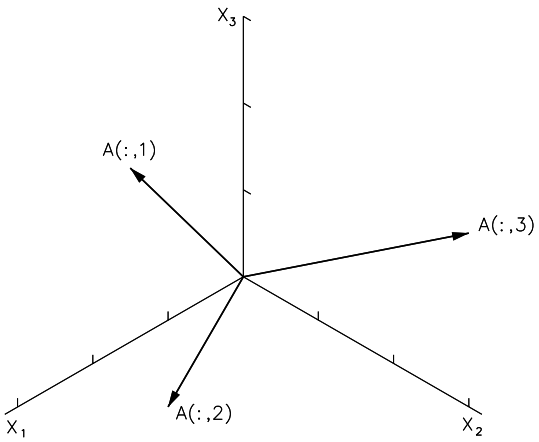


Figure 1.3: Column representation to the solution of a system of 3 linear equations

1.1.2 Singular cases

There are cases when the system does not have a solution. That can be interpreted from the geometrical point of view as follows. If we use row vector representation, then each matrix row represents a plane and the solution is the simultaneous intersection of the planes. If one of the planes is parallel to the intersection of the other two planes then the three planes never intersect in a point and therefore there is no solution. See figure 1.4(a). On the other hand, when we plot the column vectors of the same singular system, they will be lying in the same plane. See figure 1.4(b). This means the vectors do not form a base in $\mathbb{R}^3$.

1.1.3 Exercises

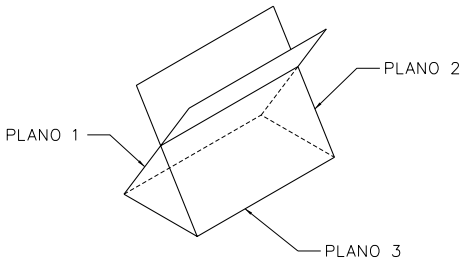
Matrix and vector operations

Assume $\lambda \in \mathbb{R}$, $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times n}$ then implement the following operations into computer code.

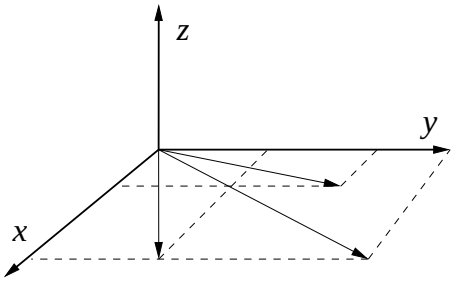
1. Scalar vector multiplication: $\mathbf{z} = \lambda \mathbf{x} \quad \Rightarrow \quad z_i = \lambda x_i$

2. Vector addition: $\mathbf{z} = \mathbf{x} + \mathbf{y} \quad \Rightarrow \quad z_i = x_i + y_i$

3. Dot product (inner product): $\lambda = \mathbf{x} \cdot \mathbf{y} = \mathbf{x}^T \mathbf{y} \quad \Rightarrow \quad \lambda = \sum x_i y_i$



(a) Row representation



(b) Column representation

Figure 1.4: Geometric representation of a singular system.

4. Vector multiplication: $\mathbf{z} = \mathbf{x} * \mathbf{y} \quad \Rightarrow \quad z_i = x_i\, y_i$

5. Scalar matrix multiplication: $B = \lambda A \quad \Rightarrow \quad B_{ij} = \lambda A_{ij}$

6. Matrix addition: $C = A + B \quad \Rightarrow \quad C_{ij} = A_{ij} + B_{ij}$

7. Matrix vector multiplication: $\mathbf{y} = A\mathbf{x} \quad \Rightarrow \quad \mathbf{y} = \sum_j a_{ij} x_j$

8. Matrix matrix multiplication: Let $A \in \mathbb{R}^{n\times p}$ and $B \in \mathbb{R}^{p\times m}$, then $C = AB \Rightarrow C_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj}$, and in colon notation $C_{ij} = A\,(i,:)\,B\,((:,j)$

1.2 Gauß Elimination

From the *geometric point-of-view* each equation (row) of a system of n equations represents a hyperplane of nth dimension. Finding the solution to the system is therefore equivalent to find the intersection of all the n hyperplanes. This procedure can be illustrated as follows: Eliminate one variable by intersecting two hyperplanes and the result is hyper-plane of $n-1$ dimension. Then, consecutively find the intersection of the resulting hyperplane with the next hyperplane. Every intersection will reduce the dimension of the resulting hyperplane by one. After $n-1$ operations, we will obtain a hyper-point of n dimension which is the solution of the system.

Gauß elimination consists of two main steps: forward elimination and backward substitution.

1.2.1 Forward elimination

The purpose of forward elimination is to reduce the set of equations to an upper triangular system. The process starts eliminating the coefficients of the first column from the second equation until the last equations. Then it eliminates the coefficient of the second column from the third equation and so forth until the $n-1$ column of the system. That way the last equation will have only one unknown. To eliminate a coefficient a_{ik} from the ith equation, equation k must be multiplied by $-1/a_{ik}$ and added to equation i. The first equation is called the pivot equation and the term a_{kk} is called the pivot coefficient.

1.2.2 Backward substitution

After the forward elimination, the original matrix is transformed into an upper triangular matrix. The last of the equations (n) will have only one unknown: $a_{n,n}\, x_n = b_n$. The unknown x_n is found and replaced back into the $n-1$ equation $a_{n-1,n-1}\, x_{n-1} + a_{n-1,n}\, x_n = b_n$ which now has only one unknown x_{n-1}. The equation is solved and the procedure is repeated until we reach the first equation and all the values are known.

Known Problems:

- During the process of forward elimination and backward substitution, a division by zero can be presented. In the same way, due to the computer arithmetics even if the number is not zero but close to zero, the same problem can be presented.
- Rounded off values can result in inexact solutions.
- Ill-conditioned system of equations where small changes in the coefficients give rise to large changes in the solution.
- Singular systems of equations.

1.2.3 Operations in a matrix

Let $A \in \mathbb{R}^{n\times n}$ and $\mathbf{x}, \mathbf{b} \in \mathbb{R}^n$. The system of n equations can be written in terms of A, $\mathbf{x}$, and $\mathbf{b}$ as

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \qquad \begin{matrix} (E_1) \\ (E_2) \\ \vdots \\ (E_n) \end{matrix}$$

where (E_i) denotes equation i of the system. The following operations can be accomplished without altering the result.

- A equation, E_i, can be multiplied by a scalar number λ, with $\lambda \neq 0$ and the resulting equation can replace E_i

$$(\lambda E_i) \rightarrow E_i$$

- A equation, E_j, can be multiplied by λ, and added to equation E_i

$$(E_i + \lambda E_j) \rightarrow E_i$$

- The order of the equation can be changed

$$(E_i) \leftrightarrow (E_j)$$

Example 1.1 Let's have the following system of two equations,

$$2x_1 + 4x_2 = 2 \tag{1.3}$$
$$4x_1 + 11x_2 = 1. \tag{1.4}$$

To eliminate the first variable we multiply equation (1.3) by $4/2$ and add the result to equation (1.4),

$$2x_1 + 4x_2 = 2$$
$$0 + 3x_2 = -3.$$

We arrive at the last equation then we solve for x_2 as $x_2 = -1$. The second step consists of replacing $x_2 = -1$ back into equation (1.3) and solving for x_1 (back substitution).

Example 1.2 Solve the following system of equations, represented in matrix form, using forward elimination and backward substitution:

$$\begin{bmatrix} 2 & 4 & 0 \\ 1 & 4 & 1 \\ 2 & 2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \\ 4 \end{bmatrix} \begin{matrix} (E_1) \\ (E_2) \\ (E_3) \end{matrix}.$$

Then we operate the system in the following way:

$E_2 \leftarrow E_2 - \frac{1}{2}E_1$,

$$\begin{bmatrix} 2 & 4 & 0 \\ 0 & 2 & 1 \\ 2 & 2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 4 \end{bmatrix}.$$

$E_3 \leftarrow E_3 - \frac{2}{2}E_1$,

$$\begin{bmatrix} 2 & 4 & 0 \\ 0 & 2 & 1 \\ 0 & -2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix}.$$

Notice that variable x_1 has been eliminated in all the equations (column one equals zero). Now we can proceed with column two.
$E_3 \quad \leftarrow \quad E_3 - E_2$,

$$\begin{bmatrix} 2 & 4 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix}.$$

the second step consists of back-replacing the unknowns. For the last equation we have:

$$x_3 = \frac{2}{3}.$$

then after successive replacements

$$2x_2 + \frac{2}{3} = 0 \qquad \Rightarrow \qquad x_2 = -\frac{1}{3}.$$

$$2x_1 + 4\left(\frac{-1}{3}\right) = 2 \qquad \Rightarrow \qquad x_1 = \frac{5}{3}.$$

Notice that the forward elimination transformed the original system $A\mathbf{x} = \mathbf{b}$ into an upper triangular system $U\mathbf{x} = \mathbf{c}$ with

$$\begin{bmatrix} 2 & 4 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 3 \end{bmatrix} = U \qquad y \qquad c = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix}$$

which has the same solution since equality was maintained by applying the same operations to A and $\mathbf{b}$ during the process.

1.2.4 Forward elimination – General case

Given the following set of n linear equations, $E1, \ldots, E_n$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \qquad (E_1) \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \qquad (E_2) \\ \vdots & \qquad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots\ldots\ldots\ldots a_{nn}x_n &= b_n \qquad (E_n) \end{aligned}$$

we want to find x_i for $i = 1 \ldots n$. The solution is obtained by applying forward elimination followed by back substitution. Forward elimination

transforms the original system into an upper triangular one. Algorithm 1 presents the forward elimination algorithm. For each i row it puts zeros in the positions underneath the diagonal (outer loop). To do this it divides equation E_i by a_{ii} so that the diagonal becomes one, that is: $a_{ii} = 1$. Then it multiplies E_i by $-a_{ji}$ (the value that we want to eliminate) and adds it to E_j. The resulting equation replaces E_j.

Algorithm 1 Forward Elimination

```
n is the dimension of the matrix                          (operations)
for i = 1 to n  do
   {divides ith equation (row) by a_ii so that a_ii = 1 }
   E_i ← E_i/a_ii                                          (n − i)
   {Put zeros in the ith column under the diagonal}
   for j > i  do
      E_j = E_j − a_ji E_i                                 (n − i)
   end for
end for
```

In order to have a measure of how long the algorithm takes to solve a system, we compute the total number of operations (multiplications, subtractions, divisions and sums). We can approximate this value by only counting the number of multiplications and divisions. For each cycle the operations per cycle op_i are $(n-i)$ divisions plus the number of operations of the j loop which is done $(n-i)$ times. The number of operations per j cycle is $(n-i)$, resulting from multiplying $a_{ji}E_i$. That is

$$\begin{aligned} op_i &= (n-i) + (n-i)(n-i+1) \\ &= (n-i)(1 + (n-i+1)) \\ &= (n-i)(n-i+2) \\ &= n^2 - 2in + i^2 + 2n - 2i \end{aligned}$$

which is the total number of operations per ith cycle. Remembering that

$$\sum^{m} 1 = m \qquad \sum^{m} j = \frac{m(m+1)}{2} \qquad \sum^{m} j^2 = m(m+1)(2m+1)$$

then the total number of operations after n cycles will be

$$\begin{aligned} op &= \sum_{i=1}^{n} (n^2 - 2in + i^2 + 2n - 2i) \\ &= n^2 \sum_{i}^{n} 1 - 2n \sum_{i}^{n} i + \sum_{i}^{n} i^2 + 2n \sum_{i}^{n} 1 - 2 \sum_{i}^{n} i \\ &= n^2 n - 2n \left(\frac{n(n+1)}{2} \right) + n(n+1)(2n+1) + 2nn - 2n \left(\frac{n+1}{2} \right) \\ &= 2n^3 + 3n^2. \end{aligned}$$

When n is large, the number of operations *op* will be of the order of $O(n^3)$.

1.2.5 Backward substitution – General case

After applying the forward elimination algorithm the matrix is transformed into an upper triangular system like this

$$U\mathbf{x} = \begin{bmatrix} U_{11} & \cdots & U_{1i} & \cdots & U_{1n} \\ & \ddots & & & \\ & & U_{ii} & \cdots & U_{in} \\ & 0 & & \ddots & \\ & & & & U_{nn} \end{bmatrix} \mathbf{x} = \mathbf{c}.$$

To solve an upper triangular system we solve/replace the values for $\mathbf{x}$ starting from the last equation and moving backwards. This way, for the last equation we have

$$U_{nn} x_n = c_n \Rightarrow x_n = {}^{c_n}/_{U_{nn}}$$

with x_n known, we replace this value into the $n-1$ row (equation)

$$U_{n-1,n-1}\, x_{n-1} + U_{n-1,n}\, x_n = c_{n-1}$$

and solve for x_{n-1}

$$\begin{aligned} U_{n-1,n-1} x_{n-1} &= c_{n-1} - U_{n-1,n} x_n \\ x_{n-1} &= \frac{c_{n-1} - U_{n-1,n} x_n}{U_{n-1,n-1}}. \end{aligned}$$

Now with x_n and x_{n-1} known, we can move up to the $n-2$ row and solve for x_{n-2}. This procedure can be repeated for equation $n-3$ and so forth. In general, suppose we are replacing in equation j. Equation j has the following form

$$U_j^T x = U_{jj}x_j + U_{j,j+1}x_{j+1} + \ldots\ldots\ldots\ldots + U_{jn}x_n = c_n.$$

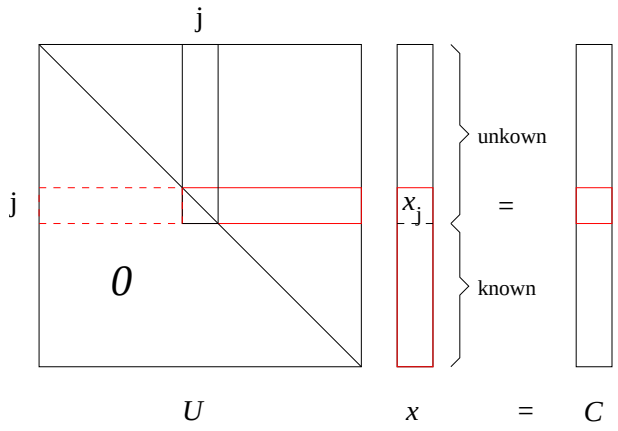


Figure 1.5: The operation of the jth row times vector x has the only unknown value of x_j

Notice that equation Ej is obtained by multiplying the jth row of the matrix by vector $\mathbf{x}$. Figure 1.5 illustrates this multiplication. At this stage, vector $\mathbf{x}$ consist of two parts. The upper consisting of unknown values and the lower known part. Then when $\mathbf{x}$ is multiplied by the jth row of the matrix, all the unknown values of the matrix are cancelled except the x_j term. This is because U is an upper triangular matrix and therefore $U_{jk} = 0$ $k < j$

$$\underbrace{U_{j,1}x_1 + \ldots +}_{0} U_{j,j}x_j + \underbrace{U_{j,j+1}x_{j+1} + \ldots + U_{jn}x_n}_{\text{x known}} = c_j$$

then solving for x_j

$$\begin{aligned} U_{j,j}x_j &= c_j - (U_{j,j+1}x_{j+1} + \ldots + U_{jn}x_n) \\ U_{j,j}x_j &= c_j - \sum_{k>j}^{n} U_{jk}x_k \\ x_j &= \frac{c_j - \sum_{k>j}^{n} U_{jk}x_k}{U_{j,j}}. \end{aligned}$$

This result is summarised in algorithm 2.

Algorithm 2 Backward Substitution

```
n is the dimension of the matrix
for j = n − 1 to 1 do
  temp = 0
  for k = j + 1 to n do
    temp = temp + U_jk x_k
  end for
  x_j = (C_j − temp)/U_jj
end for
```

The total number of multiplications and divisions will be

$$\begin{aligned} \sum_{j=n}^{1}(n-j)+1 &= n\sum_{j=n}^{1}1 - \sum_{j=n}^{1}j + \sum_{j=n}^{1}1 \\ &= nn - \frac{n(n+1)}{2} + n \\ &= n^2 - \frac{n^2}{2} - \frac{n}{2} + n \\ &= \frac{(n^2+n)}{2} \end{aligned}$$

which is of order $O(n^2)$, recalling that the first part of the algorithm was of order $O(n^3)$. This example shows the advantage of doing backward substitution instead of setting zeros in the upper triangular positions of the matrix (backward elimination). This result lead us to the conclusion that substitution must be preferred over elimination due to the order of complexity of the algorithm.

1.3 LU Decomposition

Backward substitution starts from an upper triangular system of equations and solves for $\mathbf{x}$ in $U\mathbf{x} = \mathbf{b}$. As we showed in the previous section this procedure is of a lower order of complexity than forward elimination. So if a system of equations can be written as

$$LUx = b, \tag{1.5}$$

with L lower triangular and U upper triangular matrices then it can be solved with two substitution procedures: i) Use backward substitution to find c from $Lc = b$ and ii) use forward substitution to find x from $Ux = c$. These two substitutions are of less order than one elimination plus one substitution. However not many systems can be easily expressed in terms of (1.5). In general, one can find a possible factorisation at the same time of the elimination procedure. This of course does not have any advantage in terms of efficiency because the factorisation itself is a $O(n^3)$ algorithm. Nevertheless, it can be reworded if we are solving a system with the same matrix but several b.

To illustrate the factorisation procedure, let's consider the following system of four equations:

$$Ax = \begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \\ 4 \\ 8 \end{bmatrix} = b.$$

Reducing the system to an upper triangular system is a straight-forward procedure because of the tridiagonal nature of the matrix where most of the terms are already set to zero. We proceed using forward elimination. To eliminate a_{21}, we multiply equation E_1 by $l_{21} = a_{21}/a_{11} = (1/2)$ and subtract it from equation E_2,

$$E_2 \leftarrow E_2 - \left(1/2\right) E_1 \Rightarrow \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 3/2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}.$$

Now to eliminate a_{32} we multiply equation E_2 by $l_{32} = a_{32}/a_{22} = (2/3)$

and subtract it from equation E_3,

$$E_3 \leftarrow E_3 - \left(2/3\right) E_2 \;\Rightarrow \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 3/2 & 1 & 0 \\ 0 & 0 & 4/3 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}.$$

Finally, to eliminate a_{43} we multiply equation E_3 by $l_{43} = a_{43}/a_{33} = (3/4)$ and subtract it from equation E_4

$$E_4 \leftarrow E_4 - \left(3/4\right) E_3 \;\Rightarrow \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 3/2 & 1 & 0 \\ 0 & 0 & 4/3 & 1 \\ 0 & 0 & 0 & 5/4 \end{bmatrix}.$$

The result is an upper triangular matrix. At the same time, multiply the matrix for vector $\mathbf{b}$. It is transformed into vector $\mathbf{c}$:

$$b_2 \leftarrow b_2 - l_{21}\, b_1 \Rightarrow \begin{bmatrix} 2 \\ 0 \\ 4 \\ 8 \end{bmatrix}, \quad b_3 \leftarrow b_3 - l_{32}\, b_2 \Rightarrow \begin{bmatrix} 2 \\ 0 \\ 4 \\ 8 \end{bmatrix},$$

$$b_4 \leftarrow b_4 - l_{43}\, b_3 \Rightarrow \begin{bmatrix} 2 \\ 0 \\ 4 \\ 5 \end{bmatrix} = \mathbf{c},$$

where the same multiples l_{ij} are used to operate the vector. The original problem $A\mathbf{x} = \mathbf{b}$ was transformed into an upper triangular system $U\mathbf{x} = \mathbf{c}$

$$\underbrace{\begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 3/2 & 1 & 0 \\ 0 & 0 & 4/3 & 1 \\ 0 & 0 & 0 & 5/4 \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} 2 \\ 0 \\ 4 \\ 5 \end{bmatrix}}_{\mathbf{c}}.$$

Notice that the operation involved to transform the vector involved the same multiples l_{ij} used to transform the matrix. If these multiples are put

into a matrix L, then the steps from $\mathbf{b}$ to $\mathbf{c}$ are exactly the same as solving $L\mathbf{c} = \mathbf{b}$. That is the matrix form of the forward elimination

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1/2 & 1 & 0 & 0 \\ 0 & 2/3 & 1 & 0 \\ 0 & 0 & 3/4 & 1 \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} 2 \\ 0 \\ 4 \\ 5 \end{bmatrix}}_{\mathbf{c}} = \underbrace{\begin{bmatrix} 2 \\ 1 \\ 4 \\ 8 \end{bmatrix}}_{\mathbf{b}}.$$

In conclusion, we started with a system $A\mathbf{x} = \mathbf{b}$ and then by forward elimination transformed it into $U\mathbf{x} = \mathbf{c}$. Both systems have the same solution since equality was maintained at each step by applying the same operators to A and $\mathbf{b}$.

As $\mathbf{c}$ is given by

$$L\mathbf{c} = \mathbf{b}$$

and

$$U\mathbf{x} = \mathbf{c}$$

then

$$U\mathbf{x} = L^{-1}\mathbf{b}$$

therefore

$$L\,U\,\mathbf{x} = \mathbf{b}.$$

Which implies the original matrix A was factorised into a lower L and an upper U, triangular matrix. A good Gauß elimination code consists of two main steps:

i. Factor A into L and U and

ii. Compute $\mathbf{x}$ from $LU\,\mathbf{x} = \mathbf{b}$.

Please note that we used a tridiagonal symmetric system only to simplify the computation. In general, this method applies to fully populated non-symmetrical matrices.

