Universidade do Porto

Faculdade de Engenharia

FEUP

Fracture mechanics using the Natural Neighbour Radial Point Interpolation Method

by:
José Manuel Cruz Azevedo

Thesis submitted to Faculdade de Engenharia da Universidade do Porto as a requirement to obtain the degree of MSc in Mechanical Engineering

under the supervision of:
Professor Jorge Américo Oliveira Pinto Belinha
and
Professor Lúcia Maria de Jesus Simas Dinis
and
Professor Renato Manuel Natal Jorge

February 26, 2013

Agradecimentos

Ao Doutor Jorge Belinha por ter sido literalmente um orientador durante o período de elaboração deste trabalho, o seu entusiasmo e disponibilidade fizeram com que o meu último semestre de estudante fosse o mais produtivo. À Professora Lúcia Dinis pela motivação e por me ter direcionado para o Doutor Jorge Belinha. Ao professor Renato Natal Jorge por me ter introduzido à mecânica computacional.

À Faculdade de Engenharia da Universidade do Porto por me dar a possibilidade de ter orgulho no meu local de trabalho nestes últimos seis anos e meio.

À minha mãe, Ana Maria, por sempre me ter dado tudo o que precisei mas nem sempre tudo o que quis.

À Maria Alexandra porque para além de ser única, e de ter paciência e vontade de me ter como parceiro de vida, demonstra uma sabedoria e maturidade incomuns numa pessoa que ainda viveu tão poucos anos.

A toda a minha família, em especial às minhas irmãs Ana Clara e Mafalda que ao não levarem nada muito a sério me ajudam a não me preocupar tanto com o trabalho, ao meu pai e à tia Nela, que sempre se demonstrou pronta a ajudar.

A todos os meus amigos, em especial ao Casimiro, Miguel e Salgueiro pela contribuição e ajuda que tiveram na minha formação, não só como pessoa mas também como engenheiro.

Abstract

The prediction of the crack propagation is an important engineering problem. The application of the Natural Neighbour Radial Point Interpolation Method (NNRPIM) to fracture mechanics is presented in this thesis. The NNRPIM is a meshless method used to solve computational mechanics problems. It has been extended to numerous others fields.

In this work, the crack propagation is numerically simulated using the NNRPIM. The crack is iteratively extended in line segments. At each iteration, the crack propagation direction is obtained from the calculated stress field. The crack propagation direction is calculated using the maximum circumferential stress criterion. The discontinuities in the domain are modelled taking advantage of the natural neighbour concept.

In order to numerically validate all the work developed in this thesis, the NNRPIM is used to predict the crack path on several benchmark examples previously solved with other methods.

Keywords: meshless methods; fracture; crack path.

Sumário

A previsão do caminho de propagação de fenda é um problema importante de engenharia. Nesta tese, é feita a aplicação, à mecânica da fractura, do método sem malha “Natural Neighbour Radial Point Interpolation Method” (NNRPIM). O NNRPIM é um método sem malha desenvolvido para resolver problemas de mecânica computacional. Este método foi já aplicado a vários campos da mecânica computacional e biomecânica.

Neste trabalho, a propagação de fenda é simulada usando o NNRPIM. A fenda é iterativamente estendida em segmentos de recta. Em cada iteração, a direcção de propagação de fenda é obtida através do campo de tensões do problema em questão. A direcção de propagação de fenda é calculada usando o critério da máxima tensão circunferencial. As descontinuidades no domínio do problema são tidas em consideração tirando partido do conceito de vizinho natural.

De forma a avaliar o todo o trabalho desenvolvido, o NNRPIM é usado para fazer a previsão do caminho da fenda em vários exemplos numéricos, previamente resolvidos usando outros métodos.

Palavras chave: métodos sem malha; fractura; caminho de fenda.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this work, a meshless method, the Natural Neighbour Radial Point Interpolation Method (NNRPIM) [1] is used to solve linear elastic fracture mechanics problems.

Meshless methods [2, 3] are a class of numerical methods (used to solve differential equations) that do not require the problem to be discretised in a mesh of elements. This is a different concept from the one used in the more widely adopted Finite Elements Method (FEM) [4], in which the domain of the problem is divided into smaller parts, called elements, originating a computational mesh, containing all the elements and the information about all the pre-established element connectivity relations. In a truly meshless method the typical concept of a mesh does not exist, instead, the problem is discretised in a set of points or nodes, without pre-establishing any sort of connectivity between them. The connectivity of those nodes is assured by the concept of influence domain [5], which will be later explained in detail.

The FEM presents a significant limitation when applied to problems with complex geometries. In this case it is common the generation of highly distorted elements. Another FEM limitation is the extra computational cost in remeshing problems with moving discontinuities, such as the prediction of the crack path, which reduces significantly the FEM efficiency [3]. These problems do not occur in a meshless method because it does not use elements.

1.1 Meshless methods

Meshless methods were created in order to eliminate part of the difficulties associated with the reliance on a computational mesh to construct the approximation [3]. One of the first meshless methods was the Smooth Particle Hydrodynamics (SPH) [6], originally used to solve problems in astrophysics and later applied to computational mechanics, the SPH was based on a strong form. In the 1990s other meshless methods were created that were based on a weak form. The Element-free Galerkin Method (EFGM) [7] was one of the first meshless methods based on a global weak form. The EFGM was based on the Diffuse Element Method (DEM) [8] which was one of the first meshless methods to use the moving least squares approximation to compute the test functions.

There are also meshless methods based on a local weak form, the most popular being the Meshless Local Petrov-Galerkin (MLPG) [9]. The main difference between the MLPG and methods that use a global weak form is that the local weak forms are generated over local overlapping subdomains rather than the whole domain of the problem [3].

Although the above mentioned methods were successfully applied to solid mechanics, there are some unsolved issues, such as the computational difficulty on imposing the essential and natural boundary conditions, due to the lack of the delta Kronecker property on approximation functions. In order to solve this problem, the Point Interpolation Method (PIM) [10] was created. In the PIM, the test functions are interpolant and have the Kronecker delta property making the imposition of the essential and natural boundary conditions as easy as in FEM. The original PIM used only polynomials as its basis functions and thus special techniques were required to guarantee a successful computation of the interpolation functions for a arbitrarily chosen set of points [5]. In order to simplify the computation of the interpolation functions, and to eliminate some singularities that occurred in the PIM, the Radial Point Interpolation Method (RPIM) [5] was proposed. This method uses a Radial Basis Function (RBF) combined with a polynomial basis to construct the interpolation functions used in the solution of partial differential equations.

Using the RPIM as a starting point, the NNRPIM was developed. The NNRPIM has the important advantage of not relying on a background integration mesh, thus being a truly meshless method. A truly meshless method performs both interpolation and integration without a mesh [11]. This method uses the mathematical concept of the Voronoï diagram [12] and Delaunay tessellation [13] in order to obtain the natural neighbours on an interest point. After the determination of the natural neighbours, the influence domains and the set of integration points are created. Contrary to the FEM, where geometrical restrictions are imposed to guarantee the convergence of the method, in the NNRPIM there are no such restrictions, which allows a totally random node distribution for the discretised problem [1].

Some of the major advantages of the meshless methods are [14]:

- The methods can provide more accurate approximations for structures with complex geometries when compared with the FEM.

- The test functions are constructed in terms of higher-order continuous weight functions and possess compact support.

- Possessing significant advantages in handling problems with large deformations, moving discontinuities such as crack propagation in dynamic fracture and phase transformation in the development of advanced materials.
- Nodes can be easily added (h-adaptivity) on portions where refinement is needed for a higher solution accuracy.
- The coupling possibility with FEM or boundary element methods (BEM) to avoid their inherent drawbacks.

More recently, other meshless methods were developed and applied to computational mechanics, one of them being the Maximum-entropy Meshfree Method (MEMM) [15], this method attempts to solve some of the problems of the existing meshless methods, specifically problems related to volumetric locking. Another one is the Generalised Meshfree (GMF) [16], it uses convex approximation and poses e weak Kronecker delta property.

1.2 The natural neighbour radial point interpolation method

The NNRPIM was developed at FEUP with the intent of being an improvement on radial point interpolation techniques. This meshless method was first applied in obtaining and solving the equilibrium equation of elastostatics using a Galerkin displacement based formulation [1]. The NNRPIM is the product of the combination of radial point interpolators with the natural neighbour geometric concept [17].

The RPIM uses the concept of "influence domain" to impose the nodal connectivity, instead of using the global domain of the problem. In the NNRPIM the concept of influence domain is replaced by the concept of "influence cell". In order to obtain the influence cells the NNRPIM resorts to geometrical and mathematical constructions such as the Voronoï diagrams and the Delaunay tessellation. Departing from an unstructured set of nodes, making use of the Voronoï cells, a set of influence cells is created. The Delaunay triangles are applied to create the background mesh, used in the integration of the interpolation functions. This background mesh depends on the nodal mesh, making the NNRPIM a truly meshless method. The NNRPIM interpolation functions, used in the Galerkin weak form, are constructed in a similar process to the RPIM, with some differences that modify the method's performance [1].

The NNRPIM has been extended to many fields in computational mechanics, such as the static analysis of isotropic and orthotropic plates [18] and the dynamic analysis of several solid mechanics problems [19]. The NNRPIM was also tested in more demanding applications such as the material nonlinearity [20] and the large deformation analysis [21].

1.3 Fracture mechanics and numerical methods

Numerical methods have been applied to fracture mechanics for several decades now [22]. One common problem in which numerical methods and fracture mechanics are used in conjunction, is the prediction of the crack path.

The FEM has been extensively used to predict the crack path since the 1980s [23] until very recently [24]. As was said above, the FEM is particularly unsuited to problems with complex geometries in which is difficult to align the element edges with the edges of the domain of the problem, which can be cracks. Additionally, remeshing the domain as the crack progresses is a very burdensome and costly task, using FEM. Apart from meshless methods, another method that overcomes this difficulties is the Extended Finite Element Method (XFEM) [25]. The XFEM enriches the standard finite element shape functions with additional continuous enrichment function (Ramp function) or discontinuous enrichment function (Heaviside function or Step function) to approximate the solution near an interface. The enrichment function exists only at the element's nodes that intersect the interface. This allows XFEM to accommodate elements that do not conform to the interface [26]. Several authors solved the prediction of the crack path with accurate results using XFEM [27, 28, 29], with various degrees of complexity. Another numerical method that has been recently applied to fracture mechanics and, more specifically to crack path prediction is the Edge-based Smoothed Finite Element Method (ES-FEM) [30]. The ES-FEM is a successor of the Smoothed Finite Element Method (S-FEM), which can be seen as a combination of FEMs and meshless methods [30].

Other meshless methods have also been successfully applied to crack path prediction, like the EFGM [31] and others [32, 33, 34, 35].

In this work, a significant part of the fracture mechanics analysis will be related to the prediction of the crack path for a given component or structure. The prediction of the crack path is an iterative process in which the mesh or nodal mesh have to be, to some extent, updated in every iteration [23]. In each step of the problem, the crack is extended in a straight line by a certain length, that can be calculated or just a predefined value. In order to define the crack propagation direction, in each iteration the displacement and stress fields need to be obtained. There are several criterion that define the crack propagation direction, this work uses one of the most important, the maximum circumferential stress criterion [36].

1.4 Purpose of this work

The main purpose of this work is to apply the NNRPIM to the fracture mechanics analysis of structures. More specifically, to create a numerical tool capable of automatically predicting the crack path in two-dimensional solids. In order to accomplish that, it is necessary to extend the existing NNRPIM code and formulation to fracture mechanics.

Meshless methods are inherently easier to apply to fracture mechanics problems than conventional FEMs [10], since discontinuities can be easily considered, due to the nonexistence of elements. The NNRPIM was the chosen method, because of the way the influence domains are obtained. The fact that it uses the natural neighbour concept makes it very easy and organic to model discontinuities in a problem and, more important, to automatically change the shape of the discontinuities (which model the crack path) as the problem progresses. Thus making it simple to simulate propagating cracks in solids. Additionally the NNRPIM is a very versatile method, facilitating the analysis from the point of view of solid mechanics.

All the concepts developed in this thesis were ultimately applied in numerical simulations. The developed algorithm was programmed in the commercial software Matlab, using a provided NNRPIM Matlab package.