1.3.1 Cholesky Factorisation

Theorem 1.3.1. *If A is a symmetric positive definite matrix, then A can be factorised into $A = CC^T$, and C is a lower triangular matrix and C^T is its transpose.*

Proof. This serves at the same time to deduce the algorithm. To prove that $A = CC^T$ is equivalent to find C, we proceed using induction. First we show that we can calculate the values for the first column of C and then we suppose the known values of the $p-1$ columns and show we can compute the values for the pth column.
First Column

$$A_{i1} = \sum_{k=1}^{n} C_{pk} C_{k1}^T, \qquad p = 1, \ldots, n.$$

As C is lower triangular, that means $C_{ik} = 0$ for $k > i$ therefore

$$A_{i1} = C_{i1} C_{11}^T + C_{i2} C_{21}^T + \cdots + C_{in} C_{n1}^T$$

using $C_{ij} = C_{ji}^T$, we have

$$A_{i1} = C_{i1} C_{11} + C_{i2} C_{12} + \cdots\cdots + C_{in} C_{1n}.$$

Because C is lower triangular $C_{1k} = 0$, for $k > 1$, then all the terms to the right hand side of the equation, with the exception of the first, are cancelled.

$$A_{i1} = C_{i1} C_{11}$$

solving for C_{i1}

$$C_{i1} = \frac{A_{i1}}{C_{11}}$$

and for $i = 1$ we have

$$\begin{aligned} A_{11} &= C_{11} C_{11} \\ A_{11} &= C_{11}^2 \qquad \Rightarrow \qquad \boxed{C_{11} = \sqrt{A_{11}}}. \end{aligned}$$

Because A is positive definite, then $A_{11} > 0$. We can compute the values of the first column of C as,

$$\boxed{C_{i1} = \frac{A_{i1}}{\sqrt{A_{11}}}}.$$

Now suppose we know the values of the first $p-1$ columns of C, we want to show we can compute the values of the pth column. We accomplish this by expanding a component of A in the pth column. See figure 1.6. That is

$$A_{ip} = \sum_{k=1}^{n} C_{ik} C_{kp}^T$$

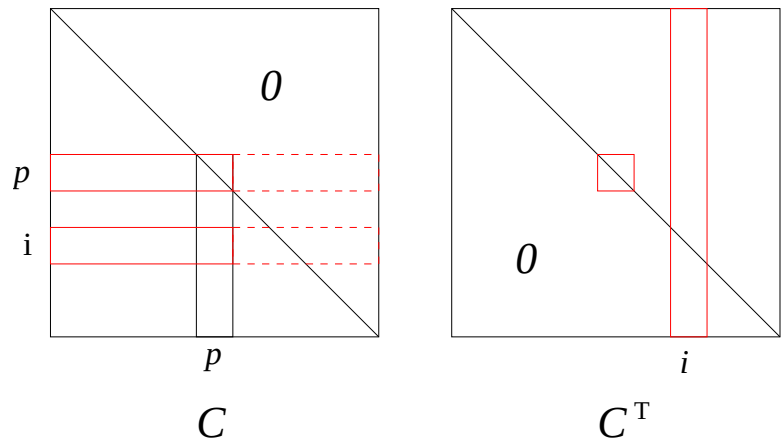


Figure 1.6: Cholesky factorisation.

by definition of C^T

$$A_{ip} = \sum_{k=1}^{n} C_{ik} C_{pk}$$

which is equivalent to multiplying rows p and i of matrix C. See figure 1.6. Notice that $C_{pk} = 0$ for $k > p$, therefore the upper limit of the sum changes is reduced from n to p

$$A_{ip} = \sum_{k=1}^{p} C_{ik} C_{pk}$$

expanding the last term

$$A_{ip} = \sum_{k=1}^{p-1} C_{ik} C_{pk} + C_{ip} C_{pp}$$

and solving for C_{ip}, we have

$$C_{ip} = \frac{A_{ip} - \sum_{k=1}^{p-1} C_{ik} C_{pk}}{C_{pp}} \tag{1.6}$$

where C_{pp} can be calculated making $p = i$, in the last equation

$$C_{pp} = \sqrt{A_{pp} - \sum_{k=1}^{p-1} C_{pk} C_{pk}}. \tag{1.7}$$

Equations 1.6 and 1.7 demonstrate that we can compute the matrix C for any A positive definite. □

1.4 Special Types of Matrices

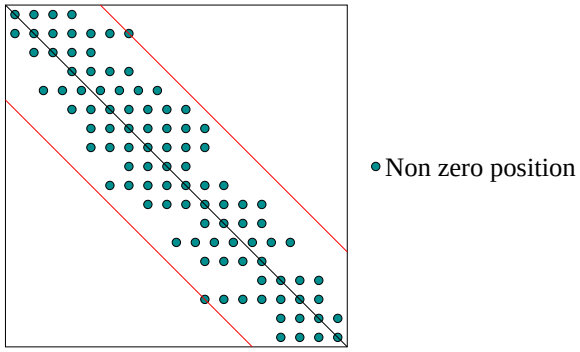


Figure 1.7: Banded matrix

Strictly dominant

A matrix $A \in \mathbb{R}^{n \times n}$ is strictly dominant in the diagonal sense if

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}|, \qquad \forall\ i = 1, \ldots, n.$$

Positive definite

A matrix $A \in \mathbb{R}^{n \times n}$ is positive definite if it is symmetric and $\mathbf{x}^t A \mathbf{x} > 0$, for all $\mathbf{x}$, with $\mathbf{x} \neq 0$, and $\mathbf{x} \in \mathbb{R}^n$.

Banded

A matrix A is banded if there exist integer numbers p and q, $1 < p,\ q < n$, such that $a_{ij} = 0$, always that $j \geq i + p$ and $i \geq j + q$. In other words the

non zero values of the matrix are around a diagonal by a distance of p. See figure 1.7.

Chapter 2

Iterative Methods

2.1 Vector Space

A vector space E is a set with two operations defined as addition and scalar multiplication. Additionally, if x, y, and z belong to E and α and β real numbers, then the following axioms are satisfied.

- **(E1)** $x + y = y + x$
- **(E2)** $(x + y) + z = x + (y + z)$
- **(E3)** There is an element in E, denoted by 0, such that $0 + x = x + 0 = x$
- **(E4)** For each x in E, there is an element $-x$ in E, such that $x + (-x) = (-x) + x = 0$
- **(E5)** $(\alpha + \beta)x = \alpha x + \beta x$
- **(E6)** $\alpha(x + y) = \alpha x + \alpha y$
- **(E7)** $(\alpha\beta)x = \alpha(\beta)x$
- **(E8)** $1 \cdot x = x$

2.2 Vector Norms

A vector norm in E is a function $\|\cdot\|$ defined in $E \rightarrow \mathbb{R}$ with the following properties: Let $\mathbf{x}$, $\mathbf{y} \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$ then

$$
\begin{aligned}
&i) \qquad N(\mathbf{x}) \geq 0, \quad \forall\, \mathbf{x} \in \mathbb{R}^n \\
&ii) \qquad N(\mathbf{x}) = 0, \quad \Leftrightarrow\ \mathbf{x} = (0, 0, \ldots, 0)^T = \mathbf{0} \\
&iii) \qquad N(\alpha\mathbf{x}) = |\alpha|\, N(\mathbf{x}) \\
&iv) \qquad N(\mathbf{x} + \mathbf{y}) \leq N(\mathbf{x}) + N(\mathbf{y}).
\end{aligned}
$$

2.2.1 Distance between two vectors

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ the distance $d \in \mathbb{R}$ between two vectors with respect to the norm $\|\cdot\|_*$ be defined as:

$$d = N(\mathbf{x} - \mathbf{y})$$

then d is referred to as a metric for $\mathbb{R}^n$.

Properties of a metric

Let $x, y, z \in E$ then

$$\begin{array}{ll} i) & d(x,y) \geq 0, \quad d(x,y) = 0 \text{ if and only if } x = y \\ ii) & d(x,y) = d(y,x) \\ iii) & d(x,y) \leq d(y,z) + d(z,y). \end{array}$$

2.2.2 Some norms in $\mathbb{R}^n$

The norms l_1, l_2, and l_∞, are defined as follows

$$\begin{aligned} \|\mathbf{x}\|_1 &= \sum |x_i|. \\ \|\mathbf{x}\|_2 &= \left\{ \sum x_i^2 \right\}^{1/2}. \\ \|\mathbf{x}\|_\infty &= \max_{1 \leq i \leq n} |x_i| . \end{aligned}$$

Example 2.1 Show that $\|x\|_\infty$ is a norm.

Properties i) to iii) are straightforward and are left as exercise. For iv) property we have

$$\begin{aligned} iv) \ \|x + y\|_\infty &= \max |x_i + y_i| \leq \max(|x_i| + |y_i|) \\ \|x + y\|_\infty &< \max |x_i| + \max |y_i| . \end{aligned}$$

Example 2.2 Find the l_1, l_2 and l_∞ norms of the following vectors

a. $x = (-1, 1, -2)^T$

b. $x = (3, -4, 0, 3/2)^T$

c. $x = (sen\, k, \cos k, 2^k)^T$, $k \in Z^+$.

For the a. case we have

$$\|x\|_2 = \sqrt{(-1)^2 + (1)^2 + (-2)^2} = \sqrt{6} \qquad \text{and}$$
$$\|x\|_\infty = \max\{|-1|, |1|, |-2|\} = 2.$$

Example 2.3 In an experiment, theory predicts that the solution to a problem is $\mathbf{x} = (1, 1, 1)$. But the experiment results are $\mathbf{x}_e = (1.001, 0.989, 0.93)$. Find the distance between the two results using l_∞ and l_2.
l_∞:

$$\begin{aligned} d &= \|(1,1,1) - (1.001, 0.989, 0.93)\|_\infty \\ &= \|(-0.001, 0.011, 0.07)\|_\infty \\ &= 0.07 \end{aligned}$$

l_2:

$$\begin{aligned} d &= \|(-0.001, 0.011, 0.07)\|_2 \\ &= \sqrt{-0.001^2 + 0.011^2 + 0.07^2} \\ &= 0.070866 \end{aligned}$$

2.2.3 The Cauchy-Buniakowsky-Shwarz inequality

This inequality states that for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ then

$$\sum |x_i y_i| \leq \left\{ \sum_{i=1}^{n} x_i^2 \right\}^{1/2} \left\{ \sum_{i=1}^{n} y_i^2 \right\}^{1/2}.$$

2.3 Convergence

A sequence of vectors $\left\{\mathbf{x}^k\right\}_{k=1}^{\infty}$ converge to $\mathbf{x}$ en $\mathbb{R}^n$, if $\forall \in > 0$, $\exists\, N(\in)$ such that

$$\left\| \mathbf{x}^k - \mathbf{x} \right\| < \epsilon, \qquad \forall\ k > N(\epsilon).$$

Theorem 2.3.1. *A sequence of vectors* $\{\mathbf{x}^k\}$ *converge to* $\mathbf{x} \in \mathbb{R}^n$ *with respect to* $\|\cdot\|_\infty$*, if and only if*

$$\lim_{k\to\infty} x_i^k = x_i, \quad \forall\ i \leq n.$$

Proof. This is left as an exercise to the reader. □

Example 2.4 Show that the following vector converges.

$$\mathbf{x}^k = (1,\ 2 + 1/k,\ 3/k^2,\ e^{-k}\sin(k))$$

According to the theorem, to show that the vector bx converges with respect to the norm l_∞ it is enough to show that each of its components converges. Then we have

$$\lim_{k\to\infty} 1 = 1, \quad \lim_{k\to\infty} 2 + 1/k = 2, \quad \lim_{k\to\infty} 3/k^2 = 0, \quad \lim_{k\to\infty} e^{-k}\sin(k) = 0$$

as each x_i converges we conclude that $\mathbf{x}$ converges.

2.3.1 Equivalent norms

Definition: Let N_1 and N_2 be two norms on a vector space E. These norms are equivalent norms if there exist positive real numbers c and d such that

$$c\ N_1(x) \leq N_2(x) \leq d\ N_1(x)$$

for all $x \in E$. An equivalent condition is that there exists a number $C > 0$ such that

$$\frac{1}{C}\ N_1(x) \leq N_2(x) \leq C\ N_1(x)$$

for all $x \in E$. To see the equivalence, set $C = \max\{1/c, d\}$.

Some key results are as follows:

i. On a finite dimensional vector space all norms are equivalent. The same is not true for vector spaces of infinite dimension [11].

ii. It follows that on a finite dimensional vector space, one can check the convergence of a sequence with respect to any norm. If a sequence converges in one norm, it converges in all norms.

iii. If two norms are equivalent on a vector space E, they induce the same topology on E [11].

Theorem 2.3.2. *Let* $\mathbf{x} \in \mathbb{R}^n$ *then the norms* l_2 *and* l_∞ *are equivalent in* $\mathbb{R}^n$

$$\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2 \leq \sqrt{n}\,\|x\|_\infty$$

Graphically this result is illustrated in figure 2.1 for the n=2 case.

Proof. Let us select x_j as the maximum component of $\mathbf{x}$, that is $x_j = \max|\,\mathbf{x}_i|$. It follows that

$$\|\mathbf{x}\|_\infty^2 = |x_j|^2 = x_j^2 \leq \sum_{i=1}^{n} x_i^2 = \|\mathbf{x}\|_2^2 .$$

Additionally,

$$\|\mathbf{x}\|_2^2 = \sum_{i=1}^{n} x_i^2 \leq \sum_{i=1}^{n} x_j^2 == x_j^2 \sum_{i=1}^{n} 1 = n x_j^2 = n\,\|\mathbf{x}\|_\infty^2 .$$

□

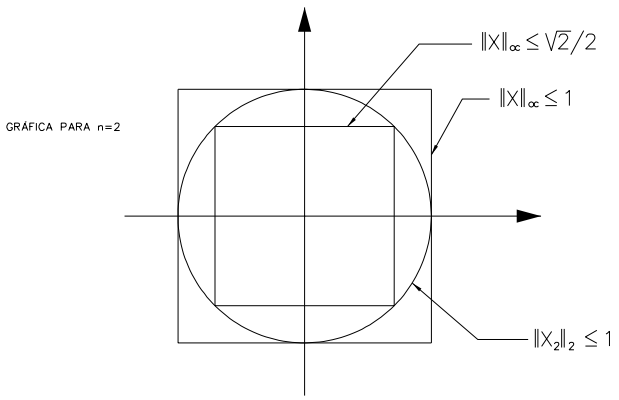


Figure 2.1: Equivalence of l_∞ and l_2 norms

Example 2.5 Given:

$$\mathbf{x}^k = (1,\ 2 + 1/k,\ 3/k^2,\ e^{-}k\sin(k))$$

Show that $\mathbf{x}^k$ converge to $\mathbf{x} = (1, 2, 0, 0)$ with respect to the norm $\|\cdot\|_2$.
Solution.

We already proved that this sequence of vectors converges with respect to the l_∞ norm. Therefore, given any $\varepsilon \in \mathbb{R}$, $\varepsilon > 0$, $\exists$ an integer $N(\varepsilon/2)$ with the property that

$$\left\|\mathbf{x}^k - \mathbf{x}\right\|_\infty < \varepsilon/2.$$

always that $k > N(\varepsilon/2)$ and using the result from theorem 2.3.2

$$\left\|\mathbf{x}^k - \mathbf{x}\right\| < \sqrt{4}\left\|\mathbf{x}^k - \mathbf{x}\right\|_\infty < 2\,(\varepsilon/2) = \varepsilon.$$

When $k > N(\varepsilon/2)$ therefore $\left\{\mathbf{x}^k\right\}$ converges to $\mathbf{x}$ with respect to $\|\cdot\|_2$.
Note: It can be shown that all the norms of $\mathbb{R}^n$ are equivalent with respect to the convergence.

2.4 Matrix Norms

Let A y $B \in \mathbb{R}^{n\times n}$, $\alpha \in \mathbb{R}$. A matrix norm in $\mathbb{R}^n$ is a function $\|\cdot\|$ defined in $\mathbb{R}^{n\times n} \to \mathbb{R}$ with the following properties:

$i)$ $\|A\| \geq 0$
$ii)$ $\|A\| = 0$ if and only if $A = 0$ zero matrix
$iii)$ $\|\alpha A\| = |\alpha|\,\|A\|$
$iv)$ $\|A + B\| \leq \|A\| + \|B\|$
$v)$ $\|AB\| \leq \|A\|\ \|B\|$

The distance between two matrices can be defined in the usual way as $d = \|A - B\|$.

2.4.1 Natural norms

A matrix natural norm is derived from vector norms. To understand how a vector norm can be used to define a matrix norm let us first observe the geometrical effect of matrix-vector multiplication. When a matrix A is multiplied by a vector x, the result is a new vector which is rotated and scaled in comparison with the original x vector. Figure 2.2 illustrates this

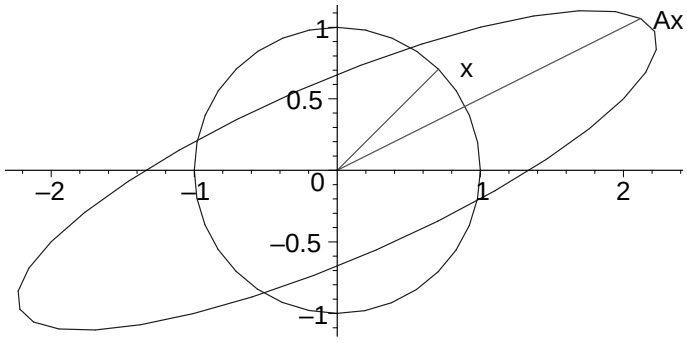


Figure 2.2: Geometrical effect of multiplying a vector $\mathbf{x}$ by a matrix A over a set of vectors with norm equal to one.

transformation for a series of vectors $\mathbf{x} \in \mathbb{R}^2$ with euclidean norm equal to one. This set of vectors represents a circle. When operator A is applied to the vectors the original circle is transformed into an ellipse. All vectors are scaled by different factors. If we choose a factor C large enough to be larger than the maximum scale factor, then we can affirm that

$$\|A\mathbf{x}\| \leq C \|\mathbf{x}\| .$$

If C is the smallest number for which the inequality holds for all $\mathbf{x}$, that is C is the maximum factor by which A can stretch a vector, then $\|A\|$ is defined as the supreme of C over all vectors

$$\|A\| = \sup C = \sup \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}$$

or equivalently

$$\|A\| = \sup_{\|\mathbf{x}\|=1} \|A\mathbf{x}\| .$$

Theorem 2.4.1. *If $\|\cdot\|$ is a vector norm in $\mathbb{R}^n$, then*

$$\|A\| = \max_{\|\mathbf{x}\|=1} \|A\mathbf{x}\|$$

is a matrix norm and is called natural norm.

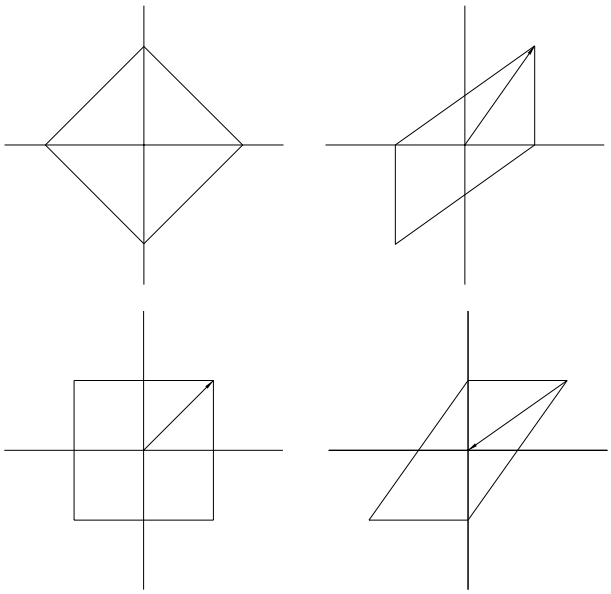


Figure 2.3: Examples of norms.

Proof. In the following proof we use Einstein notation to simplify the overuse of the summation symbol. In Einstein notation the symbol $\sum$ is suppressed and summation is assumed over the repeated indices. That is, $\sum_j a_{ij}\, x_j$ is equal to $a_{ij}\, x_j$.

i. $\|A\| > 0$,

$$\|A\mathbf{x}\| = \left\| \begin{array}{c} a_{1j}x_j \\ \vdots \\ a_{ij}x_j \\ \vdots \\ a_{nj}x_j \end{array} \right\| > 0, \text{ if and only if } a_{ij}x_j \neq 0$$

because $\|x\| \neq 0 \;\Rightarrow\; a_{ij}x_j = 0$, if and only if $a_{ij} = 0$

ii. $\|\alpha A\| = |\alpha|\, \|A\|$,

$$\|\alpha A\| = \|\alpha a_{ij}x_j\| = |\alpha|\, \|a_{ij}x_j\| = |\alpha|\, \|A\|$$

iii. $\|A+B\| \leq \|A\| + \|B\|$

$$\|A+B\| = \max_{\|\mathbf{x}\|=1} \|(a_{ij}+b_{ij})x_j\| = \max_{\|\mathbf{x}\|=1} \|a_{ij}x_j + b_{ij}x_j\|$$
$$\leq \max_{\|\mathbf{x}\|=1} (\|a_{ij}x_j\| + \|b_{ij}x_j\|) \leq \|A\| + \|B\|$$

□

Exercise

Prove that the following norms l_∞ and l_1 can be calculated using the following equations.

$$\|A\|_\infty = \max_i \sum |a_{ij}|$$
$$\|A\|_1 = \max_j \sum |a_{ij}|$$

2.5 Eigenvalues

For any square matrix A, we can look for vectors $\mathbf{x}$ that are in the same direction as $A\mathbf{x}$. These vectors are called eigenvectors. Multiplication by a matrix A normally changes the direction of a vector but for certain exceptional vectors, $A\mathbf{x}$ is a multiple of $\mathbf{x}$, that is

$$A\mathbf{x} = \lambda\mathbf{x}.$$

In this way, the effect of multiplying $A\mathbf{x}$ is to stretch, contract, or reverse $\mathbf{x}$ by a factor λ. This is illustrated in figure 2.4.

Example 2.6 The matrix:

$$A = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

has the following eigenvalues and eigenvectors:

$$\lambda_1 = 3,\ \mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix};\ \lambda_2 = 2,\ \mathbf{x}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\ \lambda_3 = 1,\ \mathbf{x}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

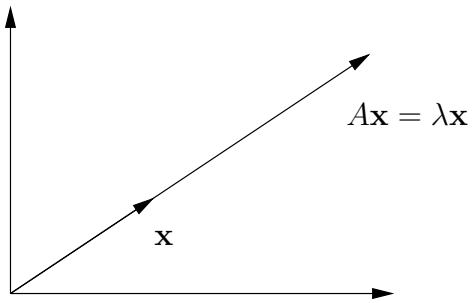


Figure 2.4: Multiplication of matrix A by special vector only changes its magnitude

Additionally, any vector in the space can be written as a linear combination of the eigenvectors (this is only if the eigenvectors are all different)

$$y = \alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \alpha_3\mathbf{x}_3$$

with $\alpha_i \in \mathbb{R}$ and $\mathbf{x}_i \in \mathbb{R}^3$. If we apply A to vector $\mathbf{y}$, we have:

$$\begin{aligned} A\mathbf{y} &= A(\alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3) \\ &= A\alpha_1 x_1 + \alpha_2 A x_2 + \alpha_3 A x_3 \\ &= \alpha_1\lambda_1 x_1 + \alpha_2\lambda_2 x_2 + \alpha_3\lambda_3 x_3. \end{aligned}$$

The action of A in any vector $\mathbf{y}$ is still determined by the eigenvectors.

Diagonal matrices are certainly the simplest. The eigenvalues are the diagonal itself and eigenvectors are in the direction of the Cartesian axes. For other matrices we find the eigenvalues and then the eigenvectors in the following form:

If $A\mathbf{x} = \lambda\mathbf{x}$, then $(A - \lambda I)\mathbf{x} = 0$, and because $\mathbf{x} \neq 0$ therefore $A - \lambda I$ has dependent columns and the determinant of $A - \lambda I$ must be zero, in other words, shifting the matrix by λI, it becames singular.

Example 2.7 Let

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

then

$$A - \lambda I = \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix}. \tag{2.1}$$

The eigenvalues of A are the numbers λ that make the determinant of $A - \lambda I$ equal to zero

$$\begin{aligned}\det(A - \lambda I) &= (2-\lambda)^2 - 1 = 0 \\ &= \lambda^2 - 4\lambda + 3 = 0 \\ &= (\lambda - 1)(\lambda - 3) = 0\end{aligned}$$

which leads to $\lambda_1 = 1$ and $\lambda_2 = 3$. Replacing $\lambda_1 = 1$ into equation 2.1

$$A - \lambda_1 I = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

because $A\mathbf{x}^{(1)} = 0$

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_x^{(1)} \\ x_y^{(1)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

which leads to,

$$\mathbf{x}^{(1)} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$\lambda_2 = 3 \;\Rightarrow$

$$A - \lambda_2 I = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}$$

and

$$\mathbf{x}^{(2)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

In the case when $n = 2$ the characteristic polynomial is quadratic and there is an exact formula for computing the roots. For $n = 3$ and $n = 4$ the characteristic polynomials are of order 3 and 4 for which there exist formulas for computing the roots. For $n > 4$ there is not (and there will never be) such a formula. Numerical methods must be used in those cases. See [7] for a detailed presentation of such algorithms.

2.5.1 Applications

2.6 Iterative Methods

2.6.1 Preliminary results

This section introduces some basic concepts essential to the understanding of the iterative methods in linear algebra.

Theorem 2.6.1 (Symmetric Matrix)**.** *If A is symmetric then*

$$\mathbf{y}^T A\mathbf{x} = \mathbf{x}^T A\mathbf{y}$$

Proof.

$$\begin{aligned}\mathbf{y}^T(A\mathbf{x}) &= y^T\left(\sum_j a_{ij}x_j\right)\\ &= \sum_i y_i \sum_j a_{ij}x_j\end{aligned}$$

Since $A = A^T$, $a_{ij} = a_{ji}$ then,

$$\begin{aligned}&= \sum_i \sum_j y_i a_{ji} x_j\\ &= \sum_j x_j \sum_i a_{ji} y_i\\ &= \mathbf{x}^T A\mathbf{y}\end{aligned}$$

which completes the proof. □

Positive definite matrix

A symmetric matrix $A \in \mathbb{R}^{n\times n}$ is said to be positive definite if

$$\mathbf{x}^T A\mathbf{x} > 0, \ \ \forall\, \mathbf{x} \in \mathbb{R}^n.$$

Theorem 2.6.2. *If A is a positive definite matrix, then the quadratic $P(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{x}^T\mathbf{b}$ is minimised at the point where $A\mathbf{x} = \mathbf{b}$. The minimum value is $P(A^{-1}\mathbf{b}) = -\frac{1}{2}\mathbf{b}^T A^{-1}\mathbf{b}$.*

Proof. For the case when $n = 1$, it is quite simple,

$$\begin{aligned}P(x) &= \frac{1}{2}ax^2 - bx\\ \frac{dP(x)}{dx} &= ax - b = 0\\ \therefore \qquad & ax = b.\end{aligned}$$

If a is positive, then $P(x)$ is a parabola which opens upward.

For $n > 1$, suppose $\mathbf{x}$ is the solution to $A\mathbf{x} = \mathbf{b}$, we want to show that at any point $\mathbf{y}$, $P(\mathbf{y})$ is larger than $P(\mathbf{x})$.

$$P(\mathbf{y}) - P(\mathbf{x}) = \frac{1}{2}\mathbf{y}^T A\mathbf{y} - \mathbf{y}^T\mathbf{b} - \frac{1}{2}\mathbf{x}^T A\mathbf{x} + \mathbf{x}^T\mathbf{b}$$

Replacing $A\mathbf{x} = \mathbf{b}$, we have

$$\begin{aligned} P(\mathbf{y}) - P(\mathbf{x}) &= \frac{1}{2}\mathbf{y}^T A\mathbf{y} - \mathbf{y}^T A\mathbf{x} - \frac{1}{2}\mathbf{x}^T A\mathbf{x} + \mathbf{x}^T A\mathbf{x} \\ &= \frac{1}{2}\mathbf{y}^T A\mathbf{y} - \mathbf{y}^T A\mathbf{x} + \frac{1}{2}\mathbf{x}^T A\mathbf{x} \end{aligned}$$

because

$$\mathbf{y}^T A\mathbf{x} = \frac{1}{2}\mathbf{y}^T A\mathbf{x} + \frac{1}{2}\mathbf{x}^T A\mathbf{y}$$

then,

$$\begin{aligned} P(\mathbf{y}) - P(\mathbf{x}) &= \frac{1}{2}\mathbf{y}^T A\mathbf{y} - \left(\frac{1}{2}\mathbf{y}^T A\mathbf{x} + \frac{1}{2}\mathbf{x}^T A\mathbf{y}\right) + \frac{1}{2}\mathbf{x}^T A\mathbf{x} \\ &= \frac{1}{2}\mathbf{y}^T(A\mathbf{y} - A\mathbf{x}) - \frac{1}{2}\mathbf{x}^T(A\mathbf{y} - A\mathbf{x}) \\ &= \frac{1}{2}\mathbf{y}^T A(\mathbf{y} - \mathbf{x}) - \frac{1}{2}\mathbf{x}^T A(\mathbf{y} - \mathbf{x}) \\ &= \frac{1}{2}A(\mathbf{y} - \mathbf{x})(\mathbf{y}^T - \mathbf{x}^T). \end{aligned}$$

Since A is positive the last expression can never be negative. It is equal to zero if $\mathbf{y} = \mathbf{x}$. Therefore $P(\mathbf{y})$ is larger than $P(\mathbf{x})$ and the minimum occurs at $\mathbf{x} = A^{-1}\mathbf{b}$

$$\begin{aligned} P_{\min} &= \frac{1}{2}(A^{-1}\mathbf{b})^T A(A^{-1}\mathbf{b}) - (A^{-1}\mathbf{b})^T\mathbf{b} \\ &= \frac{1}{2}(A^{-1}\mathbf{b})^T\mathbf{b} - (A^{-1}\mathbf{b})^T\mathbf{b} \\ &= -\frac{1}{2}(A^{-1}\mathbf{b})^T\mathbf{b}. \end{aligned}$$

In conclusion, minimising $P(\mathbf{x})$ is equivalent to solving $A\mathbf{x} = \mathbf{b}$. Figure 2.5 illustrates this result. $P(\mathbf{x})$ represents a paraboloid facing upward and with its minimum value at $\mathbf{x} = A^{-1}\mathbf{b}$. □

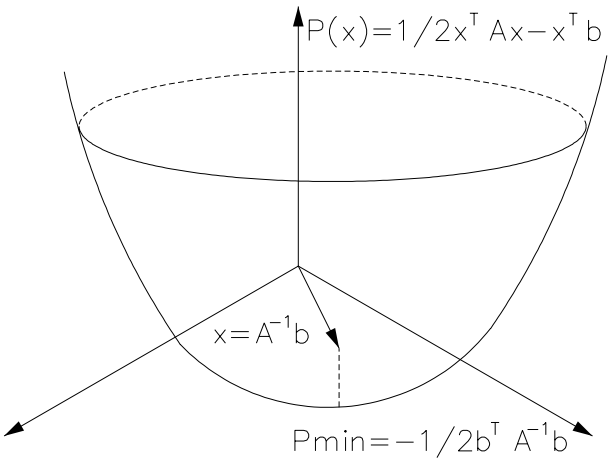


Figure 2.5: Two dimensional representation of $P(\mathbf{x})$

2.6.2 Steepest descent

One of the simplest strategies to minimise $P(\mathbf{x})$ is the steepest descent method. At a current point $\mathbf{x}_c$ the function $P(\mathbf{x})$ decreases most rapidly in the direction of the negative gradient $-\nabla P(\mathbf{x}_c) = \mathbf{b} - A\mathbf{x}_c$. We call $\mathbf{r}_c = \mathbf{b} - A\mathbf{x}_c$, the residual of $\mathbf{x}_c$. If the residual is non zero, then there exists a positive α such that $P(\mathbf{x}_c + \alpha\mathbf{r}_c) < P(\mathbf{x}_c)$. In the method of the steepest descent (with exact line search) we set $\alpha = \mathbf{r}_c^T\mathbf{r}_c/\mathbf{r}_c^T A\mathbf{r}_c$ thereby minimising $P(\mathbf{x}_c + \alpha\mathbf{r}_c)$.

To show this let us expand P at the point $(\mathbf{x}_c + \alpha\mathbf{r}_c)$:

$$\begin{aligned} P(\mathbf{x}_c + \alpha\mathbf{r}_c) &= \frac{1}{2}(\mathbf{x}_c + \alpha\mathbf{r}_c)^T A(\mathbf{x}_c + \alpha\mathbf{r}_c)^T - (\mathbf{x}_c + \alpha\mathbf{r}_c)^T\mathbf{b} \\ &= \frac{1}{2}\left[\mathbf{x}_c^T A(\mathbf{x}_c + \alpha\mathbf{r}_c) + \alpha\mathbf{r}_c A(\mathbf{x}_c + \alpha\mathbf{r}_c)\right] - (\mathbf{x}_c + \alpha\mathbf{r}_c)^T\mathbf{b} \\ &= \frac{1}{2}\left[\mathbf{x}_c^T A\mathbf{x}_c + \mathbf{x}_c^T A\alpha\mathbf{r}_c + \alpha\mathbf{r}_c^T A\mathbf{x}_c + \alpha^2\mathbf{r}_c A\mathbf{r}_c)\right] - \mathbf{x}_c^T\mathbf{b} - \alpha\mathbf{r}_c^T\mathbf{b}. \end{aligned}$$

Sorting terms we have:

$$P(\mathbf{x}_c + \alpha\mathbf{r}_c) = \frac{1}{2}\mathbf{x}_c^T A\mathbf{x}_c - \mathbf{x}_c^T\mathbf{b} + \frac{1}{2}\left[\alpha\mathbf{x}_c^T A\mathbf{r}_c + \alpha\mathbf{r}_c A\mathbf{x}_c + \alpha^2\mathbf{r}_c^T A\mathbf{r}_c\right] - \alpha\mathbf{r}_c^T\mathbf{b}.$$

Since A is symmetric

$$\begin{aligned} P(\mathbf{x}_c + \alpha \mathbf{r}_c) &= P(\mathbf{x}_c) + \frac{1}{2}\left[2\alpha \mathbf{x}_c^T A \mathbf{r}_c + \alpha^2 \mathbf{r}_c^T A \mathbf{r}_c\right] - \alpha \mathbf{r}_c^T b \\ &= P(\mathbf{x}_c) + \alpha \mathbf{x}_c^T A \mathbf{r}_c + \frac{1}{2}\alpha^2 \mathbf{r}_c^T A \mathbf{r}_c - \alpha \mathbf{r}_c^T \mathbf{b}. \end{aligned}$$

Replacing $\mathbf{b} = \mathbf{r}_c + A\mathbf{x}_c$

$$P(\mathbf{x}_c + \alpha \mathbf{r}_c) = P(\mathbf{x}_c) + \alpha \mathbf{x}_c^T A \mathbf{r}_c + \frac{1}{2}\alpha^2 \mathbf{r}_c^T A \mathbf{r}_c - \alpha \mathbf{r}_c^T (\mathbf{r}_c + A\mathbf{x}_c).$$

Expanding the last term and using the symmetry of A again

$$\begin{aligned} P(\mathbf{x}_c + \alpha \mathbf{r}_c) &= P(\mathbf{x}_c) + \alpha \mathbf{x}_c^T A \mathbf{r}_c + \frac{1}{2}\alpha^2 \mathbf{r}_c^T A \mathbf{r}_c - \alpha \mathbf{r}_c^T \mathbf{r}_c - \alpha \mathbf{r}_c^T A \mathbf{x}_c \\ &= P(\mathbf{x}_c) + \frac{1}{2}\alpha^2 \mathbf{r}_c^T A \mathbf{r}_c - \alpha \mathbf{r}_c^T \mathbf{r}_c \end{aligned}$$

which is minimum for

$$\frac{d}{d\alpha}\left(\frac{1}{2}\alpha^2 \mathbf{r}_c^T A \mathbf{r}_c - \alpha \mathbf{r}_c^T \mathbf{r}_c\right) = 0$$

that is

$$\alpha \mathbf{r}_c^T A \mathbf{r}_c = \mathbf{r}_c^T \mathbf{r}_c$$

therefore

$$\boxed{\alpha = \frac{\mathbf{r}_c^T \mathbf{r}_c}{\mathbf{r}_c^T A \mathbf{r}_c}}.$$

Chapter 3

Interpolation

3.1 Introduction

Suppose that as a result of an experiment a set of data points (x, u) related to each other is obtained. The relationship between x and u is expressed as $u(x)$ but from the experimental data we only know the values at certain points, that is, $u_i = u(x_i)$. For example suppose we have the following data:

u	1.29	1.74	2.38	3.19	4.03	4.65	4.55	3.04
x	1.00	1.90	2.80	3.70	4.60	5.50	6.40	7.30

This data is plotted in figure 3.1. The continuous line represents the function we want to interpolate. Unfortunately this function is usually unknown and the only information available is that presented in the table. The problem that rises is to evaluate the function $u(x)$ at intermediate data points x. That is $x_i < x < x_{i+1}$. Then we need to find a continuous function $\tilde{u}(x)$, that approximates $u(x)$ in such way that is exact at the data points $\tilde{u}(x_i) = u_i$.

3.1.1 Polynomial approximation

Polynomials are functions that provide advantages for approximating a set of points. They are continuous, derivable, and integrable. Furthermore, these operations can be accomplished and implemented straight-away.

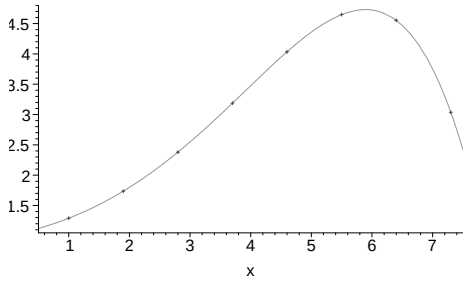


Figure 3.1: Approximation of a set of data points

In general, a polynomial of order n can be written as,

$$P_n(x) = a_n x^n + \ldots + a_0$$

with n a positive integer and $a_n, \ldots, a_0$ real coefficients. Its derivate

$$P'_n(x) = n a_n x^{n-1} + \ldots + a_1$$

and its integral

$$\int P_n(x)dx = \frac{a_n x^{n+1}}{(n+1)} + \ldots + a_0 x + C.$$

WEIERSTRASS theorem of approximation

Suppose that f is a function defined and continuous in the interval $[a, b]$. then for each $\varepsilon > 0$ exist a polynomial $P(x)$ defined over $[a, b]$ with the following property

$$|f(x) - P(x)| < \varepsilon \ \forall \ x \in .[a, b]$$

Figure 3.2 illustrate such a result.

Example 3.1 : *Taylor Polynomials.*

Taylor polynomials are very good to locally approximate a function around a point. Supouse that $f(x) \in C^n[a, b]$, that f^{n+1} exists in $[a, b]$, and that $x_0 \in [a, b]$. Then for all $x \in [a, b]$ at a distance $h = x - x_0$ it is true

$$f(x) = f(x_0+h) = f(x_0)+f'(x_0)(h)+\frac{f''(x_0)}{2!}(h)^2+\ldots+\frac{f^n(x_0)}{n!}(h)^n+O(h^{n+1})$$

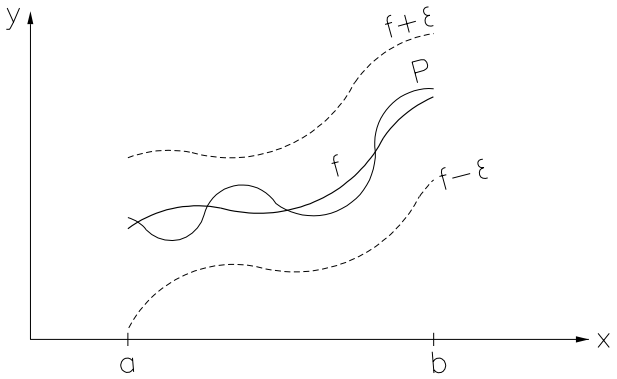


Figure 3.2: Weierstrass theorem of approximation

where $O(h^{n+1}$ is the error of the n–order approximation and is a function of h^{n+1}. Figure 3.3 plots $f(x) = cos(x)$ and the local Taylor approximations, of order zero and one, around point $x = 1$. For $n = 1$, $p0$ is a constant function defined by $p0(x) = \cos(1)$. For $n = 1$ $p1$ is a line given by $p1(x) = \cos(1) - \sin(1)(x - 1)$. Thus, when the order of the Taylor polynomial increases a better approximation of the function in the neighbourhood of $x = 1$ is obtained as it approximates not only the function but its derivatives. However, the error of approximation increases as we move away from $x = 1$.

3.2 Lagrange polynomials

Suppose that you want to find a first degree polynomial that passes through the points (x_0, u_0) y (x_i, u_i). There are many ways to find a straight line that passes through these points. The main idea behind Lagrange polynomials is to find a set of polynomials of one degree whose linear combination is equal to the desired polynomial. In this case, the line that joins the points (x_0, u_0) and (x_1, u_1) is given by the first degree polynomial $P(x)$ defined as,

$$P(x) = \frac{x - x_1}{x_0 - x_1} u_0 + \frac{x - x_0}{x_1 - x_0} u_1$$

which defines the line crossing the points (x_0, u_0) and (x_1, u_1) by adding the lines $P_1 = \frac{x-x_1}{x_0-x_1} u_0$ and $P_2 = \frac{x-x_0}{x_1-x_0} u_1$. If we define the functions $W_0(x)$

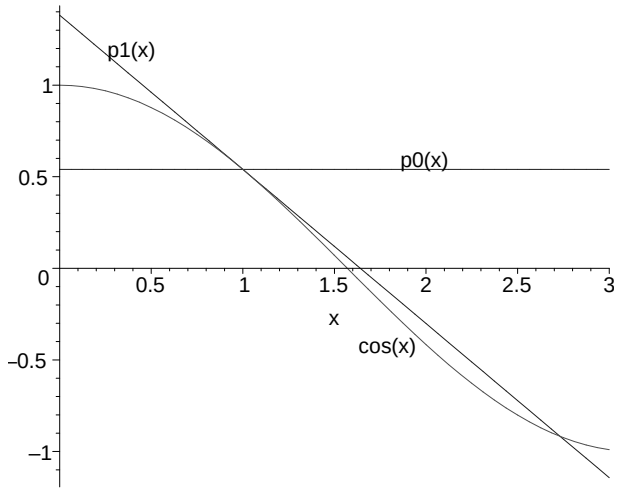


Figure 3.3: Taylor polynomial approximation of the cos function at point $x = 1$. p0(x), zero, and p1(x), fist, order approximations are shown.

and $W_1(x)$ as

$$W_0(x) = \frac{x - x_1}{x_0 - x_1}, \qquad W_1(x) = \frac{x - x_0}{x_1 - x_0}$$

then $P(x)$ can be written also as

$$P(x) = W_0(x)\, u_0 + W_1(x)\, u_1.$$

Notice that the W functions were found using the following rule

$$W_0(x_0) = 1, \qquad W_0(x_1) = 0.$$

and

$$W_1(x_0) = 0, \qquad W_1(x_1) = 1.$$

In this way the resulting polynomials passes through the given points (x_0, u_0) and (x_1, u_1).

3.2.1 Second order Lagrange polynomials

Suppose now that you have three points and you want to find the polynomial of second degree that *interpolates* those points. Using the technique

explained above, we want to find functions $W_0(x)$, $W_1(x)$, $W_2(x)$ such that

$$P(x) = W_0(x)\, u_0 + W_1(x)\, u_1 + W_2(x)\, u_2$$

with W_i second order polynomials. The result of adding second order polynomials is a second order polynomial. Additionally they must comply with

$$P(x_0) = u_0, \quad P(x_1) = u_1 \quad \text{and} \quad P(x_2) = u_2. \tag{3.1}$$

In other words the functions must go through (interpolate) the points.

In order to obtain the W_i functions we proceed in the same way as in the first order polynomials, that is, in order to guarantee the condition in equation 3.1 it is enough that functions W_i be defined as

$$W_i(x_j) = \begin{cases} 1 & \text{if} \quad i = j. \\ 0 & \text{if} \quad i \neq j. \end{cases} \tag{3.2}$$

That is $W_i(x_j)$ is equal to one at point x_i and zero at the other points. This function can be computed in different ways. For example to construct function W_0 that cancels at each x_i with $i \neq 0$, we choose a series of binomial factors each one cancelling at points x_i

$$W_0 = (x - x_1)(x - x_2). \tag{3.3}$$

Now, in order to satisfy $W_0(x_0) = 1$, we divide this result by the same product of binomials as in equation 3.3 and choosing $x = x_0$

$$W_0 = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}. \tag{3.4}$$

In this way W_0 complies with 3.2. In a similar way we can construct functions W_1 and W_2. Figure 3.4 shows these three functions. Notice that any quadratic function passing through points x_0, x_1 and x_2 can be constructed by the linear combination of these three functions.

3.2.2 General case

Given a set of $n + 1$ points describing a function, it can be interpolated by an n degree polynomial obtained by the linear combination of $n + 1$ polynomials of n degree with the following property:

$$W_i(x_k) = \begin{cases} 1 & \text{if} \quad i = k. \\ 0 & \text{if} \quad i \neq k \end{cases}$$

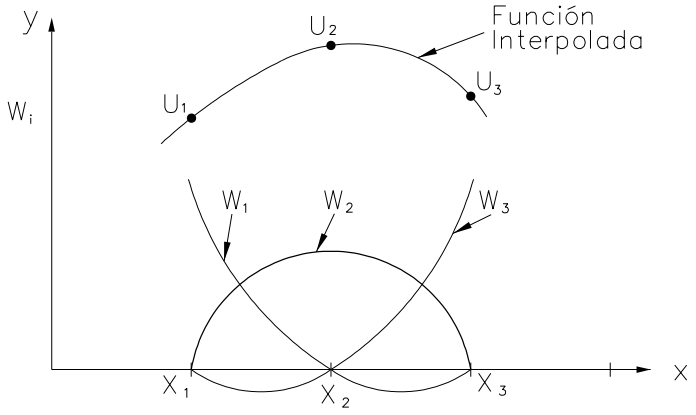


Figure 3.4: Second degree Lagrange polynomials

which can be constructed by:

$$W_i = \prod_{\substack{k=0 \\ k \neq i}}^{n} \frac{(x - x_k)}{(x_i - x_k)}$$

using Polynomial interpolation provides continuous and derivable functions. Besides derivation and integration of polynomials is straight-forward. However, when the order of the polynomials increases, the loop of the polynomials increases resulting in a poor local approximation. So high order polynomials must be avoided in the construction of interpolated functions. Instead, interpolation by parts must be considered. This will be discussed in the next section.

3.2.3 Other Approximations

Lagrange polynomials are not the only way of interpolating a set of data points. In general if v is a function defined in n points x_i by v_i with

$$v_i = v(x_i)$$

then the v can be interpolated by $\tilde{v}(x) \in V^n$ defined in terms of $W_i(x)$ functions with

$$\tilde{v}(x) = \sum_i v_i W_i(x).$$

The space of functions generated by a base of P^2 is given by

$$span\left\{W_1(x), W_2(x), W_3(x)\right\}.$$

Example 3.2 Given the functions $W_1 = 1$, $W_2 = x$, $W_3 = x^2$, any polynomial $p \in P^2$ is given by the linear combination of

$$p = a_1 W_1 + a_2 W_2 + a_3 W_3.$$

For example find a_1,a_2 and a_3 such as p is equal to $p(x) = -3x^2 + 2x + 5$

3.3 Polynomials defined by parts

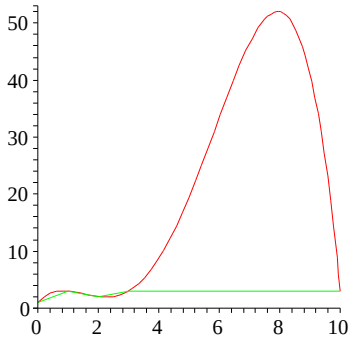


Figure 3.5: Oscillation due to high order polynomial approximation

One disadvantage of the polynomial interpolation is that as the number of points increases the order of the polynomial increases. A higer degree of polynomial does't necessarily means a good approximation as it can introduce undesirable oscilations as is shown in figure 3.5. To avoid this the domain can be subdivided and the function can be approximated by a series of function in each subdamain.

Let Ω define the domain of a function u defined as,

$$u : \Omega \rightarrow \mathbb{R}$$

Ω can be expressed in terms of subdivisions of the domain such as

$$\Omega = K_1 \cup K_2 \cup K_3 \ldots\ldots\ldots \cup K_n$$

where K_i represents a finite element and K_i intersects K_j at the maximum at the boundary.

Next we will show some examples of domains defined by parts for the one-, two-, and three-dimensional case.