1.5 Thesis outline

This thesis is organised, by chapter number, as follows:

1. The thesis is **introduced** and a brief **state of the art**, concerning meshless methods and fracture mechanics is given. Also, the purpose of this work is defined.

2. The necessary **solid mechanics fundamentals** are presented and explained. This information is necessary to understand the inner workings of the NNRPIM and also acts as the basis for the subsequent fracture mechanics analysis.

3. All the **fracture mechanics analysis** elaborated during the course of this work is demonstrated and explained. Before that, once again, the essential fracture mechanics fundamentals are shown and explained. One important point of this chapter is the explanation and demonstration of the algorithm necessary to numerically simulate the crack path.

4. The concepts defined and explained in the previous chapters are put to the practice. Various **numerical examples** are programmed and simulated using Matlab, all the relevant results are displayed in this chapter.

5. Finally all the important **conclusions** are discussed and a brief reflection of possible future work is made.

Chapter 2

Meshless method

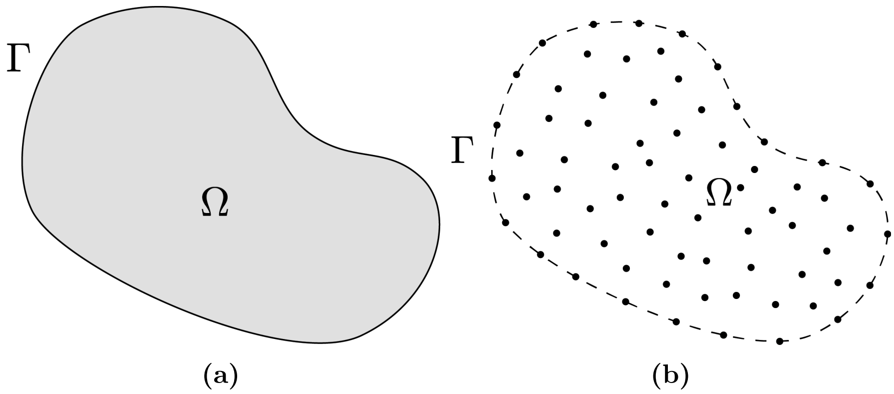


Figure 2.1: Discretisation procedure: (a) continuous domain; (b) discretised in a nodal mesh.

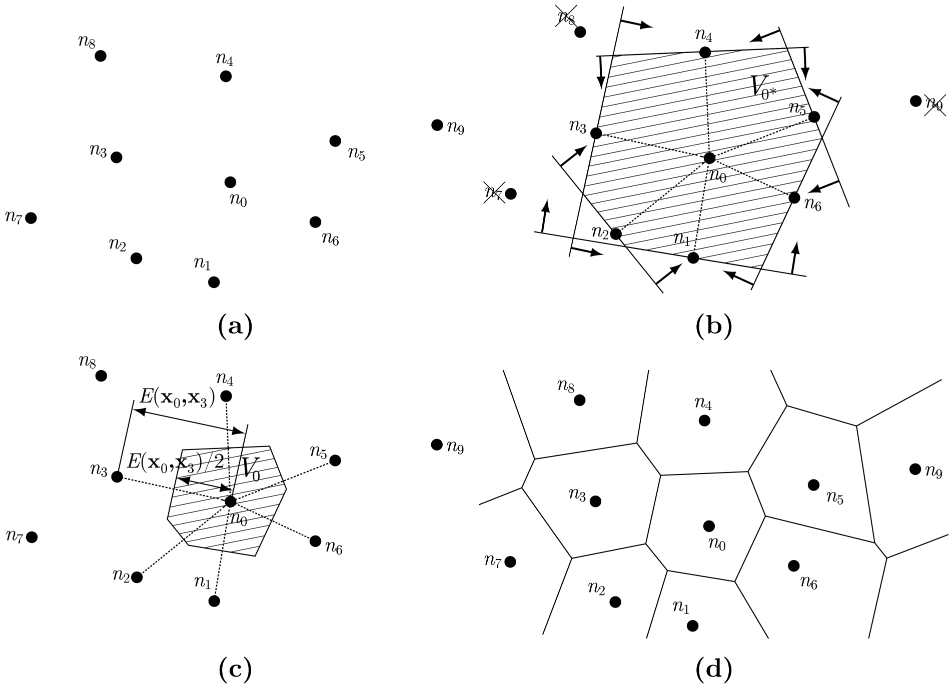
The majority of the work developed throughout the realisation of this thesis was based on the NNRPIM. For that reason, to facilitate the adaptation of the NNRPIM to fracture mechanics, it is necessary to get a good understanding of the method, and also to make sure that it is able to achieve accurate results. Also, before studying the behaviour of structures from the point of view of fracture mechanics it is necessary to understand the solid mechanics fundamentals. In figure 2.1, the discretisation procedure in meshless methods is displayed. It is possible to verify that only nodes are used to discretise de domain of the problem.

In this chapter, firstly the basic NNRPIM formulation is presented, afterwards, the theoretical and practical background concerning the application of the NNRPIM to linear elastic analysis is explained. Finally, the solid mechanics fundamentals used throughout this thesis are briefly presented.

2.1 NNRPIM formulation

The NNRPIM is a meshless method based on a local weak form, in which the weak form is supported only in subdomains of the problem. In this meshless method, these subdomains define how the nodes are connected i. e. the **nodal connectivity**. The nodal connectivity is, in turn, defined by the **natural neighbour** concept. The natural neighbours are determined making use of the Voronoï diagram. The **integration scheme** is also determined using the Voronoï diagram. All the above mentioned terms are explained in this section.

2.1.1 Natural neighbours



(a) **(b)** **(c)** **(d)**

Figure 2.2: Natural neighbours determination: (a) initial set of nodes; (b) final cell containing only neighbour nodes; (c) Voronoï cell; (d) Voronoï diagram.

The concept of natural neighbours was firstly introduced for data fitting and smoothing [1]. The theory behind it, is applicable to a $\mathscr{D}$-dimensional space [1] however, in this work, only two-dimensional problems are considered, therefore a two-dimensional Euclidian space $\mathbb{R}^2$ example is shown. Considering the set $\mathbf{N}$ of N distinct nodes,

$$\mathbf{N} = \{n_1, n_2, \ldots, n_N\} \in \mathbb{R}^2. \tag{2.1}$$

The Voronoï diagram of $\mathbf{N}$ is the partition of the domain defined by $\mathbf{N}$ in sub-regions V_I, closed and convex [1]. V_I is the geometric place where all points are closer to n_I than to any other node, the sub-regions V_k are the "Voronoï cells" which form the Voronoï diagram, with $k = 1, \ldots, N$ [1]. In mathematical terms, the Voronoï cell is defined by

$$V_I = \left\{ \boldsymbol{x} \in \mathbb{R}^2 : E_n(\boldsymbol{x}, \boldsymbol{x}_I) < E_n(\boldsymbol{x}, \boldsymbol{x}_J),\ \forall J \neq I \right\} \tag{2.2}$$

where $E_n(\boldsymbol{x}_J, \boldsymbol{x}_I)$ is the Euclidian metric norm i. e., the distance between points with coordinates defined by $\boldsymbol{x}_J$ and $\boldsymbol{x}_I$ [1]. In figure 2.2 the process of the computation of the natural neighbours is described. In figure 2.2b it is shown how the neighbour nodes are obtained. By definition, only the nodes in the perimeter of the obtained final domain, V_{0^*}, are considered neighbours nodes [1]. The Voronoï cell is obtained as figure 2.2c indicates. The cell V_0 is the homothetic form of V_{0^*}. The remaining Voronoï cells are obtained in the same manner. In the NNRPIM, the Voronoï diagram is used to create the "influence-cells", which enforce the connectivity between the nodes in $\mathbf{N}$ [1].

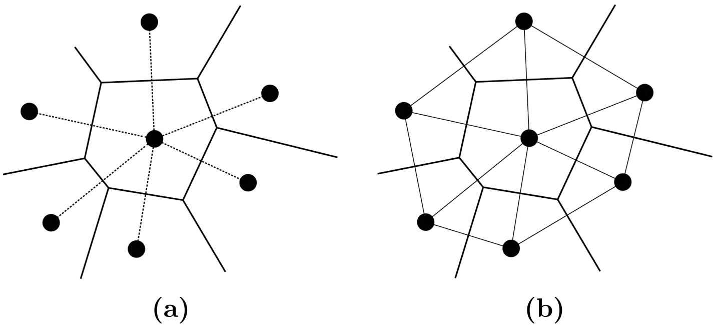


Figure 2.3: Construction of the Delaunay triangles: (a) initial Voronoï diagram; (b) the respective Delaunay triangulation.

Besides being used to define the influence domains, the natural neighbours are also utilised to construct a nodal dependent background mesh, applying the concept of Delaunay triangulation. This concept is the geometric dual of the Voronoï diagram [1]. The Delaunay triangles, which are used to create integration points, are obtained by connecting the nodes whose Voronoï cells have common boundaries with [1]. In figure 2.3 is presented an example of the construction of the Delaunay triangles.

2.1.2 Nodal connectivity

In meshless methods, the nodal connectivity is achieved by overlapping the subdomains used to obtain the local weak forms [3]. Those subdomains are called, in meshless method terminology, influence domains. Commonly, the influence domains are obtained by searching a certain amount of nodes,

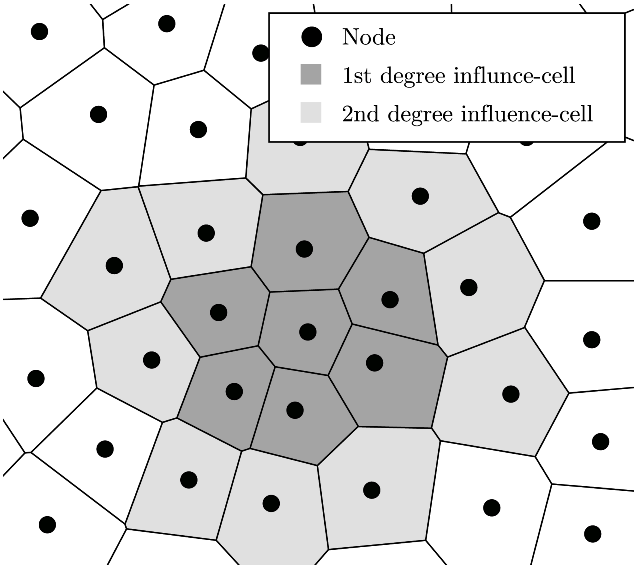


Figure 2.4: Example of a Voronoï diagram.

inside a fixed area (for 2D problems) or a fixed volume (in a 3D problem) [1]. This methodology has the following drawback: the size and shape variation of these influence domains along the problem's domain affect the performance and solution of the meshless method [1]. In order to maximise performance it is best that all the influence domains of the problem contain approximately the same number of nodes [1].

In the NNRPIM the nodal connectivity is imposed by the overlapping of the influence-cells [1]. The influence-cells are the equivalent of the influence domain in other meshless methods. In this work the terms influence domain and influence-cell are used interchangeably. Two types of influence-cells are used:

- *First degree influence-cell*: composed by the first degree natural neighbours of a given point of interest.
- *Second degree influence-cell*: composed by the first and second degree natural neighbours of a given point of interest. The second degree natural neighbours are the natural neighbours of the natural neighbours of a given point of interest.

In figure 2.4 a Voronoï diagram and an example of each type of influence-cell is presented.

2.1.3 Numerical integration

In the NNRPIM, the integration scheme is based on the Voronoï tesselation and Delaunay triangulation concepts [1]. The integration points are generated using the following procedure [1]:

1. Construct the Voronoï cells and identify the intersection points of the neighbour edges, P_{Ii}.
2. Obtain the middle points, M_{Ii}, between node I and its neighbour nodes.
3. Divide the Voronoï cells in n quadrilateral sub-cells, S_{Ii}.
4. Create the integration point(s) in each sub-cell.

In figure 2.5 there is a schematic representation of the process.

Its important to notice that in the case of regular nodal meshes, the middle points M_{Ii} are coincident with P_{Ii} thus leading to triangular instead of quadrilateral shaped Voronoï sub-cells [1]. This occurrence is shown in figure 2.5 (d, e and f).