One dimensional case

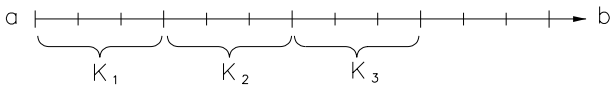


Figure 3.6: One dimensional domain defined by parts

Figure 3.6 presents a one-dimensional domain defined by parts. The segment $[a, b]$ is subdivided as a set of elements K_i. Each element consist of a segment of line and in this particular case they have four nodes per element. Additionally, the union of the elements is equal to the domain $\sum K_i = \Omega$ and the element only intersects at one point,

$$K_i \cap K_j = \begin{cases} \emptyset. \\ \text{vertex} \end{cases}$$

Two dimensional case

Figure 3.7 presents a two dimensional domain defined by parts. The domain is subdivided as a set of triangles K_i. Each triangle consist of three vertex and three line segments. In this particular case the nodes are defined at the vertex of the triangle. The union of the elements is equal to the domain $\sum K_i = \Omega$ and the element only intersects at one point or at an edge,

$$K_i \cap K_j = \begin{cases} \emptyset \\ 1\, vertex. \\ 1\ edge \end{cases}$$

Three dimensional case

Figure 3.8 presents a three dimensional domain defined by parts. The domain is subdivided as a set of tetrahedral K_i. Each tetrahedral consist

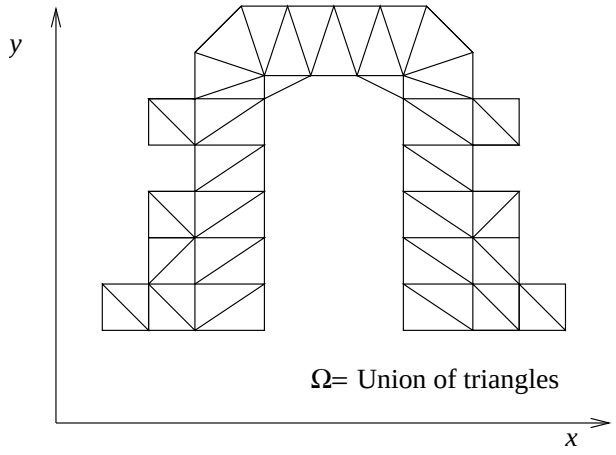


Figure 3.7: Two dimensional domain defined by parts

of four vertex, four three-dimensional triangles and six line segments. In this particular case the nodes are defined at the vertex of the triangles. The union of the elements is equal to the domain $\sum K_i = \Omega$ and the element only intersects at one face, edge or point,

$$K_i \cap K_j = \begin{cases} \emptyset \\ 1\, vertex. \\ 1\ edge \\ 1\ face \end{cases}$$

Base (Shape) functions

Base functions are defined for the elements k_i in the following way:

$$P^n = \left\{ u \in P^n(K_i) \,\middle|\, u(x) = \sum_{i=0}^{n} a_i x^i \right\}. \tag{3.5}$$

So in two dimensions we have

$$P^n = \left\{ u \in P^n(K_i) \,\middle|\, u(x) = \sum_{i+j \leq n}^{n} a_{ij} x^i y^j \right\}.$$

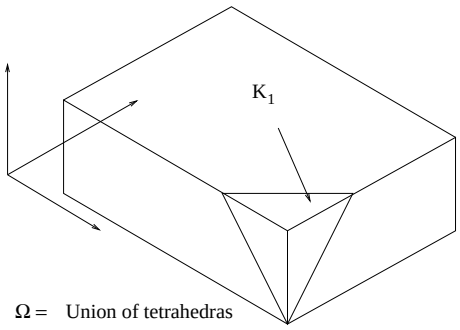


Figure 3.8: Three dimensional domain defined by parts

It can be observed that equation 3.5 can be written in terms of the base function of P^n as:

$$\begin{aligned} u_1 &= W_1(x_1) \\ u_2 &= W_2(x_2). \\ &\vdots \\ u_n &= W_n(x_n) \end{aligned}$$

Expanding the terms of the base and remembering that $W_i \in P^n$, we have:

$$\begin{aligned} W_1(x_1) &= a_0^1 x_1 + a_1^1 x_1^1 + \cdots + a_n^1 x_1^n = u_1 \\ &\vdots \\ W_i(x_i) &= a_0^i x_i + a_1^i x_1^i + \cdots + a_n^i x_i^n = u_i \end{aligned}$$

$W_i(x_i)$, can be written as:

$$[a]\,[x] = [u]$$

Notice that these functions are defined for each element $k_i \subset \Omega$. Therefore the limits of the elements can not be arbitrarily chosen. They must have a common node in such a way that continuity in the function is guaranteed. Therefore, if x_i is a common node for two elements and $u_r(x_i)$ is the function defined in element K_r and $u_s(x_i)$ is the function defined over element K_s, then it must be true that

$$u_r(x_i) = u_s(x_i).$$

3.3.1 One-dimensional Interpolation

First order polynomials

We choose finite elements of different size. We have:

$$x = \left\{ \begin{array}{c} x_1 \\ \vdots \\ x_m \end{array} \right\} \qquad u = \left\{ \begin{array}{c} u_1 \\ \vdots \\ u_m \end{array} \right\}$$

With this information it can be found that $P^n(x) = \tilde{u}(x)$ that interpolates function u at the m points. Suppose that you want to know the value of the function at certain point x in the domain. Proceed as follows:

i. Find k_i such that $x \in k_i$

ii. Compute the base functions $W\,|_{k_i}$

iii. Compute $u(x)$ as $u(x) = \sum u_i W^i(x)$.

Differential Operators

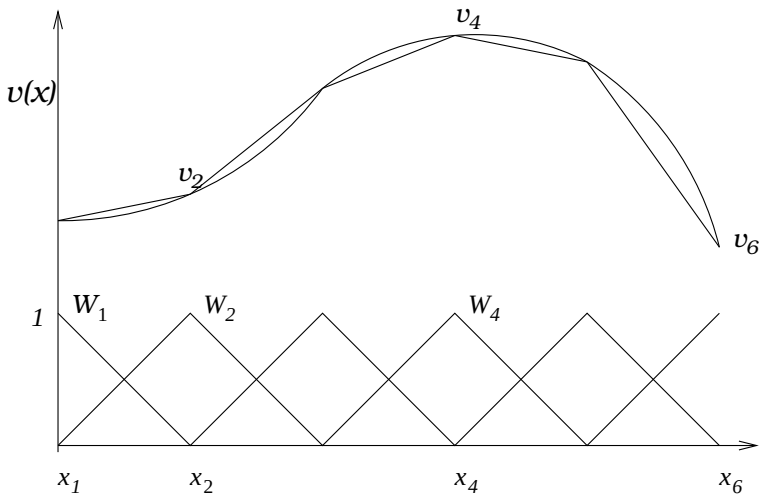


Figure 3.9: Piece-wise linear approximation of a smooth function

Let v be a function known at n points x_i. That is $v_i = v(x_i)$ is known for n points. Then the function is interpolated in V as

$$v(x) = \sum_i v_i W_i(x)$$

where the functions W_i form the basis of V. Figure 3.9 plots a function and its interpolation using five elements. The derivate of $v \in V$ is given by

$$\frac{dv}{dx} = \sum \frac{d}{dx} v_i W_i(x) = \sum v_i \frac{dW_i(x)}{dx}$$

Notice that while the continuity in the interpolated function is guarantied by the interpolation its derivate in only derivable by parts as at the node points the derivate is not well defined. One could try to interpolate again this result but in general the derivate of the interpolated function is different from the interpolated of the derivate.

Second order polynomials

Definition of the base functions :

$$W_i(x_j) = \begin{cases} 1 & \text{if} \quad i = j. \\ 0 & \text{if} \quad i \neq j \end{cases}$$

Let us analyse the meaning of having a domain defined by parts. First we take a data point inside an element, for example point $x_4 \in k_2$. Notice that W_4 is completely defined in element k_2 Outside of it takes a value of zero. However, when we look at function W_3 at point x_3, belonging to elements k_1 and k_2, we notice that the function is defined over these two elements. See figure 3.10.

Example

Figure 3.11 shows a second order polynomial interpolating u by parts. Notice that each element consists of three nodes and therefore a second order polynomial can be obtained at each element.

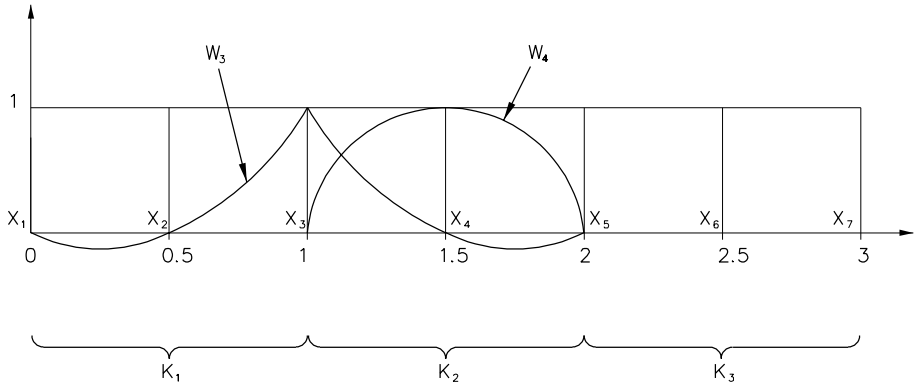


Figure 3.10: Piece-wise quadratic interpolation using Lagrange polynomials

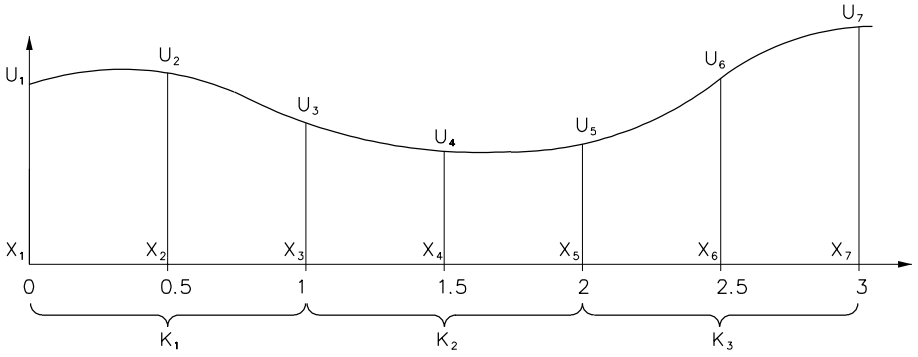


Figure 3.11: Interpolation using a second order polynomial defined by parts

Exercises

i. Using Matlab or Maple, draw function $u(x) = (x/5)\sin(5x) + x$ defined over the interval [0,4].

ii. Using the same partition of the domain of the last example, plot $\tilde{u}(x)$ second order Lagrange polynomials that interpolates $u(x)$.

iii. Global numeration Vs local numeration. From the last example we see the need of having a global numeration of the domain nodes but in order to compute the base functions it is convenient to have a local numeration.

3.3.2 Two dimensional interpolation

Let Ω be the domain of a function $u(x, y)$; Ω is bounded by a polygonal and $\Omega = \cup K_i$ (the union of subdomains where K_i can be a triangle or a square. The space of approximation $P^n(x_i)$ is defined for each element K_i.

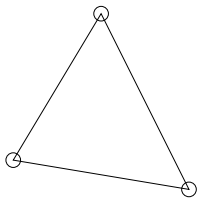
Example

Approximation space with P^2

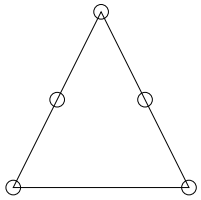
$$P \in P^2 \qquad P(x) = \sum_{i+j \leq n} a_{ij} x^i y^j.$$

The number of nodes needed to define a polynomial is given by the number of coefficients of the polynomial. That way

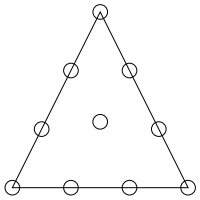
P1 N(1) = 3



P2 N(1) = 6



P3 N(3) = 10



.

Bases in the 2-D:

$$P^1 = span\left\{W_1(x), W_2(x), W_3(x)\right\}$$
$$P^2 = span\left\{W_1(x), \ldots, W_6(x)\right\}.$$

In general $W_r(x)$ is defined as:

$$W_r(x) = \sum_{i+j<n} a_{ij}^r x^i y^j$$

and the coefficients a_{ij}^r can be calculated using the following definition

$$W_r(x_s) = \delta_{rs}.$$

To guarantee continuity care must be taken at nodes belonging to several elements. The function defined over those element and evaluated at the common node must be the same.

Chapter 4

The Finite Element Method

4.1 Classification of the Partial Differential Equations PDE

The general form of a partial differential equation (PDE) with two independent variables, $u = u(x, y)$, defined over a two dimensional domain Ω, is:

$$a\frac{\partial^2 u}{\partial x^2} + 2h\frac{\partial^2 u}{\partial x \partial y} + b\frac{\partial^2 u}{\partial y^2} + f = 0 \tag{4.1}$$

where a, h, and b represent real constants or functions of x and y, and f is a function of $\partial u/\partial x$, $\partial u/\partial y$ and u. This general form (equation 4.1) is quite similar to the general equation of a conic

$$\mathrm{a}x^2 + 2\mathrm{h}xy + \mathrm{b}y^2 + 2\mathrm{c}x + 2\mathrm{d}y + \mathrm{e} = 0 \tag{4.2}$$

which represents an ellipse when $(ab - h^2 > 0)$, a parabola when $(ab - h^2 = 0)$ or hyperbole when $(ab - h^2 < 0)$.

In the same way, the *PDE*'s are classified. A *PDE* is:

Elliptic	if	$ab - h^2 > 0$
Parabolic	if	$ab - h^2 = 0$
Hyperbolic	if	$ab - h^2 < 0$.

4.1.1 Examples

i. Diffusion Equation

$$\alpha^2\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t} \tag{4.3}$$

where t represents the time and α is the diffusion coefficient of the material. (Notice that $y = t$ is in this equation.)

Then $a = \alpha^2$, $h = 0$, and $b = 0$; so therefore $ab - h^2 = 0$. We conclude that diffusion equation is parabolic.

ii. Wave Equation

$$\alpha^2 \frac{\partial^2 u}{\partial x^2} = \frac{\partial^2 u}{\partial t^2} \tag{4.4}$$

So $a = \alpha^2$, $h = 0$, and $b = -1$; so therefore $ab - h^2 = -\alpha^2 < 0$. This is the wave equation is hyperbolic.

iii. Laplace Equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \tag{4.5}$$

So $a = 1$, $h = 0$, and $b = 1$; so therefore $ab - h^2 = 1 > 0$ the Laplace equation, is elliptic.

In the last examples a, b, and h were real constants but in general they can be functions of x and y (equations with variable coefficients) and they can change type depending on the domain. (That is, they can change type when they change from one region of the $x - y$ plane to the other.

iv. The Tricomi Equation

One important application is the perturbation in the air caused by the displacement of a wing. The usual way of solving the problem is to keep the wing still and move the air (wind) around the wing.

For a non-viscous fluid

$$q = u\hat{i} + \nabla\phi$$

where q is the velocity and ϕ is the potential

$$\left(1 - M^2\right) \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \tag{4.6}$$

where M is known as the *Mach number*. That is the ratio between u and the speed of the sound.

h

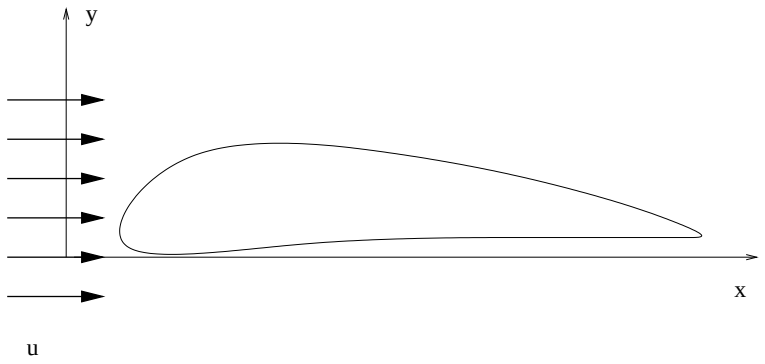


Figure 4.1: A wing profile for the tricomi equation

For the subsonic case we have $M < 1$

$$\begin{aligned} a &= (1 - M^2) \\ b &= 1 \\ ab - h^2 &= (1 - M^2) \end{aligned}$$

because the fluid is subsonic $M < 1$, so

$$ab - h^2 = (1 - M^2) > 0 \tag{4.7}$$

and therefore equation (4.6) is elliptic. If the fluid is incompressible, the speed is infinitum hence, $M = 0$ and the equation (4.6) results in the Laplace equation.

For the supersonic case $M > 1$

$$\begin{aligned} a &= (1 - M^2) \\ b &= 1 \\ ab - h^2 &= (1 - M^2) < 0 \end{aligned}$$

and equation (4.6) is hyperbolic.

When $M \approx 1$, that is transonic fluid. This case is more interesting and difficult to solve. We can not neglect the nonlinear part when obtaining the equation. After some change of variables we get

$$\frac{\partial^2 u}{\partial \xi^2} + \frac{\partial^2 u}{\partial \eta^2} = 0. \tag{4.8}$$

Now $ab - h^2 = (1)(\xi) = \xi$ so equation (4.6) is elliptic when $\xi > 0$ and hyperbolic when $\xi < 0$. This situation is reflected in the fact that the flow is mixed in the original variables x and y.

4.2 Boundary Value Problems

To determine a unique solution for equation (4.1) we need to specify the boundary conditions.

Let Ω be an open subset of the plane or space and Γ be the boundary of the region. The boundary conditions can be classified as (let $u = u(x, y)$)

$$u|_\Gamma = g \qquad \text{Dirichlet} \tag{4.9}$$

$$\left.\frac{\partial u}{\partial n}\right|_\Gamma = (\nabla u \cdot n)_\Gamma = g \qquad \text{Newman} \tag{4.10}$$

$$\left(\alpha u + \beta \frac{\partial u}{\partial n}\right)_\Gamma = g \qquad \text{Fourier.} \tag{4.11}$$

One problem could present one or more of these conditions in one or more parts of its boundary Γ.

4.2.1 One dimensional boundary problems

Let us have the following second order differential equation with boundary conditions defined over a closed domain $[a, b] \in \mathbb{R}$

$$\frac{d}{dx}\left(k\frac{du}{dx}\right) = f(x)$$

where f(x) is a known function and k is a constant. The problem can have one of the following boundary conditions:

$$u(a), u(b) \qquad \text{Dirichlet Boundary conditions}$$

$$k\left.\frac{du}{dx}\right|_a, k\left.\frac{du}{dx}\right|_b \qquad \text{Von Newman Boundary conditions}$$

$$\alpha u(b) + \beta\, k\left.\frac{du}{dx}\right|_b \qquad \text{Fourier Boundary conditions.}$$

Depending on the problem and its governing equations, the boundary conditions have different physical meaning.

Heat transfer: u represents the temperature

$$\begin{aligned}
&\text{Dirichlet} && \begin{cases} u_a \rightarrow \text{temperature at } u(a) \\ u_b \rightarrow \text{temperature at } u(b) \end{cases} \\
&\text{Von Newman} && \left\{ k\frac{\partial u}{\partial x}\Big|_a^b \rightarrow \text{heat flow} \right. \\
&\text{Fourier} && \left\{ -k\frac{\partial u}{\partial x} = h\left(T_w - T_\infty\right) \rightarrow \text{convection} . \right.
\end{aligned}$$

Elasticity: Equation of the elastic curve for beams. u represents the vertical displacement of a point in the neutral surface [2]pp:481

$$\frac{d^2u}{dx^2} = \frac{M(x)}{EI}$$

$$\begin{aligned}
&\text{Dirichlet} && \begin{cases} u_0 \rightarrow \text{displacement at } u(x_0) \\ u_n \rightarrow \text{displacement at } u(x_n) \end{cases} \\
&\text{Von Newman} && \left\{ \frac{\partial u}{\partial x}\Big|_a^b \rightarrow \text{slope} . \right.
\end{aligned}$$

Fluid mechanics In determining Potential flow, $u = \Psi$ represents the potential or stream function and the velocity is defined by

$$v_x = \frac{\partial \Psi}{\partial x} \qquad v_y = \frac{\partial \Psi}{\partial y}$$

with the boundary conditions:

$$\text{Von Newman} \left\{ v_i = \frac{\partial \Psi}{\partial x_i}\Big|_{\text{surface}} \rightarrow \text{velocity at the boundary} . \right.$$

In the following sections we will analyse the solution to the boundary problem with Dirichlet, Von Newman, and Fourier conditions. Also it will be considered the boundary problem with mixed conditions; that is, the Dirichlet boundary condition at a and Von Newman at b.

4.3 Bilinear Operators

Let

$$a\,(u,w):(u,w)\to\mathbb{R}$$

and let $\alpha,\beta\in\mathbb{R}$

$$a\,(\alpha u_1+\beta u_2,w)=\alpha a\,(u_1,w)+\beta a\,(u_2,w)$$

and in similar way

$$a\,(u,\alpha w_1+\beta w_2)=\alpha a\,(u,w_1)+\beta a\,(u,w_2)\,.$$

As an exercise, show that $a(u,v)$ in (4.29) is a bilinear operator.

4.4 Variational Formulation

Before formulating linear elliptic problems as variational problems, we first present the following abstract result.

Theorem 4.4.1 (Characterisation Theorem)**.** *Let V be a linear space, and suppose $a:V\times V\to R$ is a symmetric positive bilinear form, i.e., $a(v,v)>0$ for all $v\in V$, $v\neq 0$. In addition, let*

$$\ell:V\to R$$

be a linear functional. Then the quantity

$$J(v)=\frac{1}{2}a(v,v)-\ell(v)$$

attains its minimum over V at u if and only if

$$a(u,v)=\ell(v)\ \textit{for all}\ v\in V. \tag{4.12}$$

Moreover, there is at most one solution of (4.12).

Proof. For $u,v\in V$ and $t\in R$, we have

$$\begin{aligned}J(u+tv)&=\frac{1}{2}a(u+tv,u+tv)-\ell(u+tv)\\&=J(u)+t[a(u,v)-\ell(v)]+\frac{1}{2}t^2a(v,v).\end{aligned}\tag{4.13}$$

If $u \in V$ satisfies (4.12), then (4.13) with $t = 1$ implies

$$\begin{aligned} J(u+v) &= J(u) + \frac{1}{2}a(v,v) \quad \text{for all } v \in V \\ &> J(u), \quad \text{if } v \neq 0. \end{aligned} \tag{4.14}$$

Thus, u is a unique minimal point. Conversely, if J has a minimum at u, then for every $v \in V$, the derivative of the function $t \longmapsto J(u+tv)$ must vanish at $t = 0$. By (4.13) the derivative is $a(u,v) - \ell(v)$, and (4.12) follows. □

The relation (4.13) describes the size of J at a distance v from a minimal point u.

4.4.1 Reduction to Homogeneous Boundary Conditions

In the following, let L be a second order elliptic partial differential operator with divergence structure

$$Lu = -\sum \partial_i(a_{ik}\partial_k u) + a_0 u, \tag{4.15}$$

where

$$a_0(x) > 0 \quad \text{for} \quad x \in \Omega.$$

We begin by transforming the associated Dirichlet problem

$$\begin{aligned} Lu &= f \quad \text{in} \quad \Omega, \\ u &= g \quad \text{on} \quad \partial\Omega \end{aligned} \tag{4.16}$$

into one with homogeneous boundary conditions. To this end, we assume there is a function u_0 which coincides with g on the boundary and for which Lu_0 exists. Then

$$\begin{aligned} Lw &= f_1 \quad \text{in} \quad \Omega, \\ w &= 0 \quad \text{on} \quad \Omega, \end{aligned} \tag{4.17}$$

where $w := a - u_0$ and $f_1 := f - Lu_0$. For simplicity, we usually assume the boundary condition in (4.16) is already homogeneous. We now show that the boundary-value problem (4.17) characterises the solution of the variational problem. A similar analysis was carried out already by L. Euler, and thus the differential equation $Lu = f$ is called the *Euler equation* for the variational problem.

Theorem 4.4.2 (Minimal Property)**.** *Every classical solution of the boundary-value problem*

$$-\sum_{i,k}\partial_i(a_{ik}\partial_k u)+a_0u=f \quad in \quad \Omega,$$
$$u=0 \quad on \quad \partial\Omega$$

is a solution of the variational problem

$$J(v):=\int_\Omega\left[\frac{1}{2}\sum_{i.k}a_{ik}\partial_i v\partial_k v+\frac{1}{2}a_0v^2-fv\right]dx\to min! \tag{4.18}$$

among all functions in $C^2(\Omega)\cap C^0(\bar{\Omega})$ with zero boundary values.

Proof. The proof proceeds with the help of Green's formula

$$\int_\Omega v\partial_i w\ dx=-\int_\Omega w\partial_i v\ dx+\int_{\partial\Omega}vwn_i ds \tag{4.19}$$

Here v and w are assumed to be C^1 functions, and n_i, is the i-th component of the outward-pointing normal n. Inserting $w:=a_{ik}\partial_k u$ in (2.9), we have

$$\int_\Omega v\partial_i(a_{ik}\partial_k u)dx=-\int_\Omega a_{ik}\partial_i v\partial_k u\ dx, \tag{4.20}$$

provided $v=0$ on $\partial\Omega$. Let

$$a(u,v):=\int_\Omega\left[\sum_{i,k}a_{ik}\partial_i u\partial_k v+a_0uv\right]dx, \tag{4.21}$$

$$\ell(v):=\int fvdx. \tag{4.22}$$

Summing (4.20) over i and k gives that for every $v\in C^1(\Omega)\cap C(\overline{\Omega})$ with $v=0$ on $\partial\Omega$

$$a(u,v)-\ell(v)=\int_\Omega v\left[-\sum_{i,k}\partial_i(aik\partial_k u)+a_0u-f\right]dx \tag{4.23}$$

$$=\int_\Omega v[Lu-f]dx=0, \tag{4.24}$$

provided $Lu=f$. This is true if a is a classical solution. Now the characterisation theorem implies the minimal property. □

The same method of proof shows that every solution of the variational problem which lies in the space $C^2(\Omega) \cap C^0(\overline{\Omega})$ is a classical solution of the boundary-value problem.

The above connection was observed by Thomson in 1847, and later by Dirichlet for the Laplace equation. Dirichlet asserted that the boundedness of $1(u)$ from below implies that I attains its minimum for some function u. This argument is now called the Dirichlet principle. However, in 1870 Weierstrass showed that it does not hold in general. In particular, the integral

$$J(u) = \int_0^1 u^2(t)dt$$

4.5 The Ritz-Galerkin Method

There is a simple natural approach to the numerical solution of elliptic boundary-value problems. Instead of minimising the functional J defining the corresponding variational problem over all of $H^m(\Omega)$ or$H_0^m(\Omega)$, respectively, we minimise it over some suitable finite-dimensional subspace [Ritz 1908]. The standard notation for the subspace is S_h. Here h stands for a discretisation parameter, and the notation suggests that the approximate solution will converge to the true solution of the given (continuous) problem as $h \to 0$.

We first consider approximation in general subspaces, and later show how to apply it to a model problem. The solution of the variational problem

$$J(v) = -\frac{1}{2}a(v,v) - \ell(v) \to \min_{S_h}!$$

in the subspace S_h can be computed using the Characterisation Theorem 4.4.1. In particular, u_h, is a solution provided

$$a(u_h, v) = \ell(v) \quad \text{for all } v \in S_h. \tag{4.25}$$

Suppose $\{\psi_1, \psi_2, \ldots, \psi_N\}$ is a basis for S_h. Then (4.25) is equivalent to

$$a(u_h, \psi_i) = \ell(\psi_i), \quad i = 1, 2, \ldots, N.$$

Assuming u_h has the form

$$u_h = \sum_{k=1}^{N} z_k \psi_k, \tag{4.26}$$

we are led to the system of equations

$$\sum_{k=1}^{N} a(\psi_k, \psi_i) zk = \ell(\psi_i), \quad i = 1, 2, \dots N,$$

which we can write in matrix-vector form as

$$Az = b,$$

where $A = a_{i,k} := a(\psi_k, \psi_i)$ and $b_i := \ell(\psi_i)$. Whenever a is an H^m–elliptic bilinear form, the matrix A is positive definite:

$$\begin{aligned} z'Az &= \sum_{i,k} z_i A_{ik} z_k \\ &= a\left(\sum_k z_k \psi_k, \sum_i z_i \psi_i\right) = a(u_h, u_h) \\ &\geq \alpha \|u_h\|_m^2 \end{aligned}$$

and so $z'Az > 0$ for $z \neq 0$. Here we have made use of the bijective maping $\mathbb{R}^N \to S_h$ which is defined by (4.26). Without explicitly referring to this canonical mapping, in the sequel we will identify the function space S_h with $\mathbb{R}^N$. In engineering sciences, and in particular if the problem comes from contiuum mechanics, the matrix A is called the *stiffness matrix* or *system matrix*.

4.6 Methods.

There are several related methods:

Rayleigh-Ritz Method: Here the minimum of J is sought in the space S_h. Instead of the basis-free derivation via (4.25), usually one finds u_h as in (4.26) by solving the equation $(\partial/\partial z_i) J(\sum_k z_k \psi_k) = 0$.

Galerkin Method: The weak equation (4.25) is solved for problems where bilinear form is not necessarily symmetric. If the weak equations arise from variational problem with a positive quadratic form, then often the term Ritz-Galerkin Method is used.

Petrov-Galerkin Method: Here we seek $u_h \in S_h$ with

$$a(u_h, v) = \ell(v) \quad \text{for all } v \in T_h,$$

where the two N-dimensional spaces S_h and T_h need not be the same. The choice of a space of test functions which is different from S_h is particularly useful problems with singularities.

As we saw in previous sections that the boundary conditions determine whether a problem should be formulated in $H^m(\Omega)$ or in $H_0^m(\Omega)$. For the purposes of a unifided notation, in the following we always suppose $V \subset H^m(\Omega)$, and that the bilinear form a is always V-elliptic, i.e.,

$$a(v, v) \geq \|v\|_m^2 \quad \text{and} \quad |a(u, v)| \leq C\|u\|_m\|v\|_m \quad \text{for all } u, v \in V,$$

where $0 < \alpha \leq C$. The norm $\|\cdot\|$ is thus equivalent to the energy norm (2.14), which we use to get our first error bounds. - In addition, let $\ell \in V'$ with $|\ell(v)| < \|\ell\| \cdot \|v\|_m$ for $v \in V$. Here $\|\ell\|$ is the (dual) norm of ℓ.

4.6.1 Example

Let us have the following boundary value problem [1]

$$-\frac{d}{dx}\left(k\frac{du}{dx}\right) = f \text{ in } (0,1) \tag{4.27}$$
$$u(0) = g \text{ and } k\left.\frac{du}{dx}\right|_1 = h.$$

A functional is a special function whose domain is itself a set of functions, and whose range is another set of functions that may be numerical constants.

The idea of the method is to transform the differential equation into an integral problem. This can be obtained by multiplying the differential equation by an arbitrary function ω and then integrating over the domain. Thus for equation (4.8) we have

$$\int_0^1 \left(-\frac{d}{dx}\left(k\frac{du}{dx}\right)\right)\omega dx - \int_0^1 f\omega dx = 0$$

[1] Kikuchi, page 14.

integration by parts gives [2]

$$\begin{aligned} \int_0^1 k\frac{du}{dx}\frac{d\omega}{dx}dx - \left(k\omega\frac{du}{dx}\right)\Big|_0^1 - \int_0^1 f\omega dx &= 0 \\ \int_0^1 \left(k\frac{du}{dx}\frac{dw}{dx} - fw\right)dx - \left(kw\frac{du}{dx}\right)\Big|_a^b &= 0 \end{aligned} \tag{4.28}$$

which is called the "weak form" of the problem.

Let us define

$$\begin{aligned} a(u,\omega) &= \int_0^1 k\frac{du}{dx}\frac{d\omega}{dx}dx \\ \ell(\omega) &= \int_0^1 f\omega dx. \end{aligned}$$

The boundary value problem can then be rewritten as

$$a\left(u,\omega\right) - \ell\left(\omega\right) - \left(k\omega\frac{du}{dx}\right)\Big|_a^b = 0 \tag{4.29}$$

and is called the "abstract form" of the problem.

Let V be the vector space with the following property

$$V = \left\{\omega \in L^2\left(a,b\right) : a\left(\omega,\omega\right) < \infty \wedge \omega\left(0\right) = 0\right\}$$

where L^2 is the space defined by

$$L^2 = \left\{\omega : \Omega \to \mathbb{R} \text{ such that } \int_\Omega \omega^2 dx < \infty\right\}. \tag{4.30}$$

That is square integrable functions. Then, if u is the solution to (4.27), it is characterised by $u \in V$ such that $a(u,\omega) = (f,\omega)\ \forall\, \omega \in V$.

In order to find the variational form of the problem, let us define a functional $F(w)$ based in (4.29) as follows

$$F\left(\omega\right) = a\left(\omega,\omega\right) - \ell\left(\omega\right) - \left(k\omega\frac{dw}{dx}\right)\Big|_a^b. \tag{4.31}$$

[2] $\int_a^b f'g = fg|_a^b - \int_a^b fg'$

It can then be shown that $F(w)$ has a minimum at u, that is

$$F(u) < F(\omega) \quad \forall\, \omega \in \mathrm{V}. \tag{4.32}$$

This is called the variational form of the problem.

We can conclude that we have equivalent relationships among the following three forms:

$$-\frac{d}{dx}\left(k\frac{du}{dx}\right) = f \quad \text{in } (0,1), \quad u(0) = g, \quad \text{and} \quad k\left.\frac{du}{dx}\right|_1 = h \tag{P1}$$

$$\int_0^1 \left(k\frac{du}{dx}\frac{dw}{dx} - fw\right)dx - \left.\left(kw\frac{du}{dx}\right)\right|_a^b = 0 \tag{P2}$$

$$F(u) < F(\omega) \quad \forall\, \omega \in \mathrm{V}. \tag{P3}$$

We should call (P1), (P2), and (P3) the local, weak, and variational forms respectively. In the above, the functional form F is chosen for the Euler equation (4.27). However, for a given boundary value problem, it might be difficult to find the corresponding functional for the variational formulation. Nevertheless, it is not necessary to find $F(\omega)$ to solve the problem and the weak form (P2) can be used instead of (P3), since the form (P2) is easily obtained from the differential form by the procedure shown in this section.

The space V therefore can be seen as ω such that $\omega' \in L^2$ and $\omega(a) = 0$. The next step is to find a set of functions ω that satisfy the problem.

4.7 Discrete Problem (Galerkin Method)

Let us have the following boundary value problem

$$-\frac{d}{dx}\left(k\frac{du}{dx}\right) = f \quad \text{in} \quad (a,b) \quad u(a) = g.$$

Integration by parts take us to the weak form of the problem

$$\int_a^b k\frac{du}{dx}\frac{dw}{dx}dx - \left.kw\frac{du}{dx}\right|_a^b = \int_a^b fw dx$$

with $w \in V$ and V is the space of admissible functions. The problem can be expressed in an abstract way as

To find $u \in U$

$$a\left(u,w\right)=\ell\left(w\right)\quad\forall w\in V$$

for which in the above case

$$\begin{aligned} a\left(u,w\right) &= \int_a^b k\frac{du}{dx}\frac{dw}{dx}dx-kw\frac{du}{dx}\Big|_a^b \\ \ell\left(w\right) &= \int_a^b fw\,dx. \end{aligned}$$

Notice that $a(u,w)$ is a bilinear operator.

The Galerkin method consists of choosing u and w belonging to the same space of functions U.

$$U=\left\{u,\text{admissible function},H^1\right\}$$

if $span\left\{\phi_1,\phi_2,\ldots\right\}$ is a base of the space, we can express

$$u(x)=\sum_i \alpha_i\ \phi_i(x).$$

Notice that the dimension of the base is unknown. The function u can be approximated by a base of finite dimension as

$$u\cong\tilde{u}\left(x\right)=\sum_{i=1}^{n}u_i\ \phi_i(x)$$

where $u_i=u\left(x\right)$ is the value of a function at a point $x_i\in\Omega$.

The Galerkin method consists of choosing u and w to the same space U. The abstract problem can then be expressed as

$$\begin{aligned} a\left(u,w\right) &= \ell\left(w\right) \\ a\left(\sum u_i\phi_i,\sum w_j\phi_j\right) &= \ell\left(\sum w_j\phi_j\right). \end{aligned} \tag{4.33}$$

Because a is a bilinear operator, we have

$$\begin{aligned} a\left(\sum_i u_i\phi_i,\sum_j w_j\phi_j\right) &= \sum_i a\left(u_i\phi_i,\sum_j w_j\phi_j\right) \\ &= \sum_i u_i a\left(\phi_i,\sum_j w_j\phi_j\right) \\ &= \sum_{ij} u_i w_j a\left(\phi_i,\phi_j\right). \end{aligned}$$

In the same way, for $\ell(w)$ we have

$$\begin{aligned} \ell(w) &= \ell\left(\sum_j w_j \phi_j\right) \\ &= \sum_j w_j \ell(\phi_j). \end{aligned}$$

The equation (4.33) can be rewritten as

$$\sum_{ij} u_i w_j a(\phi_i, \phi_j) = \sum_j w_j \ell(\phi_j).$$

The term $a(\phi_i, \phi_j)$ is a real value that can be computed from computing the integrals in terms of the base functions, that is

$$a(\phi_i, \phi_j) = \int_a^b \frac{d\phi_i}{dx} \frac{d\phi_j}{dx}\, dx. \tag{4.34}$$

Then a_{ij} represent a matrix that has the dimension of the space of approximation V. The discrete problem can be expressed in vector form as

$$\langle A\mathbf{u}, \mathbf{w}\rangle = \langle \boldsymbol{\ell}, \mathbf{w}\rangle \tag{4.35}$$

where A is the matrix form by $A = a_{ij} = a(\phi_i, \phi_j)$, $\mathbf{u} = (u_1, \ldots, u_n)$, $\mathbf{w} = (w_1, \ldots, w_n)$ and $\ell_j = \ell(\phi_j)$.

Using the properties of the inner product, equation (4.35) is transformed into

$$\langle A\mathbf{u} - \boldsymbol{\ell}, \mathbf{w}\rangle = 0,$$

and because we have to satisfy this equation for all $w \in V$ then

$$A\mathbf{u} - \boldsymbol{\ell} = 0.$$

Solving for $\mathbf{u}$ we find the discrete approximation to the solution.

4.7.1 Dirichlet boundary conditions

$$u(a) = u_a,\ u(b) = u_b$$

The weak form of the problem

$$\int_a^b \frac{du}{dx}\frac{dv}{dx}\,dx - v\;k\frac{du}{dx}\bigg|_a^b = \int_a^b fv\,dx. \tag{4.36}$$

Select the appropriate space of approximations of the functions u and v as

$$\begin{array}{ll} u \in H^1 \text{ where} & H^1 = \{u \text{ such that } u \text{ is continuum and derivable by parts}\} \\ v \in H_0^1 \text{where} & H_0^1 = \{u \in H^1 \text{and } v(a) = v(b) = 0\}. \end{array}$$

Then the effect of selecting $v \in H_0^1$ is to cancel (without lost of generality) the second term on the left side of (4.36). The spaces H_1 and H_0^1 can be expressed with the same base as

$$\begin{array}{ll} (H^1)^n = & span\{\phi_i,\ i = 1 \ldots n\} \\ (H_0^1)^n = & span\{\phi_i,\ i = 2 \ldots n-1\}. \end{array}$$

That is, the values for ϕ_1 and ϕ_n in $(H_0^1)^n$ are omitted. In the Dirichlet boundary problem the values of the functions at the boundary points a and b are known. Then u can be expressed in terms of H_0^1 in the following way:

$$u(x) \approx u_a\phi_1 + \sum_{i=2}^{n-1} u_i\phi_i + u_b\phi_n.$$

Then the abstract form of the Dirichlet boundary problem in (4.36) can be written as

$$\begin{array}{rcl} a(u,v) & = & \ell(v) \\ a\left(u_a\phi_1 + \sum_{i=2}^{n-1} u_i\phi_i + u_b\phi_n,\ \sum_{j=2}^{n-1} v_j\phi_j\right) & = & \ell\left(\sum_{i=2}^{n-1} v_j\phi_j\right) \end{array}$$

and applying the bi-linearity properties of the a operator we have

$$\begin{array}{rcl} a\left(u_a\phi_1 + u_b\phi_n,\ \sum_{i=2}^{n-1} v_j\phi_j\right) + a\left(\sum_{i=2}^{n-1} u_i\phi_i,\ \sum_{j=2}^{n-1} v_j\phi_j\right) & = & \ell\left(\sum_{i=2}^{n-1} v_j\phi_j\right) \\ \sum_{i=2}^{n-1} v_j a\left(u_a\phi_1 + u_b\phi_n,\ \phi_j\right) + \sum_{i=2}^{n-1} u_i \sum_{j=2}^{n-1} v_j a\left(\phi_i,\ \phi_j\right) & = & \sum_{i=2}^{n-1} v_j \ell\left(\phi_j\right). \end{array}$$

Defining $A^D = a(\phi_i, \phi_j)$ and $\boldsymbol{\ell}^D = \ell(\phi_j)$ for $i, j = 2 \ldots n-1$, the last expression can be simplified as

$$\sum_{i=2}^{n-1} v_j a\left(u_a\phi_1 + u_b\phi_n,\ \phi_j\right) + \left\langle A^D\mathbf{u},\ \mathbf{v}\right\rangle = \left\langle \boldsymbol{\ell}^D,\ \mathbf{v}\right\rangle$$

where $\mathbf{u}$ and $\mathbf{v}$ represent a vector with dimension $(2 \ldots n-1)$. By applying the bi-linear property once again to the first term of the left side of equation, we have:

$$\begin{aligned}
\sum_{i=2}^{n-1} v_j a\left(u_a\phi_1,\ \phi_j\right) + \sum_{i=2}^{n-1} v_j a\left(u_b\phi_n,\ \phi_j\right) + \left\langle A^D\mathbf{u},\ \mathbf{v}\right\rangle &= \left\langle \boldsymbol{\ell}^D,\ \mathbf{v}\right\rangle \\
u_a \sum_{i=2}^{n-1} v_j a\left(\phi_1,\ \phi_j\right) + u_b \sum_{i=2}^{n-1} v_j a\left(\phi_n,\ \phi_j\right) + \left\langle A^D\mathbf{u},\ \mathbf{v}\right\rangle &= \left\langle \boldsymbol{\ell}^D,\ \mathbf{v}\right\rangle .
\end{aligned}$$

Notice that $a\left(\phi_1,\ \phi_j\right)$ with $j = 1+1, \ldots, n-1$ corresponds to the first column of the matrix a_1, but without the boundary rows 1 and n. We denote this column vector of the matrix by a_j^R. Similarly $a\left(\phi_n,\ \phi_j\right) = a_n^R$ is the n–th column of matrix A without the boundary rows 1 and n again. Using this definition, the last equation can be rewritten in a compact form as:

$$\left\langle A^D\mathbf{u},\ \mathbf{v}\right\rangle = \left\langle \boldsymbol{\ell}^D,\ \mathbf{v}\right\rangle - u_a\left\langle a_1^R,\ \mathbf{v}\right\rangle - u_b\left\langle a_n^R,\ \mathbf{v}\right\rangle$$

and applying the properties of the dot product we have

$$\left\langle A^D\mathbf{u},\ \mathbf{v}\right\rangle = \left\langle \boldsymbol{\ell}^D - u_a a_1^R - u_b a_n^R,\ \mathbf{v}\right\rangle . \tag{4.37}$$

Because (4.37) must be valid for all v, then it is true that

$$A^D\mathbf{u} = \boldsymbol{\ell}^D - u_a a_1^R - u_b a_n^R. \tag{4.38}$$

Solving the linear system of equations represented by (4.38) we found vector $\mathbf{u}$, that is the value of $u(x)$ at the nodes $u_i = u(x_i)$.

4.7.2 Pragmatics

How to find A^D, a_1^R and a_n^R ?

First notice that matrix A can be expressed in term of its columns as

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \dots & a_n \end{pmatrix}$$

where A_i represents the i-*th* column of matrix A. In the same way, if we multiply the matrix by a vector that has only the j-*th* position different from zero, the result is the j-*h* column of the matrix times this value

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{bmatrix} 0 \\ u_j \\ \vdots \\ 0 \end{bmatrix} = u_j \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{nj} \end{bmatrix} = u_j \, a_j.$$

Notice that the matrix vector product can be expressed in terms of the column description of a matrix as:

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = u_1 \, a_1 + u_2 \, a_2 + \ldots + u_n \, a_n.$$

This result can be used to compute the value of $u_a \, A_i^*$. Notice that if we multiply A by a vector with zeros in all the entries but the first one and this one is equal to u_a, then we have

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{bmatrix} u_a \\ 0 \\ \vdots \\ 0 \end{bmatrix} = u_a \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{bmatrix} = u_a \, a_1.$$

If we now remove the rows 1 and n from the column vector A_1. We obtain the desired value a_1^*. Matrices A, A^D, and a^* can be graphically seen as shown in figure 4.2.

- Compute A and ℓ

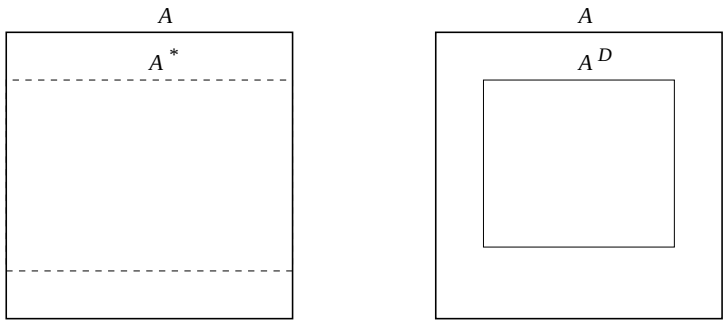


Figure 4.2: Schematic representation of matrices A, A^D and A^R

- Construct the vector

$$\mathbf{g} = \begin{bmatrix} u_a \\ 0 \\ \vdots \\ 0 \\ u_b \end{bmatrix}$$

- Compute $u_a a_1^R + u_b a_n^R$ in two steps as:

  i. Multiply $A\mathbf{g}$.

  ii. Remove boundary rows 1 and n from vector $A\mathbf{g}$.

- Remove boundary rows and columns (1, and n) from A to compute A^D, and the rows (1, and n) from vector ℓ to obtain ℓ^D.

- Solve u from $A^D\mathbf{u} = \boldsymbol{\ell}^D - u_a a_1^R - u_b a_n^R$.