Any Voronoï cell, V_I, with n natural neighbour nodes (of node I), has n sub-cells, S_{Ii}, where

$$A_{V_I} = \sum_{i=1}^{n} A_{S_{Ii}}, \quad \forall A_{S_{Ii}} \geq 0 \tag{2.3}$$

with A_{V_I} being the Voronoï cell area and $A_{S_{Ii}}$, the area of the Voronoï sub-cell [1]. If the Voronoï cells are a partition, without gaps, of the global domain, then, the set of sub-cells are also a partition, without gaps, of the global domain [1]. Starting from the geometrical shapes seen above, numerous integration schemes can be constructed, in the NNRPIM, the integration scheme is based on the Gauss-Legendre numerical integration [1].

Integration scheme – order 0

In the NNRPIM, the integration scheme can go up to order k, which, in practical terms, is an indication of the number of integration points generated. In the specific case of this work, only order 0 was used, so only that occurrence is explained, the remaining cases are explained in [1]. The coordinates of each integration point are calculated on each sub cell, as indicated in figure 2.6 and expressions (2.6), where $\boldsymbol{x}_i = \{x_i, y_i\}$. The weight of each integration point is the area of the respective sub-cell. Therefore, considering figure 2.6b, the area of the triangle sub-cell is defined by

$$A_{I\triangle} = \text{abs}\left(\frac{1}{2}\begin{vmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{vmatrix}\right) \tag{2.4}$$

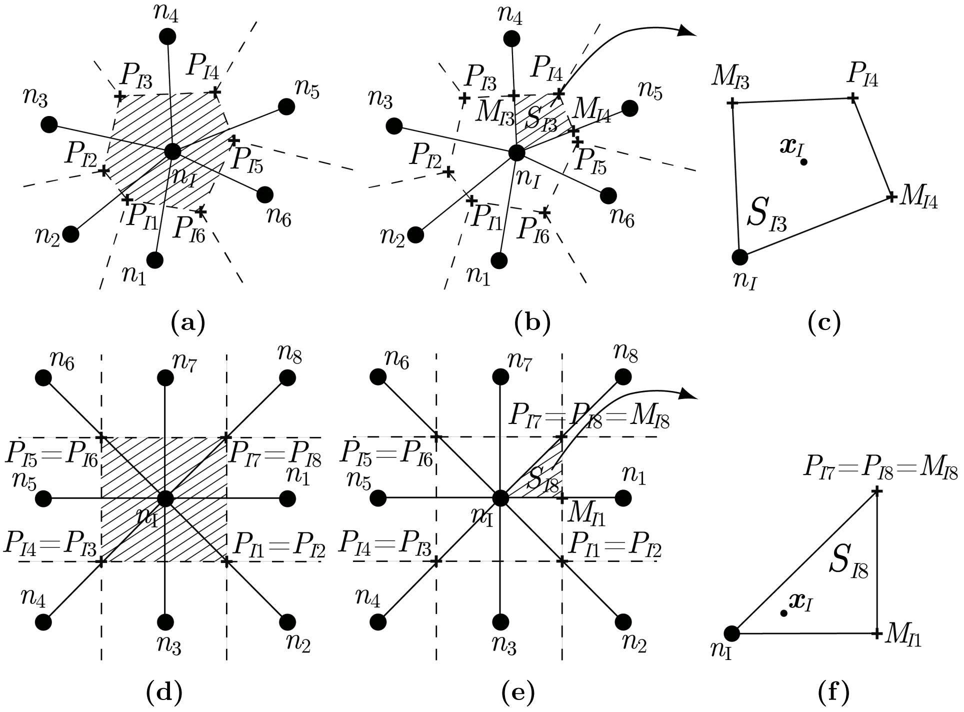


Figure 2.5: Construction of the integration mesh: (a) initial Voronoï diagram; (b) the middle points M_{Ii} and the respective generated quadrilaterals; (c) the quadrilateral $\overline{M_{I3}P_{I4}M_{I4}n_I}$; (d), (e) and (f) same procedure for regular nodal meshes.

and for the quadrilateral shape, figure 2.6a, the area is

$$A_{I\square} = \text{abs}\left(\frac{1}{2}\begin{vmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{vmatrix} + \begin{vmatrix} x_4 - x_1 & y_4 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{vmatrix}\right). \tag{2.5}$$

The coordinates of the integration points are:

$$\begin{aligned} \boldsymbol{x}_I &= \frac{1}{4}\sum_{i=1}^{4}\boldsymbol{x}_i \quad \text{for the quadrilateral,} \\ \boldsymbol{x}_I &= \frac{1}{3}\sum_{i=1}^{3}\boldsymbol{x}_i \quad \text{for the triangle.} \end{aligned} \tag{2.6}$$

2.1.4 Computation of the test functions

The NNRPIM test functions are obtained combining radial basis functions with polynomial basis functions.

Considering an approximation function $\boldsymbol{u}(\boldsymbol{x})$, in an influence domain Ω_I discretised by set of arbitrarily distributed nodes $N_i(\boldsymbol{x}_i)$ $(i = 1, 2, ..., n)$.

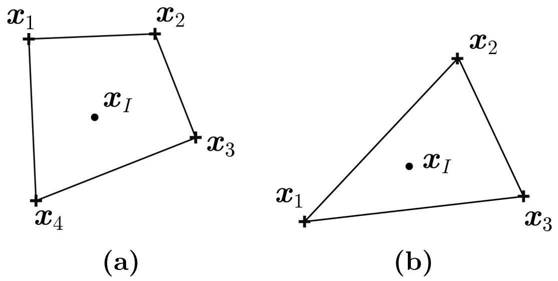


Figure 2.6: Generation of integration points: (a) quadrilateral sub-cell; (b) triangular.

n is the number of nodes in the influence domain of $\boldsymbol{x}_I$. Nodal function value is assumed to be $\boldsymbol{u}_i$ at the node $\boldsymbol{x}_i$. The NNRPIM constructs the approximation function $\boldsymbol{u}(\boldsymbol{x}_I)$ to pass through all these nodes using a radial basis function $B_i(\boldsymbol{x})$ and polynomial basis function $P_j(\boldsymbol{x})$

$$\boldsymbol{u}(\boldsymbol{x}_I) = \sum_{i=1}^{n} B_i(\boldsymbol{x}_I)a_i + \sum_{j=1}^{m} P_j(\boldsymbol{x}_I)b_j = \boldsymbol{B}^T(\boldsymbol{x}_I)\boldsymbol{a} + \boldsymbol{P}^T(\boldsymbol{x}_I)\boldsymbol{b} \tag{2.7}$$

where $\boldsymbol{x}_I$ is a point of interest in the influence domain, a_i is the non-constant coefficient for $B_i(\boldsymbol{x})$ and b_j the non-constant coefficient for $p_i(\boldsymbol{x})$ (usually, $m < n$) [5]. The vectors are defined [5] as

$$\begin{aligned} \boldsymbol{a}^T &= [a_1, a_2, a_3, \dots, a_n] \\ \boldsymbol{b}^T &= [b_1, b_2, \dots, b_m] \\ \boldsymbol{B}^T(\boldsymbol{x}) &= [B_1(\boldsymbol{x}), B_2(\boldsymbol{x}), B_3(\boldsymbol{x}), \dots, B_n(\boldsymbol{x})] \\ \boldsymbol{P}^T(\boldsymbol{x}) &= [P_1(\boldsymbol{x}), P_2(\boldsymbol{x}), \dots, P_m(\boldsymbol{x})]. \end{aligned} \tag{2.8}$$

The basis functions are function of co-ordinates $\boldsymbol{x}^T = [x, y]$ for two-dimensional problems. A radial basis function has the following general form

$$B_i(\boldsymbol{x}) = B_i(r_i) = B_i(x, y) \tag{2.9}$$

where r_i is a distance between interpolating point (x, y) and the node (x_i, y_i) [5]. This distance, in the Euclidian two-dimensional space, is expressed as

$$r_i = [(x - x_i)^2 + (y - y_i)^2]^{1/2}. \tag{2.10}$$

A polynomial basis function has the following monomial terms

$$\boldsymbol{P}^T(\boldsymbol{x}) = [1, x, y, x^2, xy, y^2, \dots], \tag{2.11}$$

in this work, a constant basis is used

$$\boldsymbol{P}^T(\boldsymbol{x}) = [1], \quad m = 1. \tag{2.12}$$

The coefficients a_i and b_i in equation (2.7) are determined by enforcing the interpolation to pass through all n nodes within the influence domain [5]. The interpolation at the kth node is defined by,

$$\boldsymbol{u}_k = u(x_k, y_k) = \sum_{i=1}^{n} B_i(x_k, y_k)a_i + \sum_{j=1}^{m} P_j(x_k, y_k)b_j, \quad k = 1, 2, \ldots, n. \tag{2.13}$$

The inclusion of the following polynomial term is an extra-requirement that guarantees unique approximation [5],

$$\sum_{i=1}^{n} P_j(x_i, y_i)a_i = 0, \quad j = 1, 2, \ldots, m. \tag{2.14}$$

The computation of the shape functions are written in matrix form as

$$\begin{bmatrix} \boldsymbol{B}_0 & \boldsymbol{P}_0 \\ \boldsymbol{P}_0^T & 0 \end{bmatrix} \begin{Bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{u}^e \\ 0 \end{Bmatrix} \quad \text{or} \quad \boldsymbol{G} \begin{Bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{u}^e \\ 0 \end{Bmatrix} \tag{2.15}$$

where the vector for function values is defined as

$$\boldsymbol{u}^e = [u_1, u_2, u_3, \ldots, u_n]^T, \tag{2.16}$$

matrix $\boldsymbol{B}_0$ is represented as,

$$\boldsymbol{B}_0 = \begin{bmatrix} B_1(x_1, y_1) & B_2(x_1, y_1) & \cdots & B_n(x_1, y_1) \\ B_1(x_2, y_2) & B_2(x_2, y_2) & \cdots & B_n(x_2, y_2) \\ \vdots & \vdots & \ddots & \vdots \\ B_1(x_n, y_n) & B_2(x_n, y_n) & \cdots & B_n(x_n, y_n) \end{bmatrix}_{n \times n} \tag{2.17}$$

and $\boldsymbol{P}_0$ is defined as,

$$\boldsymbol{P}_0 = \begin{bmatrix} P_1(x_1, y_1) & P_2(x_1, y_1) & \cdots & P_m(x_1, y_1) \\ P_1(x_2, y_2) & P_2(x_2, y_2) & \cdots & P_m(x_2, y_2) \\ \vdots & \vdots & \ddots & \vdots \\ P_1(x_n, y_n) & P_2(x_n, y_n) & \cdots & P_m(x_n, y_n) \end{bmatrix}_{n \times m} . \tag{2.18}$$

Because the distance is directionless, $B_k(x_i, y_i) = B_i(x_k, y_k)$ meaning, matrix $\boldsymbol{B}_0$ is symmetric. Unique solution is obtained if the inverse of matrix $\boldsymbol{B}_0$ exists,

$$\begin{Bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{Bmatrix} = \boldsymbol{G}^{-1} \begin{Bmatrix} \boldsymbol{u}^e \\ 0 \end{Bmatrix}. \tag{2.19}$$

The interpolation is finally expressed as

$$u(\boldsymbol{x}_I) = [\boldsymbol{B}^T(\boldsymbol{x}_I)\boldsymbol{P}^T(\boldsymbol{x}_I)]\boldsymbol{G}^{-1} \begin{Bmatrix} \boldsymbol{u}^e \\ 0 \end{Bmatrix} = \boldsymbol{\varphi}(\boldsymbol{x}_I)\boldsymbol{u}^e \tag{2.20}$$

where the vector of interpolation functions $\boldsymbol{\varphi}(\boldsymbol{x})$ is defined by

$$\boldsymbol{\varphi}(\boldsymbol{x}_I) = [\varphi_1(\boldsymbol{x}_I), \varphi_2(\boldsymbol{x}_I), \ldots, \varphi_i(\boldsymbol{x}_I), \ldots, \varphi_n(\boldsymbol{x}_I)] \tag{2.21}$$

in which

$$\varphi_k(\boldsymbol{x}_I) = \sum_{i=1}^{n} B_i(\boldsymbol{x}_I)\bar{G}_{i,k} + \sum_{j=1}^{m} P_j(\boldsymbol{x}_I)\bar{G}_{n+j,k} \tag{2.22}$$

where $\bar{G}_{i,k}$ is the (i,k) element of matrix $\boldsymbol{G}^{-1}$ [5]. Once the inverse of matrix $\boldsymbol{G}$ is obtained, the derivatives of test functions are easily obtained as [5]

$$\begin{aligned} \frac{\partial \varphi_k}{\partial x} &= \sum_{i=1}^{n} \frac{\partial B_i}{\partial x}\bar{G}_{i,k} + \sum_{j=1}^{m} \frac{\partial P_j}{\partial x}\bar{G}_{n+j,k} \\ \frac{\partial \varphi_k}{\partial y} &= \sum_{i=1}^{n} \frac{\partial B_i}{\partial y}\bar{G}_{i,k} + \sum_{j=1}^{m} \frac{\partial P_j}{\partial y}\bar{G}_{n+j,k} \end{aligned} . \tag{2.23}$$