4.8 Computation of the ℓ vector

The ℓ vector id given by

$$\ell_j = \ell(\phi_j) = \int_a^b f\phi_j dx \tag{4.39}$$

In general we can express the function f in terms of the base of the space of approximation so

$$f = \sum_s f_s \phi_s.$$

Replacing the value in the equation (4.39) we get

$$\ell_j = \int_a^b \sum_s f_s \phi_s \phi_j = \sum_s f_s \int_a^b \phi_s \phi_j$$

4.9 von Newman Boundary Conditions

Let us suppose now that the von Newman boundary conditions are known at extreme points a and b,

$$k \frac{du}{dx}\bigg|_a = q_a \qquad k \frac{du}{dx}\bigg|_b = q_b.$$

Then, the second term of the weak form (equation 4.28) can not be cancelled as in the Dirichlet boundary problem. This term can be included in the right-hand side of the equation and then the linear operator $\ell(w)$ can be redefined as

$$\begin{aligned}
\ell(w) &= \int_a^b f w dx - kw \frac{du}{dx}\bigg|_a^b \\
&= \int_a^b \sum_{sj} f_s \phi_s w_j \phi_j - k \sum_j w_j \phi_j \frac{du}{dx}\bigg|_a^b \\
&= \sum_j \int_a^b w_j \left(\sum_s f_s \phi_s \phi_j \right) - k \sum_j w_j \phi_j \frac{du}{dx}\bigg|_a^b \\
&= \sum_j w_j \left[\int_a^b \left(\sum_s f_s \phi_s \phi_j \right) - k \left(\phi_j \frac{du}{dx}\bigg|_a^b \right) \right]
\end{aligned}$$

where

$$\left(\phi_j \frac{du}{dx}\bigg|_a^b \right) = k \left. \frac{du}{dx} \right|_b - k \left. \frac{du}{dx} \right|_a$$

because

$$\begin{array}{rcccccl} \phi_j(a) & = & 1 & \text{for} & j=1 & \text{and } \phi_j(a)=0 & \forall\, j \neq 1 \\ \phi_j(b) & = & 1 & \text{for} & j=n & \text{and } \phi_j(b)=0 & \forall\, j \neq n \end{array}$$

then

$$\ell_j = \begin{bmatrix} \vdots \\ \vdots \\ \sum_s f_s \int_a^b \phi_s \phi_j \\ \vdots \\ \vdots \end{bmatrix} - \begin{bmatrix} k \left.\dfrac{du}{dx}\right|_a \\ 0 \\ \vdots \\ 0 \\ k \left.\dfrac{du}{dx}\right|_b \end{bmatrix}. \tag{4.40}$$

4.10 Example

Construct the matrix of the system corresponding to the one-dimensional conduction problem defined over an interval $[a, b]$ using second order Lagrange polynomials. See figure 4.3.

Solution

Let us consider a one-dimensional domain defined by the interval $[a, b]$ as in figure 4.3. The interval is partitioned in four elements of equal size. This is not always the case but it will simplify the computations in this example. We are interested to find the temperature at each of the internal nodes (1..9).

A second order Lagrange element consists of three base functions as can be observed in figure 4.4. Each of the functions is in the form

$$\phi_i = a_i x^2 + b_i x + c_i.$$

An element k will have nodes numerated $2k-1$, $2k$, and $2k+1$. If ne is the number of elements, then the total number of nodes will be $2(ne)+1$.

For an element whose nodes are located at r, $r + h/2$, and $r + h$ the base funcions are given by

$$\begin{aligned} \phi_1 &= 2/h^2 x^2 - (4r+3h)/h^2 x + 1/h^2(2r^2+3rh+h^2) \\ \phi_2 &= -4/h^2 x^2 + 4(h+2r)/h^2 x - 4r(r+h)/h^2 \\ \phi_3 &= 2/h^2 x^2 - (4r+h)/h^2 x + r(h+2r)/h^2 \end{aligned}$$

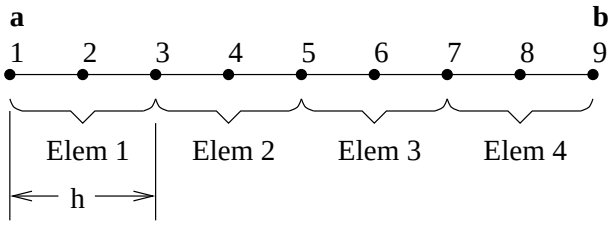


Figure 4.3: Finite element subdivision of a one-dimensional domain. This subdivision corresponds to a second order Lagrange element. For this example the elements are defined to have constant size h and all the nodes are separated by a constant distance $h/2$.

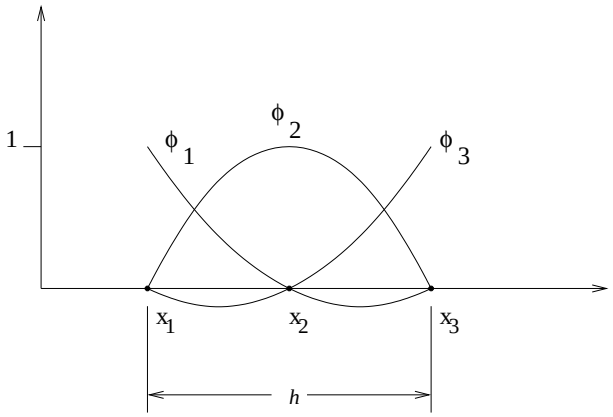


Figure 4.4: Base functions ϕ_1, ϕ_2, ϕ_3 for a one-dimensional second order Lagrange element

Here the base functions ϕ_1 ϕ_2, and ϕ_3 were localy numered when in fact they correspond to the global indices $2k-1$, $2k$, and $2k+1$. The local system matrix for this element can be calculated by solving the integrals

$$a_{\mathrm{local}_{ij}} = \int_r^{r+h} \frac{d\phi_i}{dx}\frac{d\phi_j}{dx}$$

That gives as a result

$$A_{\text{local}} = \begin{bmatrix} \frac{7}{3h} & -\frac{8}{3h} & \frac{1}{3h} \\ -\frac{8}{3h} & \frac{16}{3h} & -\frac{8}{3h} \\ \frac{1}{3h} & -\frac{8}{3h} & \frac{7}{3h} \end{bmatrix}$$

which defines completely the integrals for one general element.

Algorithm 3 Global matrix creation for example

```
ne is the number of elements
for k = 1 to ne do
  g(1) := 2k − 1
  g(2) := 2k
  g(3) := 2k + 1
  for i = 1 to 3 do
    for j = 1 to 3 do
      A[g(i), g(j)] := A[g(i), g(j)] + A_local[i][j]
    end for
  end for
end for
```

However from (4.34) we have that the global system matrix is defined as an integral over the whole domain. This integral can be decomposed as the sum of integrals over the total number of elements

$$a(\phi_i, \phi_j) = \int_a^b \frac{d\phi_i}{dx}\frac{d\phi_j}{dx}\,dx = \sum_{k_e}^{ne} \left(\int_{k_e} \frac{d\phi_i}{dx}\frac{d\phi_j}{dx} \right).$$

One could be tempted to transcript the last equation into an algorithm in order to compute the global matrix. However, this procedure can be very inefficient as most of the terms of the matrix will be equal to zero. This can be concluded by observing a function ϕ_i in the figure 4.3. In general, a function ϕ_i is only different to zero in the range from x_{i-2} to x_{i+2}. Therefore, the product $\frac{d\phi_i}{dx}\frac{d\phi_j}{dx}$ will be different to zero only if $|i-j| < 4$ for the case of 2–D Lagrange polynomials.

A better aproach to construct the global system matrix is presented in Algorithm 3.

Chapter 5

Two-dimensional problems

5.1 Preliminary mathematics

5.1.1 The Divergence (Gauß) theorem

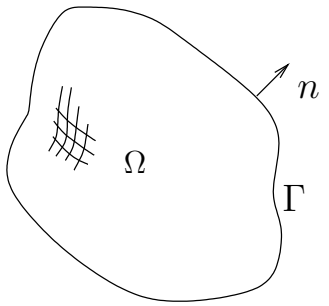


Figure 5.1: Definition of the domain and boundary

This theorem relates the volume integral of a vector function over a volume with a surface integral of the same function, over the surface delimiting its volume. Let $\boldsymbol{\omega}$ be a vector function defined over a domain Ω and let Γ be its surrounding surface. (Notice that this definition can be applied for two and three dimensions)

$$\begin{aligned} \boldsymbol{\omega} \quad &: \quad \mathbb{R}^n \rightarrow \mathbb{R}^n \\ &\quad \Omega \rightarrow \mathbb{R}^n, \quad \Gamma = \text{boundary}(\Omega). \end{aligned}$$

Then the divergence theorem states that

$$\int_{\Gamma} \langle \boldsymbol{\omega}, \mathbf{n} \rangle \, d\Gamma = \int_{\Omega} div\,(\boldsymbol{\omega}) \, d\Omega. \tag{5.1}$$

5.1.2 Green's equation

Let r, u, v be scalar functions defined $\mathbb{R}^n \to \mathbb{R}$ and $\boldsymbol{\omega}, \boldsymbol{z}$ vector functions $\mathbb{R}^n \to \mathbb{R}^n$. It can be shown that the divergence of the product of $\boldsymbol{\omega}$ times $\boldsymbol{z}$ is equal to

$$div\,(v\boldsymbol{z}) = v\; div\,(\boldsymbol{z}) + \langle \boldsymbol{z}, \nabla v \rangle\,. \tag{5.2}$$

Applying the divergence theorem (equation 5.1) with $\boldsymbol{\omega} = v\boldsymbol{z}$ and using this last result we obtain

$$\begin{aligned} \int_{\Gamma} v\,\langle \boldsymbol{z}, \mathbf{n} \rangle \, d\Gamma &= \int_{\Omega} div\,(v\boldsymbol{z})\, d\Omega \\ &= \int_{\Omega} \left(v div\,(\boldsymbol{z}) + \langle \boldsymbol{z}, \nabla v \rangle\right) d\Omega. \end{aligned}$$

If we choose $\boldsymbol{z} = \nabla u$, for some u then

$$\int_{\Gamma} v\,\langle \nabla u, \mathbf{n} \rangle \; d\Gamma = \int_{\Omega} \left(v div\,(\nabla u) + \langle \nabla u, \nabla v \rangle\right) d\Omega,$$

with $div(\nabla u) = \triangle u$, the Laplacian of u, we obtain,

$$\int_{\Gamma} v\,\langle \nabla u, \mathbf{n} \rangle \; d\Gamma = \int_{\Omega} v \triangle u \, d\Omega + \int_{\Omega} \langle \nabla u, \nabla v \rangle \; d\Omega. \tag{5.3}$$

Equation (5.3) is known as Green's equation.

5.2 Poisson's equation

Let u be a scalar field defined over a domain $\Omega \in \mathbb{R}^2$ with boundary $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$ as in figure 5.2. Then Poisson's equation is defined as

$$-\nabla^2 u = f(x). \tag{5.4}$$

In the heat transfer case, Poisson's equation governs the steady state temperature distribution for which $f(x)$ represents the internal heat source.

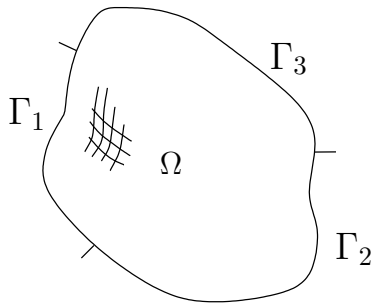


Figure 5.2: Definition of the domain and boundary

To complete the boundary problem, the following boundary conditions are defined:

Dirichlet:	$u\|_{\Gamma_1} = g$	temperature at boundary Γ_1.
Newman:	$\langle \nabla u, \mathbf{n} \rangle\|_{\Gamma_2} = \left(\frac{\partial u}{\partial n}\right)_{\Gamma_2} = q$	heat flow at boundary Γ_2
Fourier:	$(\alpha \langle \nabla u, n \rangle + \beta \mathrm{u})_{\Gamma_3} = \gamma$	convection at boundary Γ_3.

where $\Gamma_1 \cap \Gamma_2 \cap \Gamma_3 = \emptyset$

5.2.1 Weak form of the problem

To obtain the weak form of the problem we multiply (5.4) for a test function v and integrate over the whole domain

$$-\int_\Omega \nabla^2 u \, v \, d\Omega = \int_\Omega f v \, d\Omega.$$

Then we apply Green's equation (5.3) to reduce the order of the equation and obtain

$$\int_\Omega \nabla u \nabla v \, d\Omega - \int_\Gamma v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma = \int_\Omega f v \, d\Omega \tag{5.5}$$

which is the weak form of equation (5.4).

Before going further, we need to identify the function space of approximation for which (5.5) has a solution. Let L^2 be the space of functions which are square integable, that is

$$L^2(\Omega) = \left\{ v : \Omega \to \mathbb{R}, \;\middle|\; \int_\Omega |v|^2 \, d\Omega < \infty \right\},$$

and let H^1 be the space of functions whose first partial derivates are in L^2,

$$H^1(\Omega) = \left\{ u : \Omega \to \mathbb{R} \;\middle|\; u \in L^2 \text{ and } \frac{\partial u}{\partial x_i} \in L^2(\Omega) \right\}.$$

Then to guarantee a solution, functions u and v must belong to a space U which is a subset of H^1. Additionaly it is necessary to define one more space, the space of functions in H^1 that are equal to zero in the boundary of the domain,

$$H_0^1(\Omega) = \left\{ v : \Omega \to \mathbb{R} \;\middle|\; v \in H_1 \text{ and } v = 0 \text{ on the boundary} \right\}.$$

Figure 5.3 shows a two-dimensional example of a function v in H_0^1

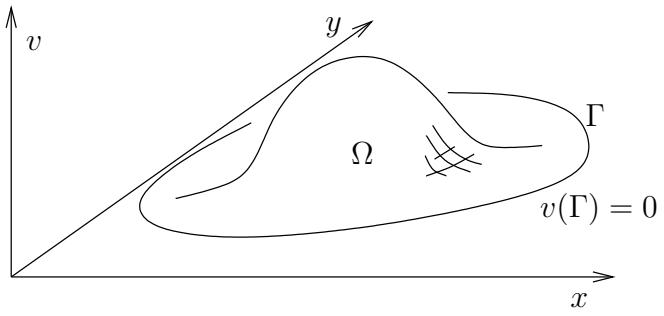


Figure 5.3: Schematic view of a function $v : \Omega \to \mathbb{R}$, $v \in H_0^1$, and $\Omega \in \mathbb{R}^2$

5.2.2 Dirichlet homogeneous boundary problem

Let g be the known values of function u at boundary Γ of the domain

$$u|_\Gamma = 0.$$

If $v \in H_0^1(\Omega)$ the second term of (5.5) is cancelled and the weak form of the problem is simplified as

$$\int_\Omega \nabla u \nabla v \, d\Omega = \int_\Omega f v \, d\Omega \tag{5.6}$$

5.2.3 Newman homogeneous

Newman boundary conditions are flow conditions over the boundary of the domain. If they are equal to zero then it is called homogeneous

$$\left.\frac{\partial u}{\partial n}\right|_{\Gamma} = 0.$$

Assuming $v \in H^1$ the second term of (5.5) is cancelled by the Newman homogeneous condition. The weak form of the problem is transformed into

$$\int_{\Omega} \nabla u \nabla v \, d\Omega = \int_{\Omega} f v \, d\Omega. \tag{5.7}$$

5.2.4 Discrete problem

Let

$$H^1 = span\left\{\phi_1, \phi_2, \ldots, \phi_n\right\}$$

then

$$u\left(x\right) = \sum_i u_i \phi_i(x) \quad \text{where} \quad u_i = u\left(x_i\right), \quad x \in \mathbb{R}^2.$$

In the same way

$$v\left(x\right) = \sum_i v_j \phi_j, (x)$$

defining:

$$a\left(u, v\right) = \int_{\Omega} \nabla u \nabla v \, d\Omega.$$

It can be shown that $a\left(u, v\right)$ is a bilinear operator, therefore

$$\begin{aligned} &= a\left(\sum_i u_i \phi_i\left(x\right), \sum_j v_j \phi_j\left(x\right)\right) \\ &= \sum_{i,j} u_i v_j a\left(\phi_i\left(x\right), \phi_j\left(x\right)\right) \end{aligned}$$

and

$$\ell\left(v\right) = \int_{\Omega} f v \, d\Omega$$

which is a lineal operator and therefore

$$\ell(v) = \sum_j v_j \ell(\phi_j).$$

Moreover, (5.6) is transformed into

$$\sum_{i,j} u_i v_j a(\phi_i, \phi_j) = \sum_j v_j \ell(\phi_j). \tag{5.8}$$

If $a(\phi_i, \phi_j) = a_{ij}$ and $\ell(\phi_j) = \ell_j$, then (5.8) can be rewritten in terms of matrices and inner products as

$$\langle A\mathbf{u}, \mathbf{v} \rangle = \langle \ell, \mathbf{v} \rangle, \tag{5.9}$$

where

$$\begin{aligned} A &= a_{ij} \\ \mathbf{u} &= (u_1, u_2, \ldots, u_n)^T \\ \mathbf{v} &= (v_1, v_2, \ldots, v_n)^T \\ \ell &= (\ell_1, \ell_2, \ldots, \ell_n)^T. \end{aligned}$$

By the properties of the inner product

$$\begin{aligned} \langle A\mathbf{u}, \mathbf{v} \rangle &= \langle \ell, \mathbf{v} \rangle \\ \langle A\mathbf{u}, \mathbf{v} \rangle - \langle \ell, \mathbf{v} \rangle &= 0 \\ \langle A\mathbf{u} - \ell, \mathbf{v} \rangle &= 0 \\ &. \end{aligned}$$

Because this is valid for all $v \in H^1$ then

$$\begin{aligned} A\mathbf{u} - \ell &= 0 \\ A\mathbf{u} &= \ell. \end{aligned}$$

Solving the system we found u that satisfies $-\nabla^2 u = f$ at node points. The matrix A is usually referred as the stifness matrix or as the system matrix. This is because of the syimilarity with the elasticity problem were the method was first developed. It can be calculated by

$$A = a_{ij} = \int_\Omega \nabla\phi_i \nabla\phi_j d\Omega. \tag{5.10}$$

Dividing the domain into a set of "finite elements", the stiffness matrix over the domain can be calculated as the sum of interals over the total number of elements, that is,

$$a_{ij} = \sum_e \int_{\Omega_e} \nabla\phi_i \nabla\phi_j d\Omega_e$$

5.2.5 Computation of $\int_\triangle \nabla\phi_i \nabla\phi_j d\Omega$

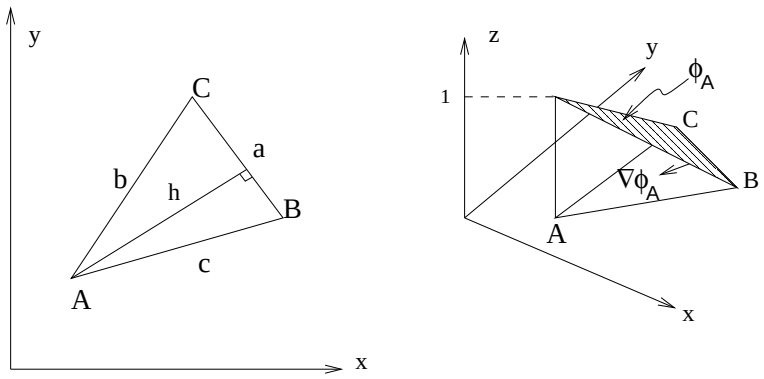


Figure 5.4: Geometric computation of Lagrange P1 integrals

This section discusses a method for the computation of the integrals that constitute matrix terms $a_{i,j}$ of the discrete system. This indeed depends on the space of approximation selected for the solution. For the present example, the space of approximation consists of first order Lagrange polynomials (P1 Elements). In order to evaluate the integrals, we first have to evaluate the gradient of these functions. So let ABC define a triangle in the domain as shown in figure 5.4. Vertex A, B, and C are ordered in the counterclockwise direction. The length of the sides of the triangles are defined in the following way:

$$a = \left\|\overrightarrow{\mathrm{BC}}\right\|, \quad b = \left\|\overrightarrow{\mathrm{CA}}\right\|, \quad c = \left\|\overrightarrow{\mathrm{AB}}\right\|.$$

Each triangle has three possible functions: $\phi_A(x), \phi_B(x), \phi_C(x)$ which are first order functions defining a plane. The gradient of each one of these functions can be easily calculated by visualising the plane and computing its slope and direction. We will compute the gradient for function $\phi_A(x)$

and then extend the result for ϕ_B and ϕ_C. If $\phi_A(x)$ is a lineal function such that $\phi_A(A) = 1$ and $\phi_A(B) = \phi_A(C) = 0$. See figure 5.4. Then $\nabla\phi_A$ is orthogonal to the BC side. (BC is the iso-level line and the gradient is always normal to the iso-level lines.) As ϕ_A goes from zero to one from the BC side to the A vertex and h is the height of the BC side of the triangle, then the slope is equal to $1/h$, that is:

$$|\nabla\phi_A| = \frac{1}{h}.$$

To obtain the direction of the gradient, we proceed in the following way. If $\hat{k}$ is a unit vector in the direction of the z axis, then a vector in the xy plane perpendicular to the BC side is given by the cross product of $\hat{k}$ and $\overrightarrow{\mathrm{BC}}$, that is:

$$\frac{\hat{k} \times \overrightarrow{\mathrm{BC}}}{\left\|\overrightarrow{\mathrm{BC}}\right\|} = \frac{\hat{k} \times \overrightarrow{\mathrm{BC}}}{a}.$$

Because the area of the triangle is given by $Area = ah/2$, then the gradient vector can be expressed as

$$\nabla\phi_A = \frac{1}{h}\frac{\hat{k} \times \overrightarrow{BC}}{a} = \frac{\hat{k} \times \overrightarrow{BC}}{2 \cdot Area}.$$

Notice that the gradient of the base function ϕ_A is equal to the cross-product between $\hat{k}$ and the vector defined by the opposite vertex (in the counter clock direction) and divided by 2 times the area of the triangle. This result can be extended to base functions ϕ_b and ϕ_C

$$\begin{aligned}\nabla\phi_B &= \frac{\hat{k} \times \overrightarrow{CA}}{2 \cdot Area} \\ \nabla\phi_C &= \frac{\hat{k} \times \overrightarrow{AB}}{2 \cdot Area}.\end{aligned}$$

Next we need to compute the dot product of each of the two gradients in the triangle. For example:

$$\begin{aligned}\nabla\phi_A \cdot \nabla\phi_B &= \frac{\hat{k} \times \overrightarrow{BC}}{2 \cdot Area} \cdot \frac{\hat{k} \times \overrightarrow{CA}}{2 \cdot Area} \\ &= \frac{\overrightarrow{BC} \cdot \overrightarrow{CA}}{4 \cdot Area^2}.\end{aligned}$$

And the integral of this product over the triangle

$$\begin{aligned}\int_\Omega \nabla\phi_A \cdot \nabla\phi_B d\Omega &= \int_\Omega \frac{\overrightarrow{BC}\cdot\overrightarrow{CA}}{4\cdot A^2} d\Omega \\ &= \frac{\overrightarrow{BC}\cdot\overrightarrow{CA}}{4\cdot A^2}\int_\Omega d\Omega \\ &= \frac{\overrightarrow{BC}\cdot\overrightarrow{CA}}{4\cdot A}.\end{aligned}$$

In a similar way it can be found that

$$\int_\Omega \nabla\phi_A \cdot \nabla\phi_A d\Omega = \frac{a^2}{4\cdot A}.$$

5.2.6 Non-homogeneous Dirichlet boundary problem

In this case, the general integral form of the problem is given by

$$\int \nabla u \cdot \nabla v - \int_\Gamma v\ \nabla u \cdot n = \int fv$$

with boundary conditions

$$u|_\Gamma = g,$$

where g is a known value. We select $u \in U$ and $v \in V$ defined as before,

$$\begin{aligned} U &\to H^1[\Omega] \text{ continuum and derivable by parts} \\ V &\to H_0^1[\Omega] \text{ equal to zero in the boundary}\end{aligned}$$

in such a way that the second term of the integral equation, $\int_\Gamma v\ \nabla u \cdot n$, is equal to zero. The abstract form of the problem becomes,

$$a(u,v) = \ell(v).$$

The solution space H^1 is selected as a finite space with base $H^1 = span\{\phi_1, \phi_2, \ldots, \phi_n\}$. A function ϕ_i is referred as base (or shape) function. The main characteristic of these functions is that they only take values for a small domain around the point where they are defined and zero in the rest. That is, ϕ_i is defined around the point (x_i) and zero somewhere else. As such, there are functions that corresponds to boundary points ϕ_j, such

as $x_j \in \Gamma$ and functions that correspond to inside points ϕ_i such as $x_i \in I$. This way we can classify the nodes into two sets Γ and I, with $\Gamma \cap I = \varnothing$. If a function $v \in H_0^1$, then $v_i = v(x_i) = 0$ for all $i \in \Gamma$ so H_0^1 can be expressed in terms of the same base ϕ_i if we only use the functions with inside index. Given

$$v \in \left(H_0^1\right)^n \qquad \text{and} \qquad H^1 = span\left\{\phi_i\right\}_1^n,$$

then v can be expressed in terms of the H_0^1 base if we only take the functions ϕ_i that do not belong to boundary points ($x_i \notin \Gamma$)

$$v = \sum_1^n v_i \phi_i(x) = \sum_{i \in I} v_i \phi_i(x)$$

where $i \in I$ means $x_i \in I$. Notice that in general, the node index $j = i \ldots n$ can be classified as inside ($i \in I$) if the point $x_j \in \Omega - \Gamma$ or in the boundary $j \in \Gamma$, if the point $x_j \in \Gamma$.