Properties of the test functions

The test functions $\varphi_i(\boldsymbol{x})$ depend uniquely on the distribution of scattered nodes after all basis function are determined [5]. The radial point interpolation functions have the following properties [5]:

1. Radial point interpolation functions are linearly independent in the influence domain.

2. Radial point interpolation functions have the Kronecker delta property,

$$\varphi_i(\boldsymbol{x} = \boldsymbol{x}_j) = \begin{cases} 1, & i = j,\ j = 1, 2, \ldots, n \\ 0, & i \neq j,\ i, j = 1, 2, \ldots, n \end{cases} \tag{2.24}$$

3. $\varphi_i(\boldsymbol{x})$ is of unity partition as

$$\sum_{i=1}^{n} \varphi_i(\boldsymbol{x}) = 1 \tag{2.25}$$

4. $\varphi_i(\boldsymbol{x})$ is of reproducing properties as

$$\sum_{i=1}^{n} \varphi_i(\boldsymbol{x}) x_i = \boldsymbol{x} \tag{2.26}$$

5. $\varphi_i(\boldsymbol{x})$ has simple derivatives.

6. Local compact support.

The radial basis function used in the NNRPIM is given by [1]

$$B_i(x,y) = (r_i^2 + c^2)^p. \tag{2.27}$$

Its partial derivatives are obtained as follows

$$\begin{aligned} \frac{\partial B_i}{\partial x} &= 2p(r_i^2 + c^2)^{p-1}(x - x_i) \\ \frac{\partial B_i}{\partial y} &= 2p(r_i^2 + c^2)^{p-1}(y - y_i) \end{aligned}. \tag{2.28}$$

where c and p are the RBF shape parameters [5]. The variation of these parameters can affect the performance of the RBFs [1]. Dinis et al. [1] showed the optimal values are

$$c \ll 1 \quad \text{and} \quad p \simeq 1, \tag{2.29}$$

the values used in this work are $c = 0.0001$ and $p = 0.9999$.

2.2 Linear elastic analysis using the NNRPIM

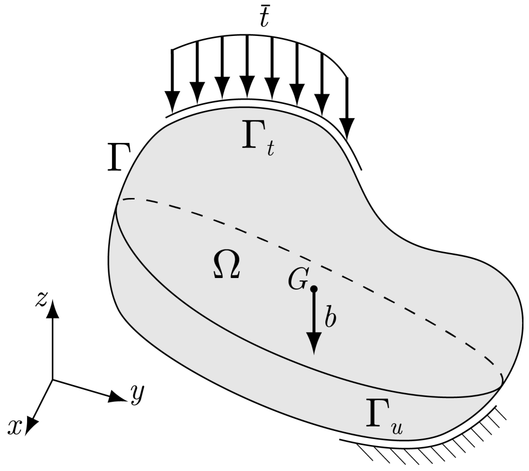


Figure 2.7: Representation of a solid with domain Ω and boundary Γ.

In this work, the NNRPIM is used to numerically solve elasto-static solid mechanics problems. Only small displacements are considered in this thesis. In practical terms, what is expected from the numerical method is to, starting from a domain discretisation and boundary conditions impositions, obtain the displacement, stress and strain fields. The basic solid mechanics problem consists in a body with domain Ω and boundary Γ that is subjected to various loads and conditions. The loads (i. e. the natural boundary conditions) are applied in the the natural boundary Γ_t. The prescribed displacements (i. e. the essential boundary conditions) are imposed in the essential boundary Γ_u. The body can also be subjected to a body force $\boldsymbol{b}$, distributed along the entire domain of the problem. In figure 2.7 there is a schematic representation of the general problem.

2.2.1 Galerkin weak form

In the NNRPIM, the discrete equation system is obtained using the Galerkin weak form [17]. Considering the problem described above, the Lagrangian functional is

$$L = T - U + W_f \tag{2.30}$$

where T is the kinetic energy, U is the strain energy and W_f is the work produced by external forces [17]. After substituting the corresponding explicit expressions and performing various mathematical operations in (2.30) [17], the "Galerkin weak form" is obtained

$$\mathscr{L} = \int_\Omega \delta\boldsymbol{\varepsilon}^T\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_\Omega \delta\boldsymbol{u}^T\boldsymbol{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \delta\boldsymbol{u}^T\bar{\boldsymbol{t}}\,\mathrm{d}\Gamma = 0. \tag{2.31}$$

2.2.2 Discrete system equations

In the NNRPIM, the weak form has local support, that means the discrete system of equations is developed firstly for every influence domain. Then, the local systems of equations are assembled to form the global system of equations, that afterwards is solved.

The NNRPIM trial function is given by,

$$\boldsymbol{u}(\boldsymbol{x}_I) = \sum_{i=1}^{n} \varphi_i(\boldsymbol{x}_I) u_i \tag{2.32}$$

where $\varphi_i(\boldsymbol{x}_I)$ is the NNRPIM interpolation function, u_i is the nodal displacement of the n nodes that belong to the influence domain of interest node $\boldsymbol{x}_I$.

Substituting the expression (2.32) in (2.31), yields

$$\begin{aligned} \mathscr{L} = \sum_{I}^{n}\sum_{J}^{n} \delta\boldsymbol{u}_I \underbrace{\int_\Omega \boldsymbol{B}_I^T\boldsymbol{c}\boldsymbol{B}_J\,\mathrm{d}\Omega}_{K_{IJ}}\,\boldsymbol{u}_J - \sum_{I}^{n} \delta\boldsymbol{u}_I^T \underbrace{\int_\Omega \varphi_I^T\boldsymbol{b}\,\boldsymbol{d}\Omega}_{f_I} \\ - \sum_{I}^{n} \delta\boldsymbol{u}_I^T \underbrace{\int_{\Gamma_t} \varphi_I^T\bar{\boldsymbol{t}}\,\mathrm{d}\Gamma}_{\bar{f}_I} = 0 \end{aligned} . \tag{2.33}$$

where the matrix $\boldsymbol{B}_I$ is given by

$$\boldsymbol{B}_I = \begin{bmatrix} \dfrac{\partial \varphi_I}{\partial x} & 0 & 0 & \dfrac{\partial \varphi_I}{\partial y} & 0 & \dfrac{\partial \varphi_I}{\partial z} \\ 0 & \dfrac{\partial \varphi_I}{\partial y} & 0 & \dfrac{\partial \varphi_I}{\partial x} & \dfrac{\partial \varphi_I}{\partial z} & 0 \\ 0 & 0 & \dfrac{\partial \varphi_I}{\partial z} & 0 & \dfrac{\partial \varphi_I}{\partial y} & \dfrac{\partial \varphi_I}{\partial x} \end{bmatrix}^T . \tag{2.34}$$

Equation (2.33) can be represented as

$$\mathscr{L} = \delta \boldsymbol{U}^T [\boldsymbol{K}\boldsymbol{U} - \boldsymbol{F}] = 0 \tag{2.35}$$

where $\boldsymbol{U} = \boldsymbol{u}$ and $\boldsymbol{F} = \boldsymbol{f} + \bar{\boldsymbol{f}}$. The above equation results in the linear system of equations, represented by

$$\boldsymbol{K}\boldsymbol{U} = \boldsymbol{F}. \tag{2.36}$$

The essential boundary conditions are directly imposed in the stiffness matrix, $\boldsymbol{K}$.

2.2.3 Linear-elastic algorithm

The NNRPIM numerical implementation is summarised in table 2.1. In the case of this work, the algorithm was implemented in MATLAB. All the pre and post processing was also done in MATLAB.

Table 2.1: Linear elastic algorithm [1].

1. Determine the natural neighbours of each node and construct the correspondent Voronoï cells.
2. Construct the integration points based on the Voronoï cells.
3. Set the influence-cells (first or second degree).
4. Loop over the integration points in order to integrate the Galerkin Weak form.
   (a) Determine the nodes that directly influence the specified integration point, based on the previously defined integration-cells.
   (b) Compute the shape functions and its derivatives for each integration point.
   (c) Evaluate stiffness and load at each integration point.
   (d) Assemble the contribution of the specified integration point in order to form the system of equations.
5. Apply the boundary conditions.
6. Solve the algebraic system to obtain the nodal displacement.
7. Evaluate the strain and stress at each integration point.

2.3 Solid mechanics fundamentals

When a load is applied to a solid or structure it creates, in the solid, a stress field. That is, the solid is under the effect of stress. The stress field, in turn, originates strains. Solid mechanics [37, 38] defines and quantifies stresses and strains and provides a relation between them.

Throughout this work it is assumed that the solids only have linear-elastic behaviour. By *linear* it is meant that the stresses relate linearly to the strains. By *elastic*, it is meant that, when a solid is deformed (due to the application of loads), it returns to the same undeformed shape when the loads are removed.

The standard solid mechanics problem is described in 2.2.

2.3.1 The state of stress at a point

The stress is the fundamental quantity of solid mechanics. The state of stress at a point is given by the stress tensor

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{bmatrix}, \tag{2.37}$$

the stresses can also be written in a vector form

$$\boldsymbol{\sigma} = \left\{ \sigma_{xx} \quad \sigma_{yy} \quad \sigma_{zz} \quad \tau_{xy} \quad \tau_{yz} \quad \tau_{zx} \right\}^T. \tag{2.38}$$

There are two types of stress, normal stress, denoted by the letter σ and shear stress, represented by τ.

2.3.2 Equations of equilibrium

In a solid subjected to an arbitrary system of loads, that can be volume and surface forces, the stress distribution must be compatible with the global equilibrium of the body and, at the same time, must ensure the equilibrium of every part of the system. Considering an infinitesimal element in a body, calculating its equations of equilibrium originates the equilibrium equations of elasticity, which are given by

$$\begin{aligned} \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + F_x = 0 \\ \frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + F_y = 0 \\ \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} + F_z = 0 \end{aligned} \tag{2.39}$$

or simply,

$$\nabla \boldsymbol{\sigma} + \boldsymbol{F} = 0 \tag{2.40}$$

where $\boldsymbol{\sigma}$ is the stress tensor, ∇ is the gradient and $\boldsymbol{F}$ the body force vector [38].

2.3.3 Definitions of displacement and strain

Strain is a geometric quantity which depends on the relative movement of two or three points in a body [38]. In the preceding sections, two types of stress were discussed: normal and shear stress. This same qualification is used for strains, where ε and γ represent normal and shear strains respectively.

In the small deformation theory, a normal strain, ε, is defined as the change in length of a line segment between two points divided by the original length of the line segment. A shearing strain, γ, is defined as the angular change between two line segments which were originally perpendicular.