In order to express a function u in terms of the base, we have to take into account the known values of the function at the boundary Γ

$$u|_\Gamma = g$$

where g is the Dirichlet boundary condition expressed in terms of node boundaries

$$u(x_i) = g_i, \qquad \text{for all } x_i \in \Gamma.$$

Using these values, the function u can be expressed in terms of the same base of functions by separating the unknown values $\bar{u}$ from the known values g at the boundary

$$u = \sum_{i \in I} \bar{u}_i w_i + \sum_{i \in \Gamma} g_i w_i.$$

Replacing the values of functions u and v for their interpolated approxima-

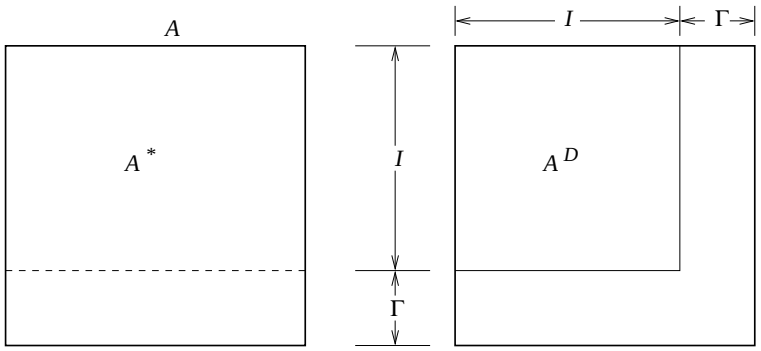


Figure 5.5: Graphic representation of a matrix for a two or three dimensional problem. The rows and columns with indexes corresponding to inside points (I) are organised at the beginning of the matrix while the rows and columns with indexes belonging to boundary points Γ are located at the end.

tion into the abstract form, we have

$$\begin{aligned} a(u,v) &= a\left(\sum_I \bar{u}_i\phi_i + \sum_\Gamma g_i\phi_i, \sum_I v_j\phi_j\right) \\ &= a\left(\sum_I \bar{u}_i\phi_i, \sum_I v_j\phi_j\right) + a\left(\sum_\Gamma g_i\phi_i, \sum_I v_j\phi_j\right) \\ &= \sum_{i,j\in I} \bar{u}_i v_j a(\phi_i,\phi_j) + \sum_{i\in\Gamma, j\in I} g_i v_j a(\phi_i,\phi_j) \\ &= \sum_{i,j\in I} \bar{u}_i v_j a_{ij} + \sum_{i\in\Gamma, j\in I} g_i v_j a_{ij}. \end{aligned} \tag{5.11}$$

The indices for the first term of the right-hand side correspond to inside points only and therefore can be written in compact notation as

$$\sum_{i,j\in I} \bar{u}_i v_j a_{ij} = \left\langle A^D \bar{u}, v\right\rangle$$

where A^D is the matrix that contains only the indices of the inside points. Figure 5.5 shows a schematic representation of this matrix.

The second term of (5.11) can be decomposed as

$$\sum_{i\in\Gamma, j\in I} g_i v_j a_{ij} = \sum_{j\in I} v_j d_j \qquad \text{with} \quad d_j = \sum_{i\in\Gamma} a_{ij} g_i.$$

Remember that g_j are the values of u at the boundary points, that is $g_j = u(x_j)$. Then, if we define the vector g as

$$g_i = \begin{cases} 0 & \text{if} \quad i \in \mathrm{I} \\ u(x_i) & \text{if} \quad i \in \Gamma \end{cases} \tag{5.12}$$

then we can extend the limits of the sum until $I+\Gamma$ and apply the symmetry property of matrix a_{ij} to obtain

$$d_j = \sum_{i \in I+\Gamma} a_{ij} g_i = \sum_{i \in I+\Gamma} a_{ji} g_i \qquad \text{for all } j \in I,$$

which is a matrix vector multiplication operation between matrix A times vector g. As $j \in I$ from this multiplication we supress rows corresponding to boundary points, $j \in \Gamma$.

$$d = (Ag)^D,$$

See figure 5.6. Then (5.11) is transformed into

$$a(u, v) = \left\langle A^D \bar{u}, v \right\rangle + \langle d, v \rangle .$$

Using this result the abstract form of the Dirichlet problem is transformed into

$$\left\langle A^D \bar{u}, v \right\rangle = \langle \ell, v \rangle - \langle d, v \rangle .$$

And by the properties of the inner product we have,

$$\left\langle A^D u - \ell + d, v \right\rangle = 0.$$

Because this result must be valid for all v in V we have

$$A^D \bar{u} = \ell - d \tag{5.13}$$

which is the discrete form of the Dirichlet problem of the Poisson equation (5.4).

5.2.7 Non-homogeneous von Newman boundary problems

Let $u(x)$ defined over $\Omega \in \mathbb{R}^n$ satisfy the Poisson equation (5.4)

$$-\nabla^2 u = f(x),$$

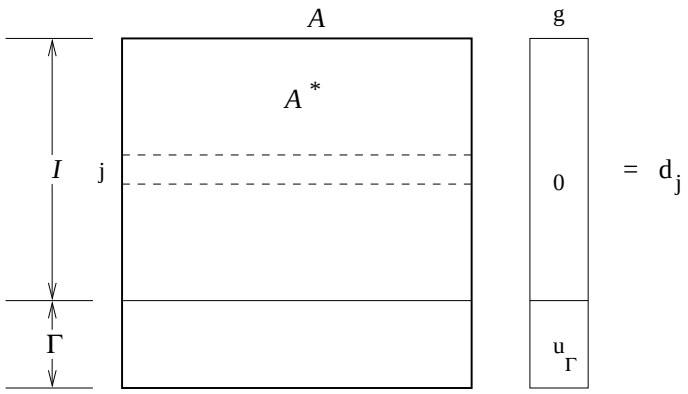


Figure 5.6: Graphic representation of the computation of vector d as the multiplication of the jth row of A times vector g. Notice that $j \in I$ only.

with boundary conditions:

Dirichlet:	$u\|_{\Gamma_1} = g$	function defined at boundary Γ_1.
Newman:	$\langle \nabla u, \mathbf{n} \rangle\|_{\Gamma_2} = \left(\frac{\partial u}{\partial n}\right)_{\Gamma_2} = q$	gradient defined at boundary Γ_2

If we subdivide the boundary Γ in $\Gamma_1 + \Gamma_2$, the weak form (equation 5.5) can be written as

$$\int_\Omega \nabla u \nabla v \, d\Omega - \int_{\Gamma_1} v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma - \int_{\Gamma_2} v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma = \int_\Omega f v \, d\Omega$$

As a difference of the dirichlet boundary problem, the interal over the boundary is known over a section of it, Γ_2.

To find the solution in a finite space we select the solution space for u as $u \in H^1$. For the trial function v we select a variation of the space H_0^1 where a function is equal to zero only at the boundary Γ_2. In this way we can cancel the integral over the section Γ_1 of the boundary and left the integral over the section Γ_2 which is given by the von Newman boundary condition. Replacing the value of the boundary condition into the weak form we have

$$\int_\Omega \nabla u \nabla v \, d\Omega - \int_{\Gamma_2} vq \; d\Gamma = \int_\Omega f v \, d\Omega$$

The only difference at this point is in the second term of the left hand side of the equation. Besides we will consider an index as $i \in \Gamma_1$ if $x_i \in \Gamma_1$

and $i \in I$ if $x_i \notin \Gamma_1$ then the functions u and v can be expresed in terms of the base as

$$u = \sum_{i \in I} u_i \phi_i + \sum_{i \in \Gamma_1} g_i \phi_i$$
$$v = \sum_{j \in I} v_j \phi_j$$

then

$$\begin{aligned} \int_{\Gamma_2} q(x)\ v\ d\Gamma &= \int_{\Gamma_2} q(x) \sum_{j \in I} v_j \phi_j\ d\Gamma \\ &= \sum_{j \in I} v_j \int_{\Gamma_2} q(x) \phi_j\ d\Gamma \\ &= \sum_{j \in I} v_j \hat{q}_j \\ &= \langle \boldsymbol{v}, \hat{\boldsymbol{q}}_j \rangle \end{aligned}$$

where $\hat{q}_j$ was defined as

$$\hat{q}_j = \int_{\Gamma_2} q(x) \phi_j\ d\Gamma$$

Notice that this integral is only defined over the boundary Γ_2, this means that $\hat{q}_j \neq 0$ only for the indices j where $x_j \in \Gamma_2$ expresing $q(x)$ in terms of the interpolation base fuctions, $q(x) = (\sum_i q_i \phi_i)$, we have

$$\hat{q}_j = \int_{\Gamma_2} \left(\sum_i q_i \phi_i \right) \phi_j\ d\Gamma,$$

that is

$$\hat{q}_j = \sum_i q_i \int_{\Gamma_2} \phi_i \phi_j\ d\Gamma.$$

5.2.8 Example

Solve the equation $-\nabla^2 u = -10$ for the domain represented in figure 5.7, for the following set of boundary conditions:

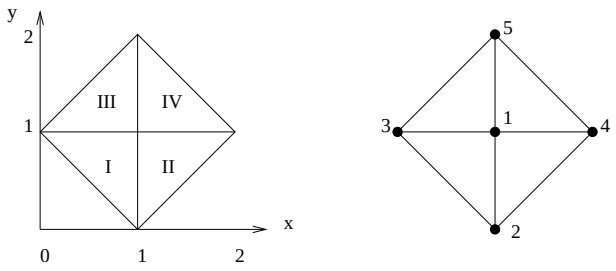


Figure 5.7: Four elements domain for a simple example

a. **Dirichlet Only**
Dirichlet at the nodes $u(x_2) = 2$, $u(x_3) = 3$, $u(x_4) = 11$, $u(x_5) = 14$

b. **Dirichlet + von Newman**
Dirchlet at the nodes $u(x_4) = 2$, $u(x_5) = 2$ and von Newman $\dfrac{\partial u}{\partial \hat{n}} = ...$ along the segment formed by nodes 2 and 3.

Before proceed notice that this problem has an analytical solution equal to $u = 2x^2 + 3y^2$ this can be proved by computing the Laplacian of u

$$\nabla^2 u = \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = 10 \tag{5.14}$$

Notice that for each element the matrices are constants

$$\int_e \langle \nabla w_i, \nabla w_j \rangle = \begin{bmatrix} 1 & -1/2 & -1/2 \\ -1/2 & 1/2 & 0 \\ 0 & -1/2 & 1/2 \end{bmatrix} \tag{5.15}$$

$$\int_e \langle w_i, w_j \rangle = \begin{bmatrix} 1/12 & 1/24 & 1/24 \\ 1/24 & 1/12 & 1/24 \\ 1/24 & 1/24 & 1/12 \end{bmatrix} \tag{5.16}$$

Solution a. Dirichlet Only Notice that for the first the nodes that belong to Γ_1 are $2, 3, 4, 5$ and only the node number one is left after removing the dirichlet nodes. We end up with a one dimensional problem with $A = a_{11}$,

$$d = (Ag)^D \rightarrow d = d_1 = \sum_j a_{ij} g_j$$

with g formed by the dirichlet boundary conditions as $g = [0, 2, 3, 11, 14]^T$

To compute A we need to calculet the first row of A, a_1^T. To compute A, the integral over the domain is splited into integrals over the elements

$$a_{ij} = \int_\Omega \langle \nabla w_i, \nabla w_j \rangle = \sum_e \int_e \langle \nabla w_i, \nabla w_j \rangle$$

For example

$$\begin{aligned} a_{11} &= \int_I \langle \nabla w_1, \nabla w_1 \rangle + \int_{II} \langle \nabla w_1, \nabla w_1 \rangle + \int_{III} \langle \nabla w_1, \nabla w_1 \rangle + \int_{IV} \langle \nabla w_1, \nabla w_1 \rangle \\ &= 1 + 1 + 1 + 1 = 4 \end{aligned}$$

$$a_{12} = \int_I \langle \nabla w_1, \nabla w_2 \rangle + \int_{II} \langle \nabla w_1, \nabla w_2 \rangle = -1$$

$$a_{13} = \int_I \langle \nabla w_1, \nabla w_3 \rangle + \int_{III} \langle \nabla w_1, \nabla w_3 \rangle = -1$$

De igualmente calculamos para a_{14} y a_{15} y completamos

$$a_1^T = [4, -1, -1, -1, -1]$$

de esta forma $d_1 = (Ag)^D = \langle a_1^T, g \rangle = -30$

Vector ℓ consiste solo de una componente

$$\ell_1 = \int_\Omega f(x)\, w_1 = \int_\Omega \left(\sum_{j=1}^{5} f_j w_j\, w_1 \right)$$

and dividing the integral over the domain as the sum of integrals over the elements an with $f_i = -10$ constant for all i we have

$$\ell_1 = (-10) \sum_e \left(\sum_{j=1}^{5} \int_e w_j\, w_1 \right)$$

As all the elements in this example are equal then the inner sum is the sum over the first row of (5.16)

$$\ell_1 = (-10)(4)(1/12 + 1/24 + 1/24) = -20/3$$

$$A^D u = \ell - d \tag{5.17}$$
$$4u = -20/3 - 30 \tag{5.18}$$

form where $u = 5.83$. As the analitical solution for u(1,1) is equal to 6 then the error is 2.8%.

Solution b. Dirichlet + von Newman

5.2.9 Fourier boundary conditions

Find $u(x) : \Omega\mathbb{R}^n \rightarrow \mathbb{R}$ that satisfies

$$-\nabla^2 u = f(x), \tag{5.19}$$

with boundary conditions:

Dirichlet: $$u(x)\Big|_{\Gamma_1} = g(x) \qquad \text{for all } x \in \Gamma_1 \tag{5.20}$$

von Newman: $$\langle \nabla u(x), \mathbf{n} \rangle|_{\Gamma_2} = \left(\frac{\partial u}{\partial n}\right)_{\Gamma_2} = q \tag{5.21}$$

Fourier: $$\left(\alpha u + \beta \frac{\partial u}{\partial \hat{n}}\right)\Big|_{\Gamma_3} = \gamma \tag{5.22}$$

In a heat transfer context Dirichlet boundary conditions represent the known temperatures at a given surface. Von Newman boundary conditions the heat flux and Fourier boundary conditions repesent a convection surfaces and it is usually writen in terms of the temperature of the surrounding fluid u_∞ and the convection coeficient h, as

$$\frac{\partial u}{\partial \hat{n}} = h(u - u_\infty) \tag{5.23}$$

The equivalence with (5.22) can be easily demostrated by doing $h = -\alpha/\beta$ and $u_\infty = \gamma/\alpha$

The weak form of the problem (5.5) can be written as

$$\int_\Omega \nabla u \nabla v \, d\Omega - \int_{\Gamma_1} v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma - \int_{\Gamma_2} v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma - \int_{\Gamma_3} v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma = \int_\Omega f v \, d\Omega$$

where the integral of the second term of (5.5) was divided into three sections $\Gamma = \Gamma_1 + \Gamma_2 + \Gamma_3$ corresponding to each boundary condition. Additionaly, we select $u \in H^1$, and in order to cancel the integral over Γ_1 we select $v \in H_0^1$. then the weak form becomes

$$\int_\Omega \nabla u \nabla v \, d\Omega - \int_{\Gamma_2} v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma - \int_{\Gamma_3} v \, \langle \nabla u, \mathrm{n} \rangle \, d\Gamma = \int_\Omega f v \, d\Omega$$

Where the integral over Γ_3 can be replaced by the Fourier boundary condition (5.23) in the following way,

$$\int_{\Gamma_3} \langle v \nabla u, \hat{n} \rangle \, d\Gamma = \int_{\Gamma_3} v \frac{\partial u}{\partial \hat{n}} d\Gamma = \int_{\Gamma_3} v \, h(u - u_\infty) d\Gamma_3 \tag{5.24}$$

Expresing u and v in terms of a base of a certain finite space of functions, we have

$$\begin{aligned} u &= \sum_{i \in I} u_i w_i(\boldsymbol{x}) + \sum_{i \in \Gamma_1} g_i w_i(\boldsymbol{x}) \\ v &= \sum_j v_j w_j(\boldsymbol{x}) \end{aligned} \tag{5.25}$$

and replacing (5.25) into (5.24), we have,

$$\begin{aligned} &\int_{\Gamma_3} v \, h(u - u_\infty) d\Gamma_3 = \\ &\qquad \sum_j v_j \sum_i u_i \int_{\Gamma_3} h \, w_j(\boldsymbol{x}) \, w_i(\boldsymbol{x}) d\Gamma_3 - \sum_j v_j \int_{\Gamma_3} h \, u_\infty \, w_j(\boldsymbol{x}) d\Gamma_3 \end{aligned} \tag{5.26}$$

defining the matrix C_{ji} as

$$C_{ji} = \int_{\Gamma_3} h \, w_j(\boldsymbol{x}) \, w_i(\boldsymbol{x}) d\Gamma_3$$

and the vector m_j as

$$m_j = \int_{\Gamma_3} h \, u_\infty \, w_j(\boldsymbol{x}) \, d\Gamma_3$$

then (5.26) can be written as

$$\int_{\Gamma_3} v \, \langle \nabla u, \hat{n} \rangle \, d\Gamma_3 = \langle Cu, v \rangle - \langle m, v \rangle \tag{5.27}$$

The discrete form of the Laplace equation with fourier boundary conditions can be written as

$$\langle A\, u, v \rangle - \langle C\, u, v \rangle + \langle m, v \rangle = 0 \tag{5.28}$$

The complete solution including all the boundary conditions is

$$\left\langle A^D\, u, v \right\rangle + \left\langle (Ag)^D, v \right\rangle - \langle p, v \rangle - \langle C\, u, v \rangle + \langle m, v \rangle \; = 0 \tag{5.29}$$

and because it should be valid for any arbitrary v

$$(A - C)^D u = \ell - d + p - m \tag{5.30}$$

where $(A - C)^D \in \mathbb{R}^{(\Omega - \Gamma_1) \times (\Omega - \Gamma_1)}$ and ℓ, d, p, and $m \in \mathbb{R}^{(\Omega - \Gamma_1)}$

Chapter 6

Afin Transformations

6.1 Change of variable in an integral

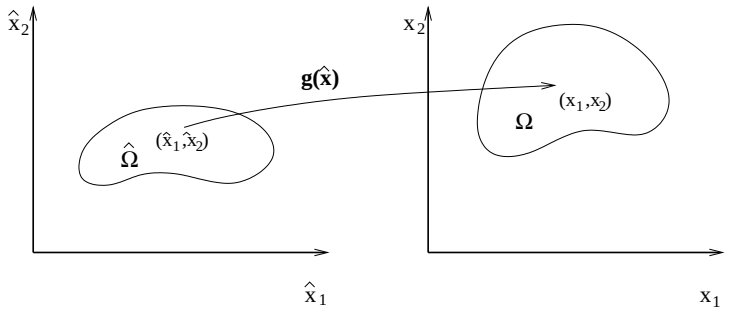


Figure 6.1: Change of variable

Let f be a scalar function defined over a domain Ω and let $\mathbf{x} \in \Omega$. The integral of $\int_\Omega f(\mathbf{x})d\mathbf{x}$ extended over a region Ω in the x, y coordinate system can be transformed into an integral $\int_{\hat{\Omega}} F(\hat{\mathbf{x}})d\hat{\Omega}$ extended over a domain $\hat{\Omega}$ in the $\hat{x}\hat{y}$ plane. Next, we are going to study the relationship between the regions Ω and $\hat{\Omega}$ and the integrals $f(\mathbf{x})$ and $F(\hat{\mathbf{x}})$. Variables $\mathbf{x}$ and $\hat{\mathbf{x}}$ are related by function $\mathbf{g}$ as $\mathbf{g}(\hat{\mathbf{x}}) = \mathbf{x}$. Notice that $\mathbf{g}(\mathbf{x})$ is a vector application, so it is composed of scalar functions g_i as

$$\mathbf{g}(\hat{\mathbf{x}}) = \begin{bmatrix} g_1(\hat{x_1}, \hat{x_2}) \\ g_2(\hat{x_1}, \hat{x_2})) \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{6.1}$$

Geometrically, it can be considered that the two equations 6.1 define an application that takes a point $(\hat{x_1}, \hat{x_2})$ in the plane $\hat{x_1}\hat{x_2}$ and corresponds to a point (x, y) in the xy plane. The set of $\hat{\Omega}$ points in the $\hat{x_1}\hat{x_2}$ plane is mapped into the set Ω in the xy plane as is represented in figure 6.1. Sometimes the system of equations 6.1 can be solved for the $\hat{x}$'s variables in function of the x's variables. When this is possible, we can express the result in the form

$$\hat{\mathbf{x}} = \mathbf{g}^{-1}(\mathbf{x}).$$

These equations define an application from the x_1x_2 plane to the $\hat{x_1}\hat{x_2}$ plane and are the *inverse application* of $\mathbf{g}(\hat{\mathbf{x}})$, as defined in (6.1), because they transform the points from Ω into $\hat{\Omega}$. Among these applications, those called *one-to-one* applications are of special importance. They transform the *different* points from $\hat{\Omega}$ into *different* points in Ω. In other words, two different points in $\hat{\Omega}$ are not mapped into the same point in Ω by a one-to-one application.

We will consider applications for which the functions g_1, g_2 are continuous and have continuous partial derivatives $\partial g_i/\partial x_j$ for $i, j = 1, 2$. For functions g_i^{-1} we make similar assumptions. These considerations are not very rigorous given that this is valid for most applications resulting from practical problems.

The formula for transformation of double integrals can be written in the following way:

$$\int_{\Omega} f(\mathbf{x}) d\Omega = \int_{\hat{\Omega}} f(\mathbf{g}(\hat{\mathbf{x}})) \det \left| \frac{\partial g_i}{\partial x_j} \right| \, d\hat{\Omega}. \tag{6.2}$$

Where the factor $\det |\partial g_i/\partial x_j|$ that appears in the integral of the right-hand side of the equation is called the Jacobian of the transformation

$$\det \left| \frac{\partial g_i}{\partial x_j} \right| = \begin{vmatrix} \dfrac{\partial g_1}{\partial x_1} & \dfrac{\partial g_1}{\partial x_2} \\ \dfrac{\partial g_2}{\partial x_1} & \dfrac{\partial g_2}{\partial x_2} \end{vmatrix}. \tag{6.3}$$

6.2 Transformation of a Standard Element

Let k represent a triangular finite element in the x_1x_2 plane and let $\hat{k}$ be a finite element in the $\hat{x}_1\hat{x}_2$ plane with vertex $\hat{A}, \hat{B}$, and $\hat{C}$ as defined in figure

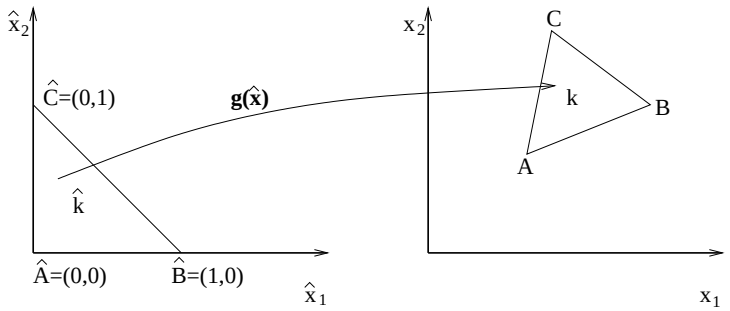


Figure 6.2: Transformation to a standard element

6.2. Then there is a mapping function $\mathbf{g}(\mathbf{x})$ that transforms any point in the element $\hat{k}$ into an element k. Such transformations are unique to each triangle k and can be calculated in the following way:

$$\begin{aligned} \mathbf{g}(\hat{\mathbf{x}}) \quad &: \mathbb{R}^2 \to \mathbb{R}^2 \\ &\hat{k} \to k \end{aligned}$$

where

$$\mathbf{g}(\hat{A}) = A \tag{6.4}$$

$$\mathbf{g}(\hat{B}) = B \tag{6.5}$$

$$\mathbf{g}(\hat{C}) = C. \tag{6.6}$$

Equations 6.4, 6.5, and 6.6 conform to a set of six equations that give rise to the following afin transformation:

$$\mathbf{g}(\hat{\mathbf{x}}) = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{6.7}$$

The matrix M plays the role of scaling and rotating while vector b is responsible for the translation to the origin. It can be shown that a transformation defined in this way is one-to-one and preserves the proportion and the relative position among the transformed points. For example, a point $\mathbf{p}$ located half way between A and B will be transformed into a point $\hat{\mathbf{p}}$ which is located half way between the points $\hat{A}$ and $\hat{B}$.