The strains are related to the displacements by the following equations [38]

$$\begin{aligned} \varepsilon_{xx} &= \frac{\partial u}{\partial x} & \gamma_{xy} &= \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \\ \varepsilon_{yy} &= \frac{\partial v}{\partial y} & \gamma_{yz} &= \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \varepsilon_{zz} &= \frac{\partial w}{\partial z} & \gamma_{zx} &= \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \end{aligned} \tag{2.41}$$

where u, v, and w are the displacements in the x, y, and z directions, respectively. In the above equations

$$\gamma_{xy} = \gamma_{yx},\ \gamma_{yz} = \gamma_{zy} \text{ and } \gamma_{zx} = \gamma_{xz}. \tag{2.42}$$

Equations (2.41) are known as the cartesian strain displacement relationships of solid mechanics. The system of equations can be written in matrix form as follows

$$\boldsymbol{\varepsilon} = \boldsymbol{L}\boldsymbol{u} \tag{2.43}$$

where $\boldsymbol{L}$ and $\boldsymbol{u}$ are given by

$$\boldsymbol{L} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 & \frac{\partial}{\partial y} & 0 & \frac{\partial}{\partial z} \\ 0 & \frac{\partial}{\partial y} & 0 & \frac{\partial}{\partial x} & \frac{\partial}{\partial z} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix}^T \quad \text{and} \quad \boldsymbol{u} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}. \tag{2.44}$$

Just as the state of stress at a point is described by a tensor, the strain is also defined by a tensor

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{xx} & \gamma_{xy} & \gamma_{xz} \\ \gamma_{yx} & \varepsilon_{yy} & \gamma_{yz} \\ \gamma_{zx} & \gamma_{zy} & \varepsilon_{zz} \end{bmatrix} \tag{2.45}$$

and can be also written in a vector form

$$\boldsymbol{\varepsilon} = \left\{ \varepsilon_{xx} \quad \varepsilon_{yy} \quad \varepsilon_{zz} \quad \gamma_{xy} \quad \gamma_{yz} \quad \gamma_{zx} \right\}^T . \tag{2.46}$$

2.3.4 Constitutive relations

Unless otherwise stated, throughout this thesis, the material of the body is considered isotropic and homogenous, therefore its elastic properties are completely identified by the mutually independent constants E and ν, denoting Young's modulus and Poisson's ratio, respectively. The relation between stress and strain is given by the generalised Hooke's law

$$\boldsymbol{\sigma} = \boldsymbol{c}\boldsymbol{\varepsilon} \tag{2.47}$$

where $\mathbf{c}$ is the material matrix, it is a symmetric non-singular matrix, one can say that

$$\boldsymbol{\varepsilon} = \boldsymbol{s}\boldsymbol{\sigma} \tag{2.48}$$

where $\boldsymbol{c} = \boldsymbol{s}^{-1}$, $\boldsymbol{s}$ is given by

$$\boldsymbol{s} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} . \tag{2.49}$$

2.3.5 Two dimensional problems

If a body consists of two parallel planes a constant thickness apart and bounded by any closed surface, it is said to be a plane body. Associated with this type of body there is a class of elasticity problems that are prone to some simplifications in the analysis. For one, only two dimensions are necessary to define its geometry, also, all the variables are independent from the zz axis.

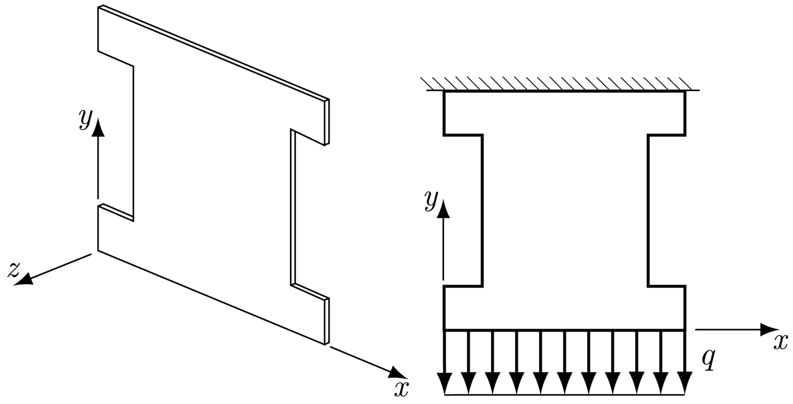
All the analysis in this work was made in regard to plane problems. From now on, all the mathematical treatment is made in only two dimensions. For the two-dimensional state of stress, the equilibrium equation of elasticity becomes

$$\begin{aligned} \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + F_x = 0 \\ \frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + F_y = 0 \end{aligned} . \tag{2.50}$$

The relation between strain and displacement is given by equation 2.43, in the two dimensional case, its matrices are

$$\boldsymbol{L} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial y} \\ 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix}^T \quad \text{and} \quad \boldsymbol{u} = \begin{bmatrix} u \\ v \end{bmatrix}. \tag{2.51}$$

Plane stress problems



(a) (b)

Figure 2.8: Plane stress configuration.

A plane stress condition can be used to analyse elasticity problems in which a body is subjected to in-plane forces and its thickness is much smaller than the other dimensions. In figure 2.8 it is shown a schematic representation of such problem, in which

$$\sigma_{zz} = \tau_{xz} = \tau_{yz} = 0 \tag{2.52}$$

thus, the stress and strain vectors are now

$$\boldsymbol{\sigma} = \begin{Bmatrix} \sigma_{xx} & \sigma_{yy} & \tau_{xy} \end{Bmatrix}^T \quad \text{and} \quad \boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_{xx} & \varepsilon_{yy} & \gamma_{xy} \end{Bmatrix}^T. \tag{2.53}$$

In the above representation, only the independent components of strain and stress are written. The constitutive matrix $\boldsymbol{c}$ becomes

$$\boldsymbol{c} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix}. \tag{2.54}$$

Plane strain problems

When a body has one of its dimensions much larger than the others, and the body's cross section does not vary throughout the direction of that

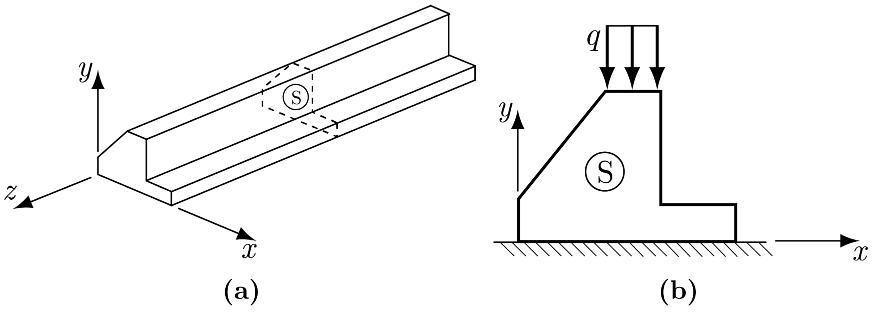


Figure 2.9: Plane strain configuration.

dimension. It can be said that the body is in plane strain. Assuming the applied loads are in the xy-plane, the resulting state of strain at such a section is two-dimensional, with

$$w = 0 \quad \text{and} \quad \frac{\partial}{\partial z} = 0 \tag{2.55}$$

which results in

$$\varepsilon_{zz} = \gamma_{xz} = \gamma_{yz} = 0. \tag{2.56}$$

The stresses and strains are now represented by equation (2.53). In this case, the material matrix $\boldsymbol{c}$ is reduced to

$$\boldsymbol{c} = \begin{bmatrix} \lambda + 2G & \lambda & 0 \\ \lambda & \lambda + 2G & 0 \\ 0 & 0 & G \end{bmatrix} \tag{2.57}$$

where λ is the Lamé coefficient and G is the modulus of rigidity. Those constants are related to E and ν by

$$G = \frac{E}{2(1+\nu)} \quad \text{and} \quad \lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}. \tag{2.58}$$

Chapter 3

Fracture mechanics using the NNRPIM

In this chapter the existing NNRPIM formulation is extended to fracture mechanics. The provided NNRPIM linear-elastic code has to be adapted to consider discontinuities in the domain. This requires the introduction of new concepts, one of the most important being the visibility between nodes.

Firstly, some relevant fracture mechanics fundamentals are presented. Along the state of art, the discontinuities main features in numerical methods are studied and the specific solutions encountered for the NNRPIM are shown. Finally, all the specifics of the application of the NNRPIM to fracture mechanics are demonstrated and explained.

3.1 Fracture mechanics fundamentals

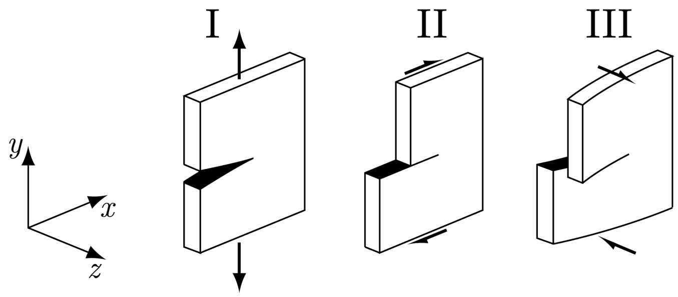


Figure 3.1: Fundamentals modes of deformation.

Fracture mechanics [39, 40] is the field of applied mechanics that deals with fractured bodies and structures, its main objective is to quantify and predict the behaviour of cracked structures under service conditions. It uses systematic mathematical rules to characterise cracks and their effects [22].

Ultimately, fracture mechanics predicts, if and when a given structure may become unsafe during its operational service life [22].

There are three basic modes of deformation of a cracked body. These modes are characterised by the movements of the upper and lower crack surfaces with respect to each other, they are shown if figure 3.1, each deformation mode has an associated stress intensity factor, K_I, K_{II} and K_{III} respectively. From these modes any crack deformation can be presented by an appropriate superposition [22]. In the specific case of this work, only mode I and mode II will be present because only two-dimensional problems are analysed.

3.1.1 The stress intensity factor

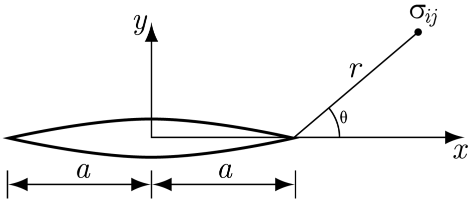


Figure 3.2: Stresses at a point ahead of a crack tip.

The fundamental postulate of linear elastic fracture mechanics (LEFM) is that the behaviour of a crack (i. e whether it grows or not and how fast it grows) is determined solely by the value of the stress intensity factor [22]. That implies, in order to be able to predict the behaviour of cracked solids, it is necessary to evaluate the stress intensity factor.

From linear elastic theory, Irwin [41] showed that the stresses in the vicinity of a crack tip take the form

$$\sigma_{ij} = \frac{K}{\sqrt{2\pi r}} f_{ij}(\theta) + \dots \tag{3.1}$$

where r, θ are the polar coordinates of a point with respect to the crack tip, see figure 3.2. K is a constant and is called the *Stress Intensity Factor* (SIF). Dimensional analysis shows that K must be linearly related to stress and directly related to the square root of e characteristic length [39]. The general form of the stress intensity factor is given by

$$K = \sigma\sqrt{\pi a} \cdot f\left(\frac{a}{W}\right) \tag{3.2}$$

where $f\left(\frac{a}{W}\right)$ is a dimensionless parameter that depends on the geometries of the specimen and crack [39].

3.1.2 Crack propagation direction

The simplest approach to predict the crack propagation direction, relies upon the premise that crack growth takes place in a direction perpendicular to the maximum principal stress at or near the crack tip [22]. That premise is called the maximum circumferential stress criterion [36].

The criterion for crack growth can be applied in several different ways, in the case of this work, the criterion is applied by:

1. Determining the stresses near the crack tip.
2. Calculating the principal stresses and its direction.
3. Propagating the crack in the direction perpendicular to the direction of the maximum principal stress.

There are other criterion for crack propagation direction, namely the strain energy density criterion [42] and the maximum energy release rate criterion [43]. Those can be implemented in the fracture mechanics algorithm developed in this work, but the maximum tangential stress criterion is the simplest and provides accurate results. For those reasons, it will be the only criterion used in the analysis made in this work.

3.2 Discontinuities in the domain

Cracks are discontinuities in the volume of a given continuous domain. In this work, cracks are modelled taking advantage of the natural neighbour concept.

Before going into the specifics of crack modelling with the NNRPIM, it is necessary to review the existing literature on the modelling of discontinuities and cracks with meshless methods. For that reason, in this section, firstly, a state of the art about cracks with meshless methods is made, then the concept of cracks with the NNRPIM is explained.

3.2.1 Visibility criterion for discontinuous approximations

The visibility criterion [44] is a simple way to introduce discontinuities in meshless approximations [2]. In this approach the boundaries of the body and any discontinuities are considered opaque when constructing the weight functions [2]. By opacity it is meant that, when the influence domain of a weight function is constructed, if there is a discontinuity between two nodes it is said that those do not see each other and therefore cannot be in the same influence domain [2]. The visibility criterion is schematically explained in figure 3.3.

As a consequence of the visibility criterion, a discontinuity is introduced in the weight and test functions wherever the influence domain is cut by a

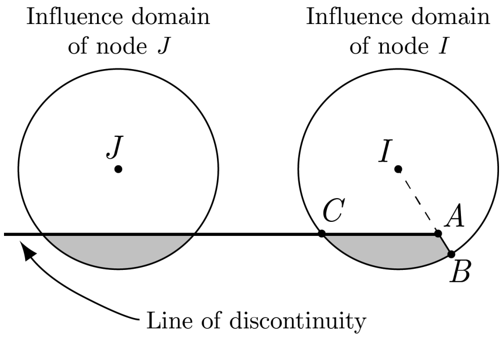


Figure 3.3: Influence domains near a discontinuity. The shaded areas are removed from the influence domain.

line of discontinuity. However, for a test function near the tip of the line of discontinuity, a byproduct of the visibility criterion is a discontinuity in the approximation within the influence domain [2]. As a consequence, the approximation will not be continuous within the domain, which is undesirable in a Galerkin method [2]. Although this discontinuities are not favourable they do not lead to failure in convergence [44].

3.2.2 Diffraction method for discontinuous functions

The diffraction method [45] is motivated by the way light diffracts around a sharp corner. It applies only to polar-type weight functions, defined as a function of a single parameter, s. Considering the line of discontinuity shown in figure 3.3, the essence of the diffraction method is to treat the line of discontinuity as opaque but to evaluate the length of the ray s by a path that passes around the corner of the discontinuity. The weight parameter is computed by

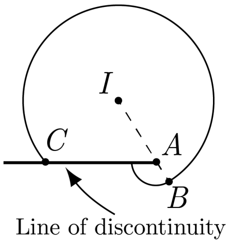


Figure 3.4: Influence domain of node I, considering the diffraction method.

$$s(\boldsymbol{x}) = \left(\frac{s_1 + s_2(\boldsymbol{x})}{s_0(\boldsymbol{x})} \right)^{\lambda} s_0(\boldsymbol{x}) \tag{3.3}$$

where

$$s_0(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{x}_I\|, \qquad s_1(\boldsymbol{x}) = \|\boldsymbol{x}_c - \boldsymbol{x}_I\|, \qquad s_2(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{x}_c\|. \tag{3.4}$$

The points $\boldsymbol{x}_I$ and $\boldsymbol{x}_c$ are represented in figure 3.5. As a result, the domain of influence, given by the criterion contracts on the far side of the discontinuity, as shown in figure 3.4. The diffraction method results in continuous weight and test functions within the influence domain but discontinuous across the line of discontinuity [45]. Using the diffraction method, the influence domain is defined by the following condition

$$w(s) > 0, \quad \boldsymbol{x} \in \Omega_I \tag{3.5}$$

where w is the weight function and Ω_I is the influence domain of node I.

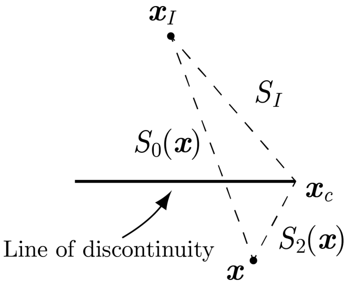


Figure 3.5: Scheme for the diffraction method for a node near the tip of the discontinuity.

3.2.3 Transparency method for discontinuous functions

In the transparency method [45], the weight function is smoothed at the tip of the discontinuity by varying the transparency as the tip is approached. At the tip, the line of discontinuity is considered to be completely transparent. When a ray from the evaluation point $\mathbf{x}$ to the node $\mathbf{x}_I$ intersects a line of discontinuity, the parameter s is given by

$$s(\boldsymbol{x}) = s_0(\boldsymbol{x}) + s_{max} \left(\frac{s_c(\boldsymbol{x})}{\bar{s}_c} \right)^{\lambda}, \quad \lambda \geq 2 \tag{3.6}$$

where $s_0(\boldsymbol{x})$ is defined in (3.5), s_{max} is the radius of nodal support, and $s_c(\boldsymbol{x})$ is the distance from the crack tip to the intersection point. The parameter $\bar{s}_c$ sets the intersection distance at which the crack segment is completely opaque, and is given by

$$\bar{s}_c = kh \tag{3.7}$$

where the parameter k is used to vary the opacity, and h is a measure of nodal spacing. As in the diffraction method, the influence domain is determined by the condition in (3.5).

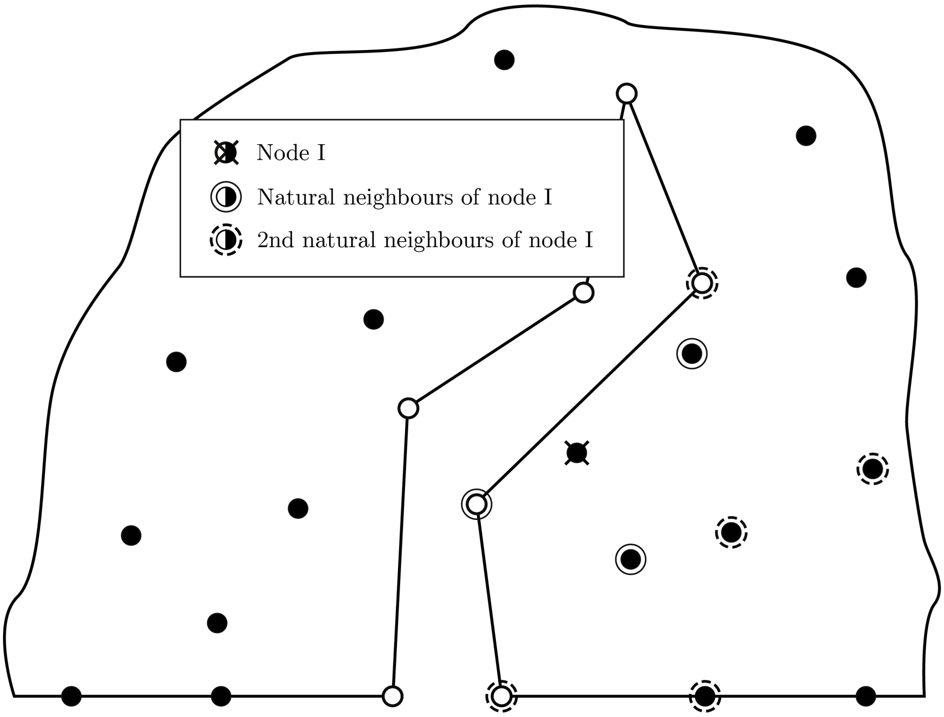

Node I
Natural neighbours of node I
2nd natural neighbours of node I

Figure 3.6: Natural neighbours and influence domains of a node near the crack.

3.2.4 Discontinuities in the NNRPIM

The concept of the natural neighbours greatly simplifies the inclusion of discontinuities in the domain of a given problem. Using the concept of the natural neighbour, a discontinuity is modelled by not allowing the nodes that represent each of the boundaries of the discontinuity to be natural neighbours of each other. This means that nodes on opposite sides of the discontinuity are never in the same influence domain and, therefore, there is no connectivity between them, as the natural neighbours of a given node represent the influence domain of the considered node.

The methods described above could be applied to the NNRPIM to model discontinuities. But that would mean not to take advantage of the natural neighbour concept, which provides a simpler way of dealing with cracks in meshless methods.

As with the diffraction and transparency methods, the treatment of the influence domains of the nodes in the middle of the line of discontinuity is different from the one in the influence domains of the nodes near the crack tip. The two situations are explained below.

Influence domains in the middle of the crack

In figure 3.6 the natural neighbours of a node near the crack are represented. It can be seen that the nodes of one side are not natural neighbours of nodes on the other side of the crack and therefore are not in the same influence domain, meaning that there is no connectivity between then.

Influence domains near the crack tip

In figure 3.7 it is shown the natural neighbours of the node at the crack tip. Analysing the figure it is possible to verify that nodes from different sides of the crack are able to see each other. This is somewhat similar to the transparency and diffraction methods where the line of discontinuity near its tip is not completely opaque. Similarly to the afore mentioned methods, smooth shape functions (inside the influence domain) are obtained, allowing the NNRPIM to retain the advantages of these methodologies.

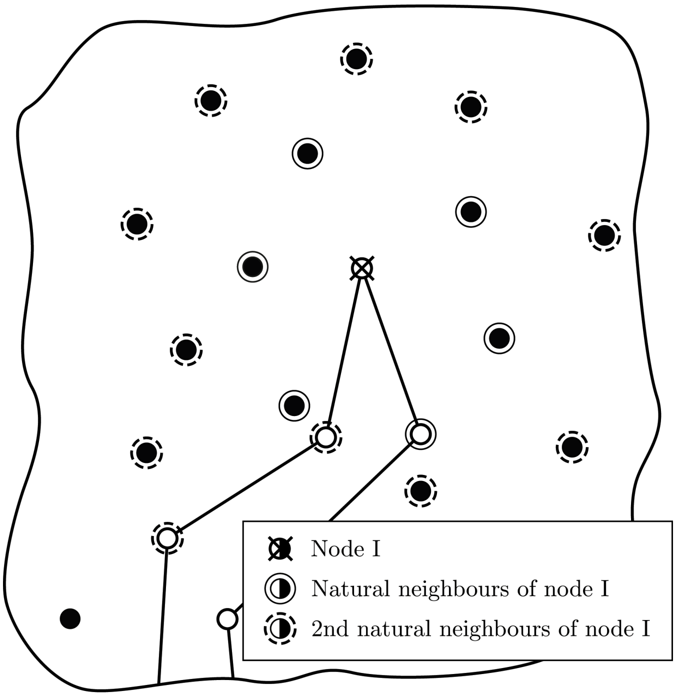

Node I
Natural neighbours of node I
2nd natural neighbours of node I

Figure 3.7: Natural neighbours and influence domains of a node at the crack tip.

Using second degree natural neighbours would be equivalent to increase the transparency of the crack tip. Analysing figure 3.7 it is possible to conclude that, as the size of the influence domain increases, more nodes from one side of the crack are able to see nodes on the other side. From a physical point of view, this means that the "real" crack tip is not in the same point as the crack tip node, but is somewhere behind in the middle of the crack. It is also important to notice that the "real" crack length is different when using first or second degree natural neighbours.

3.3 The crack propagation algorithm

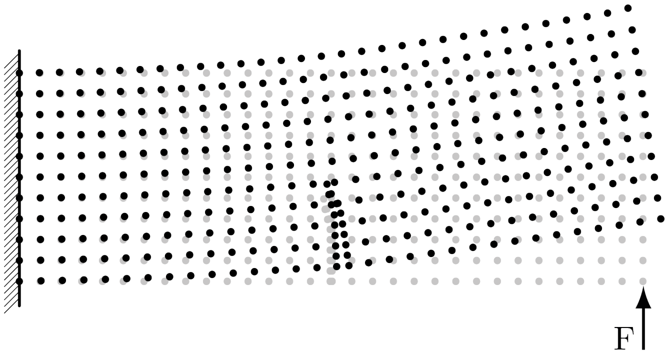


Figure 3.8: Example of a fracture mechanics problem, solved with the NNRPIM.

In order to prepare the NNRPIM code to solve fracture mechanics problems it is necessary to extend it to accommodate discontinuities in the problem's domain. Apart from that, it is necessary to create an automatic way of crack growth simulation. In figure 3.8 an illustration of a fracture mechanics problem solved with the NNRPIM is shown. The developed NNRPIM software solves fracture mechanics problems executing the sequence in table 3.1. Some of the steps are common to the provided NNRPIM code, the ones developed specifically to fracture mechanics are: 2, 4, 5, 6, 10a and 10b. These are explained in detail in this section.

In **step 1** all the necessary input data is specified. In **step 2** the problem is discretised in a nodal mesh and the nodes that are part of the crack, which are another boundary of the problem, are identified. In **step 3**, the integration mesh is generated, the influence cells and the natural neighbours are calculated. In **step 4**, the natural neighbours of the nodes that are in influence domains that are crossed by a line of discontinuity, are recalculated to accommodate it. Between the boundaries that represent the crack there is not material, therefore, in **step 5**, the integration points inside the boundaries of the crack are eliminated. In **step 6**, the second degree neighbours are calculated. With all the necessary data, in **step 7**, a set of standard NNRPIM operations is performed. In **step 8** the boundary conditions of the problem are applied. In **step 9** the discrete system of equations is solved. In **step 10** the stresses and strains are calculated. If the crack has reached its final length, the output data is presented in **step 11**. Else-wise,

Table 3.1: Crack propagation algorithm. In bold are the steps developed for fracture mechanics.