6.2.1 Computation of $\mathbf{g}(\mathbf{x})$

i.

$$\hat{A} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \text{and} \qquad \mathbf{g}(\hat{A}) = A$$

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \Rightarrow \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \tag{6.8}$$

ii.

$$\hat{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \text{and} \qquad \mathbf{g}(\hat{B}) = B$$

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \Rightarrow \begin{bmatrix} m_{11} \\ m_{21} \end{bmatrix} = \begin{bmatrix} B_1 - A_1 \\ B_2 - A_2 \end{bmatrix} \tag{6.9}$$

iii.

$$\hat{C} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad \text{and} \qquad \mathbf{g}(\hat{C}) = C$$

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \Rightarrow \begin{bmatrix} m_{12} \\ m_{22} \end{bmatrix} = \begin{bmatrix} C_1 - A_1 \\ C_2 - A_2 \end{bmatrix} \tag{6.10}$$

Then from equations 6.8, 6.9,and 6.10 the application $\mathbf{g}(\hat{\mathbf{x}})$ can be written in matrix notation as

$$\mathbf{g}(\hat{\mathbf{x}}) = \begin{bmatrix} B_1 - A_1 & C_1 - A_1 \\ B_2 - A_2 & C_2 - A_2 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}. \tag{6.11}$$

6.2.2 Base functions for the standard element

Figure 6.3 shows the equivalence between the base function ϕ_A defined over an element k and its equivalent version $\hat{\phi}_A$ defined over the standard element $\hat{k}$. Acording to the definition of the Lagrange base function, it must true that

$$\hat{\phi}_i(\hat{\mathbf{x}}_j) = \delta_{ij} \qquad \text{for} \qquad i = A, B, C \qquad \text{and} \qquad \hat{\mathbf{x}}_j = \{\hat{A}, \hat{B}, \hat{C}\}.$$

If $\mathbf{g}(\hat{\mathbf{x}})$ is the coordinate transformation function as defined in the last numeral, then for all $\mathbf{x} \in k$ with $\mathbf{x} = \mathbf{g}(\hat{\mathbf{x}})$ we have

$$\hat{\phi}_i(\hat{\mathbf{x}}) = \phi_i(\mathbf{x}). \tag{6.12}$$

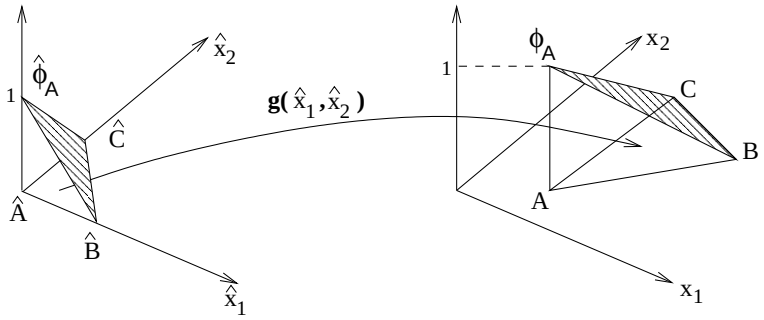


Figure 6.3: Representation of base functions for a standard element

6.2.3 Computation of integrals over a finite element

We can use the results from the change of variable equation, equation 6.2, to calculate integrals over a finite element domain by computing the integrals over a standard element in the following way.

If

$$f(\mathbf{x}) \approx \sum f(\mathbf{x}_j)\phi_j(\mathbf{x})$$

$$\int_k f(\mathbf{x})dk = \int_{\hat{k}} f(\mathbf{g}(\hat{\mathbf{x}})) \det \left| \frac{\partial g_i}{\partial x_j} \right| d\hat{k} \tag{6.13}$$

where the Jacobian $|\partial g_i/\partial x_j|$ can be computed from equation 6.11 as

$$\left| \frac{\partial g_i}{\partial x_j} \right| = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \tag{6.14}$$

and integrals over the standard finite element k can be defined as $r = x_1$ and $s = x_2$

$$\int_0^1 \int_0^{-r+1} f(r,s)\, ds\, dr. \tag{6.15}$$

6.2.4 Example

Let us consider a triangle with coordinates $A = (6,6)$, $B = (8,6)$ and $C = (7,8)$ as shown in figure 6.4. If ϕ_A, ϕ_B, and ϕ_C are the Lagrange first order polynomials defined over the triangle in the usual way, then they can

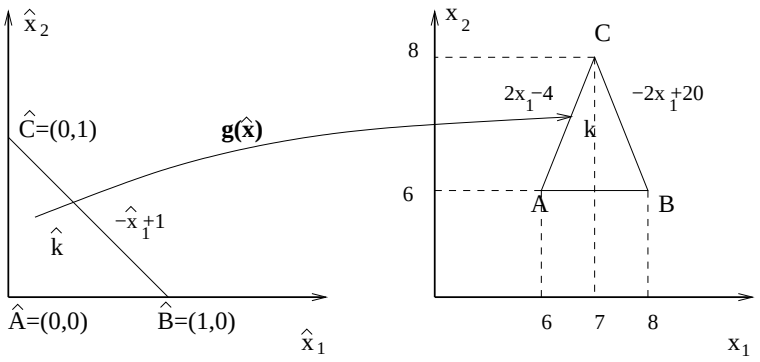


Figure 6.4: Representation of the base functions for a standard element

be written in terms of the x_1 and x_2 coordinates as follows:

$$\begin{aligned}\phi_A &= -1/2x_1 - 1/4x_2 + 11/2 \\ \phi_B &= 1/2x_1 - 1/4x_2 - 3/2 \\ \phi_C &= -3 + 1/2x_2.\end{aligned} \tag{6.16}$$

For example if we want to compute the integral of ϕ_A over the element, then we divide the integral in two parts: one from $x_1 = 6$ until $x_1 = 7$, and the second from $x_1 = 7$ until $x_1 = 8$ in the following way.

$$\int_k \phi_A dk = \int_6^7 \int_6^{2x_1-4} \phi_A dx_2\, dx_1 + \int_7^8 \int_6^{-2x_1+20} \phi_A dx_2\, dx_1.$$

By replacing the value of ϕ_A from equation 6.16 into the last equation, this integral can be calculated by hand or by using a computer algebra system to obtain

$$\int_k \phi_A dk = \frac{2}{3}.$$

To compute the value of the integral by using the method of transformation of variables exposed in this section, we need first to compute the values of the transformed function $\mathbf{g}(\hat{\mathbf{x}})$ defined in equation 6.11 in order to obtain the Jacobian and then compute the integral over the standard element. Computation of the Jacobian for the present example leads us to

$$\left|\frac{\partial g_i}{\partial x_j}\right| = \begin{bmatrix} B_1 - A_1 & C_1 - A_1 \\ B_2 - A_2 & C_2 - A_2 \end{bmatrix} = \begin{bmatrix} 8-6 & 7-6 \\ 6-6 & 8-6 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix}$$

and

$$\det \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix} = 4.$$

Then we compute the *standard* integral of $\hat{\phi}_A$ over the $\hat{k}$ domain

$$\int_{\hat{k}} \hat{\phi_A} d\hat{k} = \int_0^1 \int_0^{-x_1+1} \hat{\phi}_A(\hat{x}_1, \hat{x}_2)\, d\hat{x}_2\, d\hat{x_1}.$$

For a standard finite element $\hat{k}$ the base functions, $\hat{\phi}_j$, can be calculated from the definition of Lagrange polynomials. The following values are obtained:

$$\hat{\phi}_A = -x_1 - x_2 + 1 \tag{6.17}$$

$$\hat{\phi}_B = x_1 \tag{6.18}$$

$$\hat{\phi}_C = x_2. \tag{6.19}$$

Replacing the value for $\hat{\phi}_A$ enables us to compute the integral

$$\int_{\hat{k}} \hat{\phi_A} d\hat{k} = \int_0^1 \int_0^{-x_1+1} (-x_1 - x_2 + 1)\, d\hat{x}_2\, d\hat{x_1} = \frac{1}{6}.$$

Finally we replace this value into the change of coordinated equation (6.13) to obtain:

$$\int_k \phi_A(\mathbf{x}) dk = \int_{\hat{k}} \hat{\phi}_A(\mathbf{g}(\hat{\mathbf{x}}))\, \det \left| \frac{\partial g_i}{\partial x_j} \right| \, d\hat{k} = \frac{1}{6}\, 4 = \frac{2}{3} \tag{6.20}$$

which is the same value obtained by the direct computation of the integral.

Chapter 7

Heat Conduction Time Dependant Problems

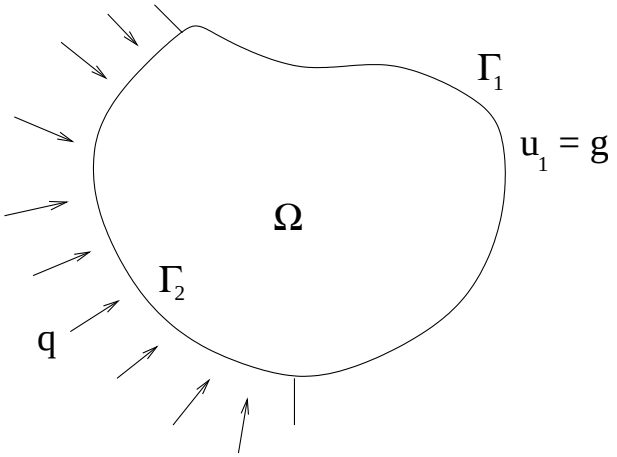


Figure 7.1: Domain definition and boundary conditions for a heat transfer problem, Notice that $\Gamma_1 \cap \Gamma_2 = \emptyset$

In the previous chapters we have used Finite Elements to solve heat conduction problems that involve temperature and heat flow boundary conditions. We will discuss in this chapter heat conduction problems that involves the time derivative. If $u = u(\mathbf{x}, t)$ represents the temperature of a point $\mathbf{x}$ at a time t, then the initial boundary value problem of the heat

conduction can be stated as

$$\frac{\partial u}{\partial t} = \nabla^2 u + f \qquad \text{in } \Omega \tag{7.1}$$

with boundary conditions:

$$u|_{\Gamma_1} = g(x), \qquad \left.\frac{\partial u}{\partial n}\right|_{\Gamma_2} = m\ q(x) \tag{7.2}$$

and initial conditions

$$u(x,0) = u_0(x) \tag{7.3}$$

To derive the finite element approximation studied in Chapter 5, the corresponding weak form must be obtained. Multiplying both sides of equation 7.1 by an arbitrary virtual function v (virtual temperature) and applying the Green's equation, we have

$$\begin{aligned}\int_\Omega \frac{\partial u}{\partial t} v &= \int_\Omega \nabla^2 u\ v + \int f\ v \\ \int_\Omega \frac{\partial u}{\partial t} v &= -\int_\Omega \nabla u \nabla v + \int_\Gamma \frac{\partial u}{\partial n} v + \int f\ v\end{aligned}$$

The right hand side of this equation can be treated as usual. That is, by expressing u and v in terms of a base of approximation we obtain the abstract form of the right hand side of the equation and then obtain the discrete form

$$\begin{aligned}\int_\Omega \frac{\partial u}{\partial t} v &= -\ a(u,v) + \ell(v) \\ \int_\Omega \frac{\partial u}{\partial t} v &= \langle -Au, v\rangle + \langle \ell, v\rangle\end{aligned} \tag{7.4}$$

Now the left hand side of the equation contains the time derivate of u that can be expressed in terms of the same base as

$$\frac{\partial u}{\partial t} = \sum_i \left(\frac{\partial u_i \phi_i}{\partial t}\right) = \sum_i \frac{\partial u_i}{\partial t}\phi_i = \sum_i \dot{u}\phi_i \tag{7.5}$$

where $\dfrac{\partial u_i}{\partial t} = \dot{u}$. Then the left hand side can be computed as

$$\int_\Omega \frac{\partial u}{\partial t} v = \int_\Omega \sum_i \dot{u_i}\phi_i \sum_j v_j \phi_j = \sum_{ij} \dot{u_i} v_j \int_\Omega \phi_i \phi_j \tag{7.6}$$

which can be expressed in matrix form as

$$\int_\Omega \frac{\partial u}{\partial t} v = \langle M\dot{u}, v\rangle \tag{7.7}$$

Incorporating this result into (7.4) gives,

$$\langle M\dot{u}, v\rangle = \langle -Au, v\rangle + \langle \ell, v\rangle \tag{7.8}$$

Because this results must be true for any function v in admissible space of approximation, we have,

$$M\dot{u} + Au = +\ell \tag{7.9}$$

7.1 Finite Difference Approximation

The time derivate of $\dot{u}$ can be approximated by its forward finite difference approximation in the following way, If u^n is the temperature at time $n\Delta t$, $u^n = u(\mathbf{x}, n\Delta t)$, then

$$\dot{u} = \frac{u^n - u^{n-1}}{\Delta t} \tag{7.10}$$

replacing (7.1) into (7.9)

$$M\left(\frac{u^n - u^{n-1}}{\Delta t}\right) + Au + g = \ell$$

$$\frac{Mu^n - Mu^{n-1}}{\Delta t} + Au = \ell - g$$

$$Mu^n - Mu^{n-1} + \Delta t(Au^n) = \Delta t(\ell - g)$$

$$Mu^n + \Delta t(Au^n) = \Delta t(\ell - g) + Mu^{n-1}$$

then the implicit form of the transient heat conduction problem

$$(M + \Delta tA)u^n = \Delta t(\ell - g) + Mu^{n-1} \tag{7.11}$$

7.2 Θ Method for time integration

A more general way of approximating the time derivate is called the Θ method for time integration, to see how this method works, let us consider a simplified form a time dependent problem

$$\frac{dx}{dt} = f(x), \quad x = x_0 \text{ at } t = 0.$$

For a given increment Δt, with $x^n = x(n\Delta t)$, this equation can be approximated by

$$\frac{x^n - x^{n-1}}{\Delta t} = f(x^n)$$

which is called the forward difference scheme or if we choose $f(x^{n-1})$ instead of $f(x^n)$ it will lead as to the backward difference scheme

$$\frac{x^n - x^{n-1}}{\Delta t} = f(x^{n-1})$$

Selection of each scheme will depend upon the problem, as each one seems to be equally approximated. A weighted averaged approximation of the right hand side of the backward and forward schemes take us to a general form

$$\frac{x^n - x^{n-1}}{\Delta t} = \theta f(x^n) + (1-\theta) f(x^{n-1})$$

with $0 \leq \theta \leq 1$. Notice that when $\theta = 0$ we have the backward difference approximation and when $\theta = 1$ we obtain the forward difference approximation.

$$M\dot{u} + Au + g^* = \ell \tag{7.12}$$

Applying the θ method approximation we have

$$M\left(\frac{u^n - u^{n-1}}{\Delta t}\right) + \theta A u^n + (1-\theta) A u^{n-1} + g^* = \ell \tag{7.13}$$

expanding the first term and grouping the terms with u^n and the terms with u^{n+1},

$$M\frac{u^n}{\Delta t} - M\frac{u^{n-1}}{\Delta t} + \theta A u^n + (1-\theta) A u^{n-1} + g^* = \ell,$$

$$\left(\frac{M}{\Delta t} + \theta A\right) u^n + \left(-\frac{M}{\Delta t} + (1-\theta)A\right) u^{n-1} + g^* = \ell.$$

Multiplying by Δt

$$(M + \Delta t\, \theta A)u^n + (-M + \Delta t(1-\theta)A)\, u^{n-1} + g^* \Delta t = \ell \Delta t,$$

which led as to the final system

$$(M + \Delta t\, \theta A)u^n = (M - \Delta t(1-\theta)A)\, u^{n-1} - g^* \Delta t + \ell \Delta t \qquad (7.14)$$

Example

As an example consider the one dimensional case of this equation. Solving for u^n we have

$$u^n = \frac{(M - \Delta t(1-\theta)A)\, u^{n-1} - g^* \Delta t + \ell \Delta t}{(M + \Delta t\, \theta A)}$$

Appendix A

Barycentric Coordinates

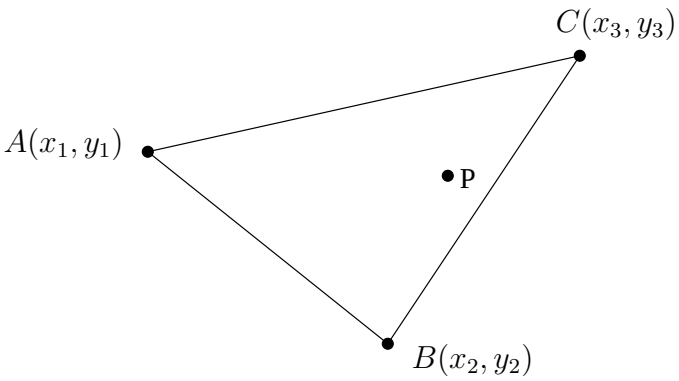


Figure A.1: Barycentric triangle coordinates

Given a triangle defined by ABC and some point P, we defined barycentric coordinates, m_1, m_2, and m_3 as the mass associated to each triangle vertex such that the mass centre is located at P, that is

$$M\,P = m_1 A + m_2 B + m_3 C$$

where M is the sum of all the mases, $M = \sum m_i$, and $A = (x_1, y_1)$ $B = (x_2, y_2)$ $C = (x_3, y_3)$. Figure A.2. Assuming that $M = 1$ then

$$m2 = -(y_1\,x - y_1\,x_3 - x_1\,y + y\,x_3 - y_3\,x + x_1\,y_3)/b \tag{A.1}$$

$$m3 = (y_1\,x - x_2\,y_1 - y_2\,x + y_2\,x_1 + x_2\,y - x_1\,y)/b \tag{A.2}$$

$$m1 = (-y_2\,x_3 + x_2\,y_3 + y\,x_3 - y_3\,x + y_2\,x - x_2\,y)/b \tag{A.3}$$

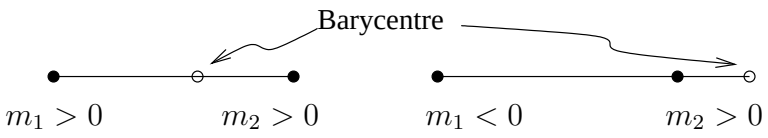


Figure A.2: Barycentric triangle coordinates

with $b = (y_1\, x_3 + y_2\, x_1 - y_2\, x_3 - x_1\, y_3 - x_2\, y_1 + x_2\, y_3)$

If all the masses are equal then then mass centre is equal to the geometric centre of the triangle. if $m_3 = 0$ and m_1 and $m_2 > 0$ then the barycentre will be located in the segment of line that joints the two points. If however, one of the masses is negative then the barycentre's will be located over the line that crosses the two points but outside of the segment that joint the two points. See figure A.2. This can be easily proved by setting the origin of the coordinate system at point 1, the the barycentre is given by $m_1 x_1 + m_2 x_2 = M\bar{x}$, as $x_1 = 0$ and solving for $\bar{x}$, $\bar{x} = x\,({}^{m_2}/_{M})$. Because $m_1 < 0$ then $({}^{m_2}/_{M}) > 1$ therefore $\bar{x} > x$.

This concept can be generalised to conclude that if a point P is outside a triangle ABC then at least one its barycentric coordinates is equal to zero. The prof is left to the enjoyment of the reader.

Appendix B

Gradient

B.1 Computation of the gradient of a function

Let $\psi = u'$ where u' means $\frac{\partial u}{\partial x_i}$ with $i = 1..3$

$$\int_\Omega (\psi - u')\, v = 0$$

defining

$$\psi = \psi_i\, w_i, \qquad u = u_k\, w_k, \qquad v = v_j\, w_j$$

and replacing this into the previous expression we have:

$$\int_\Omega \left(\psi_i\, w_i - u_k\, w_k'\right)(v_j\, w_j) = 0 \tag{B.1}$$

$$\int_\Omega \psi_i\, w_i\, v_j\, w_j - \int_\Omega u_k\, w_k'\, v_j\, w_j = 0 \tag{B.2}$$

$$\psi_i\, v_j \int_\Omega w_i\, w_j - u_k\, v_j \int_\Omega w_k'\, w_j = 0 \tag{B.3}$$

defining

$$\int w_i w_j = H_{ij} \qquad \int w_k' w_j = B_{kj}$$

$$\psi_i\, v_j H_{ij} - u_k\, v_j\, B_{kj} = 0$$

as H_{ij} is symmetric

$$v_j(H_{ij}\psi_i) - v_j(B_{kj}u_k) = 0$$

it is valid for all v_j

$$[\mathbf{H}]\boldsymbol{\psi} = [\mathbf{B}]^T u_k = \mathbf{L}$$

Bibliography

[1] Terrence Akay. *Métodos Numéricos aplicados a la Ingeniería*. Limusa.

[2] F. P Beer and R. Johnston Jr. *Mechanics of Materials*. McGraw-Hill, New York, second edition, 1992.

[3] Dietrich Braess. *Finite Elements*. Cambridge University Press, 1997.

[4] Susanne C. Brenner and L. Ridgway Scott. *The Mathematical Theory of Finite Elements*. Springer-Verlag, 1994.

[5] Richard Burden. *Analisis Numerico*. Grupo Editorial Iberoamérica, 1985.

[6] Luz Myriam Echevery. Notas de clase del curso de análisis numérico. Departamento de Matemáticas, Universidad de los Andes.

[7] G. Golub and C. Van Loan. *Matrix Computations*. The John Hopkings University Press, third edition, 1996.

[8] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, November 1996.

[9] Morris W. Hirsch and Stephen Smale. *Differential Equations, Dynamical Systems, and Linear Algebra*. Academic Press, 1974.

[10] Noboru Kikuchi. *Finite Element Methods in Mechanics*. Cambridge University Press, 1986.

[11] E. Kreyszig. *Introductory Functional Analysis With FES Applications*. John Wiley & Sons, 1978.

[12] Gilbert Strang. *Introducction to Applied Mathematics*. Wellesley Cambridge press, 1986.

[13] Jose Rafael Toro. Notas de clase de análisis numérico. Departamento de Ingeniería Mecánica, Universidad de los Andes, 1988.

[14] Lloyd N. Trefethen and David Bau. *Numerical Linear Algebra*. SIAM, 1997.

[15] O. C. Zienkiewicz and R L Taylor. *Finite Element Method*. Butterworth-Heinemann, 5th edition, 2000.