1. Read all the input variables.
2. **Generate the nodal mesh.**
3. Generate the integration mesh and the natural neighbours.
4. **Recalculate the influence domains that are crossed by discontinuities.**
5. **Eliminate the integration points situated in the middle of the crack.**
6. **Calculate the second degree natural neighbours.**
7. Calculate:
   (a) The influence domains.
   (b) The interpolation functions.
   (c) Stiffness matrix.
8. Apply the boundary conditions.
9. Calculate the displacement vector.
10. Calculate the stress and strain at every integration point.
    - If the crack has not reached its final length:
      (a) Calculate the crack propagation direction.
      (b) Extend the crack, with a prescribed length.
      (c) Return to step 3.
    - If the crack has reached its final length: go to step 11.
11. Output variables: crack path, displacement field and stress and strain fields.

the previously determined stresses are used to calculate the crack propagation direction, the nodal mesh is then updated to accommodate the crack's extension and the program returns to step 3.

3.3.1 Nodal mesh generation and update (Step 3 and 10b)

The nodal mesh is generated in the same manner as in the provided NNRPIM code, with the exception of the generation of the nodes that represent the crack, which are identified as "crack nodes". The nodal mesh is generated in the following sequence:

1. Generate the nodes minus the nodes of the initial crack.
2. Generate the nodes of the initial crack. Identify the crack tip(s) and the remaining crack nodes.

The nodal mesh generation process consists in creating a set of points in specified coordinates. This process is much simpler than in FEM because there is not any geometry related restriction, or any requirement concerning

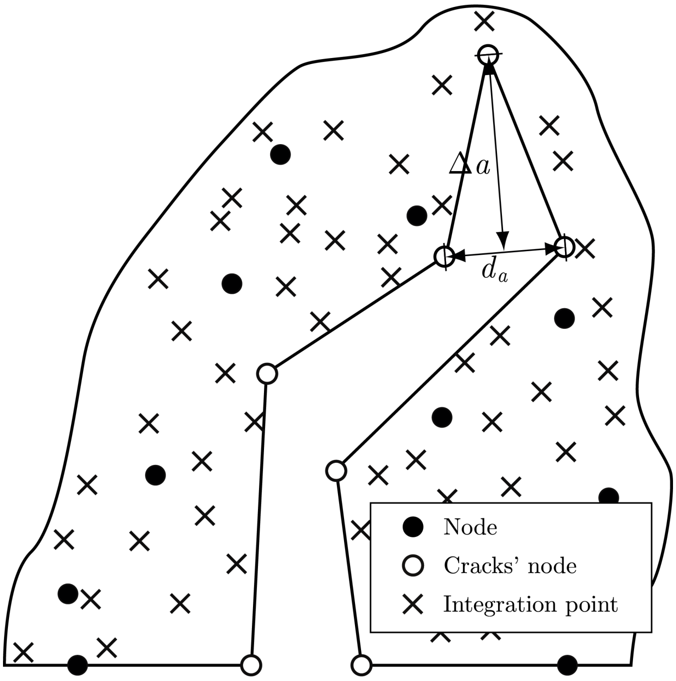


Figure 3.9: Example of a nodal mesh with crack. Δa is the distance in which the crack is extended at each iteration and d_a is the distance between the two sides of the crack.

the nodal connectivity. The nodal mesh is the set of all nodes which is represented by

$$\mathbf{N} = \{n_1, n_2, \ldots, n_N\} \in \mathbb{R}^2. \tag{3.8}$$

The nodes that define the crack are generated in groups of two, one for each side of the crack. That procedure represents a standardised way of generating the nodes and facilitates the process of defining which side the nodes belong. This process is used in several meshless methods [33, 46, 47]. In figure 3.9 it is shown how a crack is modelled, it can be seen that the nodes of the crack are in groups of two, in opposite sides in relation to the crack. The set of nodes that belong to the crack is given by

$$\mathbf{N}_k \subset \mathbf{N} \tag{3.9}$$

Nodal mesh update (Step 10b)

The nodal mesh is updated at each iteration of the program to take the extension of the crack into account. This process takes the following steps:

1. Eliminate the nodes whose presence would degrade the quality of the results, due to the fact of being too close to the crack tip. All the

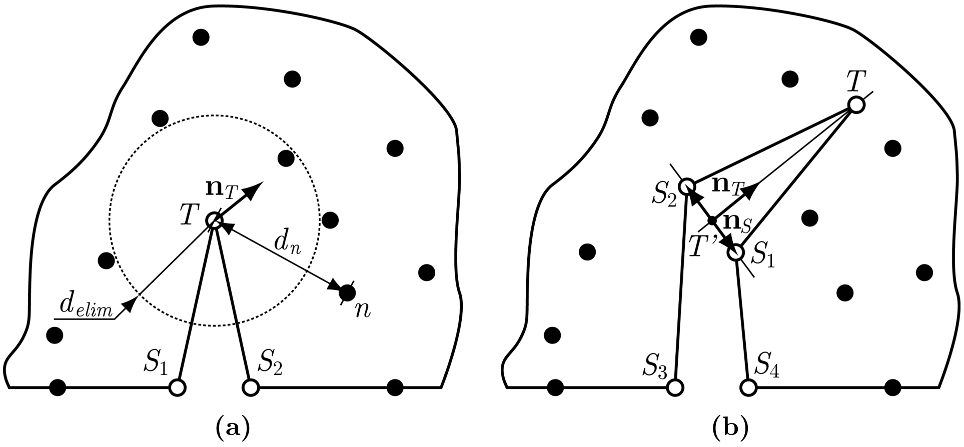


Figure 3.10: Representation of a mesh update: (a) initial mesh; (b) updated mesh.

nodes that verify the following condition are eliminated

$$d_n \leq d_{elim} \tag{3.10}$$

where d_n is the distance between the crack tip and the other nodes and $d_{elim} > \Delta a \wedge d_{elim} < 1.5 \cdot \Delta a$, figure 3.10a.

2. Create the new nodes of the crack. The new crack tip is created making

$$\boldsymbol{x}_T = \boldsymbol{x}_{T'} + \Delta a \cdot \boldsymbol{n}_T \tag{3.11}$$

where T' is the old crack tip and $\boldsymbol{n}_T$ is the crack propagation direction. The old crack tip becomes two additional crack nodes

$$\boldsymbol{x}_{S_n} = \boldsymbol{x}_{T'} + \alpha \cdot \boldsymbol{n}_S \tag{3.12}$$

where $\boldsymbol{n}_S$ is a unitary vector perpendicular to $\boldsymbol{n}_T$ and $\alpha = (-1)^n \cdot d_a/2$.

3. Update all the matrices that contain information about the crack tip and the crack's nodes.

3.3.2 Calculation of the crack's propagation direction (step 10a)

The crack's propagation direction depends on the stress field at the crack tip. As referred in the fracture mechanics fundamentals, the crack propagates in a direction perpendicular to the first principal stress direction, it is determined as follows:

1. Select the integration points to be used in the calculation, according to the following condition

$$d_n \leq r_s, \tag{3.13}$$

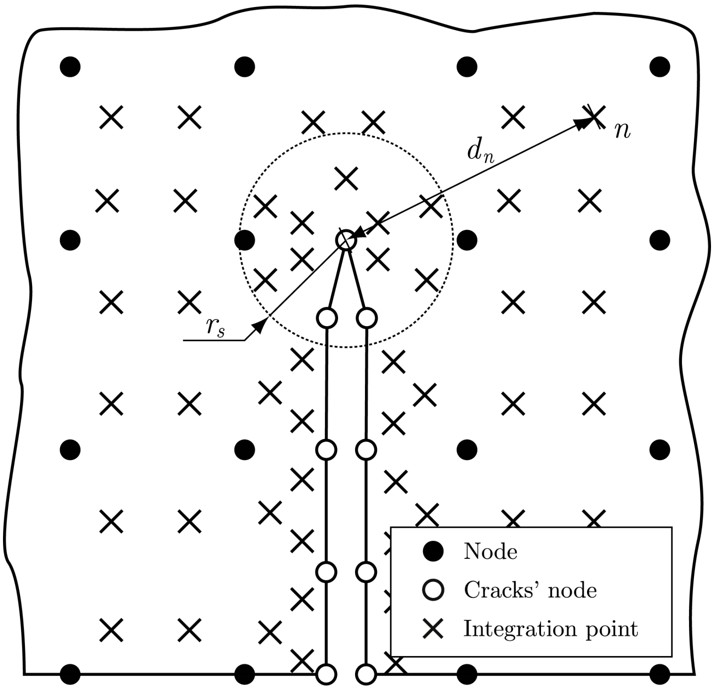


Figure 3.11: Representation of the process used to select the integration points.

where d_n is the distance from the crack tip to the integration point n and r_s is the prescribed radius, figure 3.11.

2. Calculate the weighted average of the stresses in the integration points selected in the previous step.

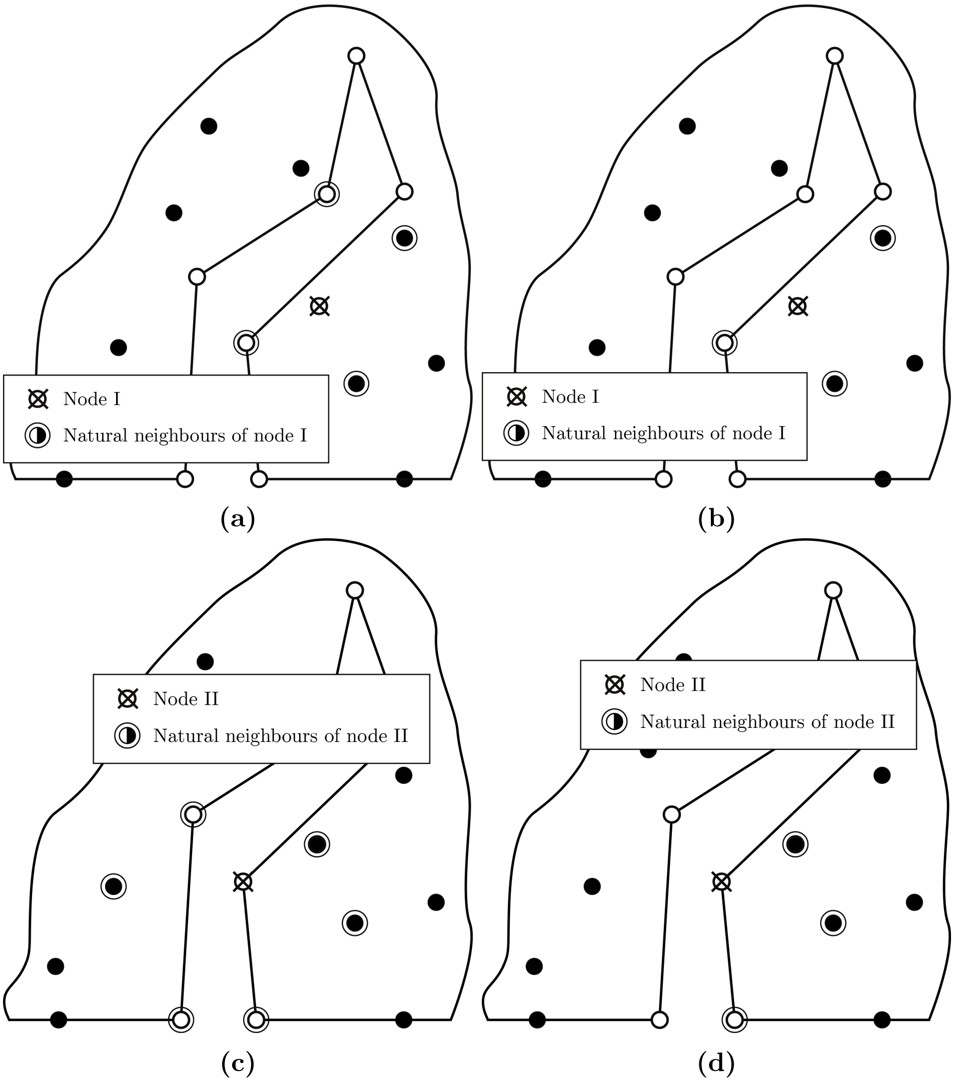
3. The result of the previous step is considered to be the stress tensor at the crack tip. Using that value, calculate the principal stresses at the crack tip.

4. The vector $\boldsymbol{n_T}$ is obtained and is then employed in the nodal mesh update.

In figure 3.11 there is a representation of how the integration points are selected. The radius, r_s, is a parameter that has to be carefully examined and has a significant influence in the crack's propagation direction. Its influence in the results is studied in chapter 4.

3.3.3 Computation of the crack's nodes natural neighbours (step 4)

The natural neighbours are calculated by the provided NNRPIM code. Discontinuities in the middle of the domain are not taken into consideration. To take the cracks into account, it is necessary to eliminate certain natural neighbours, from some nodes that are not supposed to be seen or see the original natural neighbours. Analysing figure 3.12 it is possible to to intuitively understand the process. On the right side are shown the natural neighbours calculated by the provided code of the NNRPIM, on the left are displayed the fracture mechanics ready natural neighbours.



Node I
Natural neighbours of node I
(a)
Node I
Natural neighbours of node I
(b)
Node II
Natural neighbours of node II
(c)
Node II
Natural neighbours of node II
(d)

Figure 3.12: Recalculation of the crack nodes natural neighbours: (a) and (c) initial NNRPIM natural neighbours; (b) and (d) fracture ready natural neighbours.

The following steps are used to recalculate the natural neighbours:

1. Separate the crack's nodes in two different sets, one for each side of

the crack. The two sets are

$$\mathbf{N}_{ks1} \subset \mathbf{N}_k \quad \text{and} \quad \mathbf{N}_{ks2} \subset \mathbf{N}_k. \tag{3.14}$$

There is no previous information as to each side a given node belongs, it is necessary to distribute the nodes in the correct set. That is made by passing a line, r, through a node for which the side is known and through a node that is immediately next in the crack and for which the side is unknown. Having the data of the line, the following condition is tested

$$y_{S_n} > x_{S_n} \cdot m_r + b_r \tag{3.15}$$

where x_{S_n} and y_{S_n} are the coordinates of the nodes being tested, in the case of the example in figure 3.13 those would be S_3, S_4, S_5 and S_6. m_r and b_r are the parameters of the line r. Depending of the number of nodes that verify the condition, the nodes being tested are assigned to one or other margin:

- If zero or two nodes verify the condition: the node through which the line r passes is assigned to the same side as the node that is the other point that defines r.
- If only one node verifies the condition: the node through which the line r passes is assigned to the opposite side as the node that is the other point that defines r.

It should be noted that expression (3.15) is only valid for non-vertical lines, in case r is vertical the expression is different. Also, in the initial iteration, that in the example of figure 3.13 would correspond to the nodes S_1 and S_2, the two nodes are assigned to $\mathbf{N}_{ks1}$ or $\mathbf{N}_{ks2}$ arbitrarily, because only the relative position of the nodes is relevant.

2. The nodes that belong to $\mathbf{N}_{ks1}$ cannot be natural neighbours of the nodes that belong to $\mathbf{N}_{ks2}$ and vice-versa. The sets of the natural neighbours of the affected nodes are updated.

3. The natural neighbours of the nodes that belong to the crack are correctly defined. However, there may be nodes that are not part of the crack whose natural neighbours are not correctly defined. In this step the natural neighbours of additional nodes are verified. In figure 3.14 a general view of the procedure described in this step is shown. This procedure is performed in each node of the crack with the exception of the crack tip(s). In order to know if there are additional natural neighbours that need to be eliminated, two auxiliary lines are created, r_1 and r_2. These lines are used in the following inequality

$$y_{N_n^I} < x_{N_n^I} \cdot m_{r_1} + b_{r_1} \ \cup \ y_{N_n^I} > x_{N_n^I} \cdot m_{r_2} + b_{r_2} \tag{3.16}$$

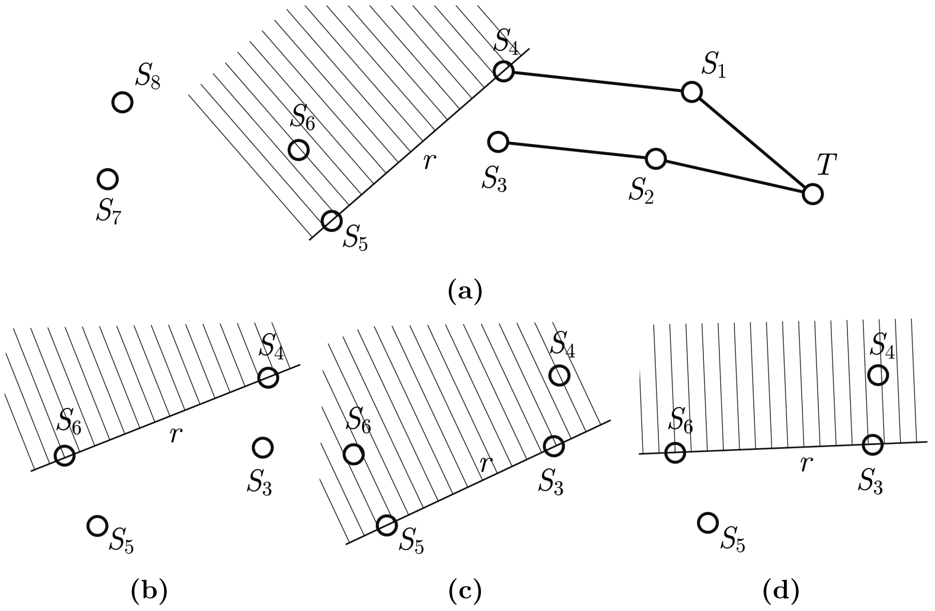


Figure 3.13: Method of assignment of the cracks' nodes to their respective sides: (a) general view in the middle of the process; (b) (c) and (d) remaining cases for the position of the line r.

where $x_{N_n^I}$ and $y_{N_n^I}$ are the coordinates of the natural neighbours of the node being verified. Analysing the example in figure 3.14, there are two nodes that are not supposed to be natural neighbours of node I, nodes N_1 and N_2. They verify the inequality (3.16) so they cease to be natural neighbours of node I. Also, node I cannot be a natural neighbour of the nodes N_1 and N_2, in this step this is also taken into account.

It is important to notice that, at this point, only the first degree natural neighbours are used. If, at this point in the program, the second degree natural neighbours were used, the algorithm would have to be much more complex because: nodes that would not be part of the crack and belong to one side (in relation to the crack) would be natural neighbours of nodes that also would not belong to the crack but are be on the other side of the domain (in relation to the crack). Using only the first degree natural neighbours this situation does not occur.

3.3.4 Elimination of the integration points in the interior of the crack (step 5)

The space between the faces of the crack is not part of the volume of the solid that is being analysed. Therefore, the area in the interior of the crack cannot be considered part of the problem's domain, hence the integration

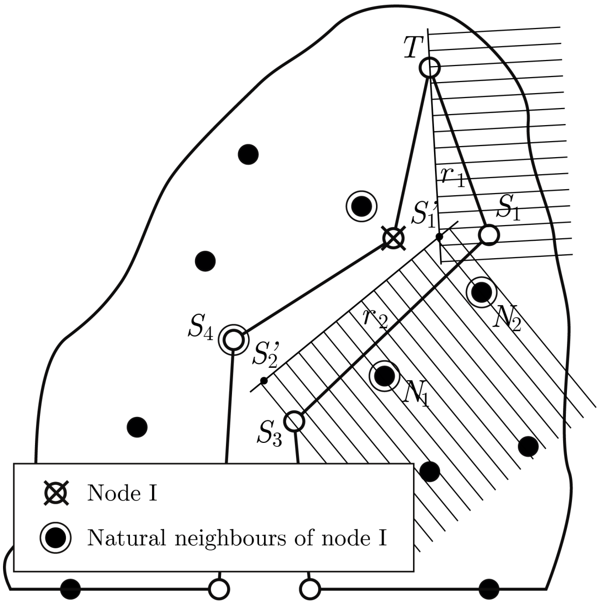


Figure 3.14: Natural neighbours of node I, after step 2. N_1 and N_2 are not supposed to be natural neighbours of node S_1' but, before step 3, still are. After that step they cease to be.

points that were generated by the NNRPIM in the interior of the crack must be eliminated.

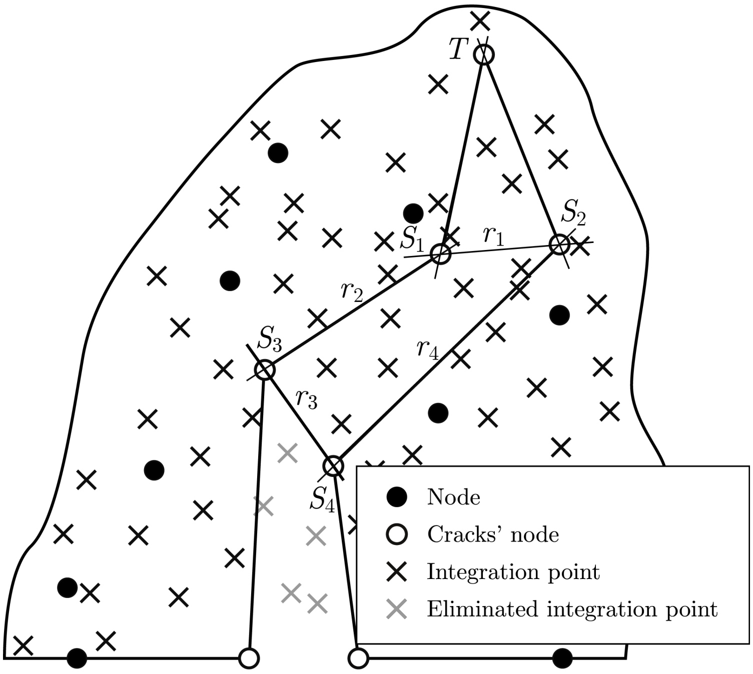


Figure 3.15: Example of a nodal mesh with crack.

The integration points are eliminated in a sequential matter. At each step, the following operations are performed, going from one end of the crack to the other:

1. Select four consecutive nodes of the crack. In the example of figure 3.15 those would be S_4, S_3, S_1 and S_2.
2. Define the set of integration points to be eliminated, which is bounded by lines r_1, r_2, r_3 and r_4. These lines are calculated in this step, the points have to be selected in a manner such that the straights do not intersect each other. In the case of the example in figure 3.15, the set of integration points to be eliminated is given by

   $$\bigcap_{j=1}^{4} y_{q_I} < x_{q_I} \cdot m_{r_j} + b_{r_j} \tag{3.17}$$

   where x_{q_I} and y_{q_I} are the coordinates of the integration points and m_{r_j} and b_{r_j} are the parameters of the lines. The set of integration points is given by

   $$\mathbf{Q} = \{q_1, q_2, \ldots, q_I\} \in \mathbb{R}^2. \tag{3.18}$$

3. Eliminate the integration points that belong to the set defined by the inequality 3.17.

It is important to notice that, near the crack tip, the set of integration points to be eliminated has the shape of a triangle, contrary to what happens in the other cases, where the set has a trapezoidal shape. In the first case, the inequality 3.17 goes from four lines to just three. Also, if any of the lines of inequality (3.17) are vertical, the expressions are different.

3.3.5 Calculation of the second degree natural neighbours (step 6)

The second degree natural neighbours must be calculated after the discontinuities in the domain have been considered. The second degree natural neighbours are obtained adding, to the group of the natural neighbours of a given node, the groups of the natural neighbours of the nodes that are natural neighbours of said node. That is given by

$$\mathbf{N}_{2N_N} = \mathbf{N}_{N_N} \bigcup_{j=1}^{V} \mathbf{N}_{N_j} \tag{3.19}$$

where $\mathbf{N}_{2N_N}$ is the set of the second degree natural neighbours of node N, $\mathbf{N}_{N_N}$ are the first degree natural neighbours of the same node, $\mathbf{N}_{N_j}$ are the sets of the elements of $\mathbf{N}_{N_N}$, V is the number of natural neighbours of node N. In figure 3.16 there is an example of first and second degree natural neighbours.

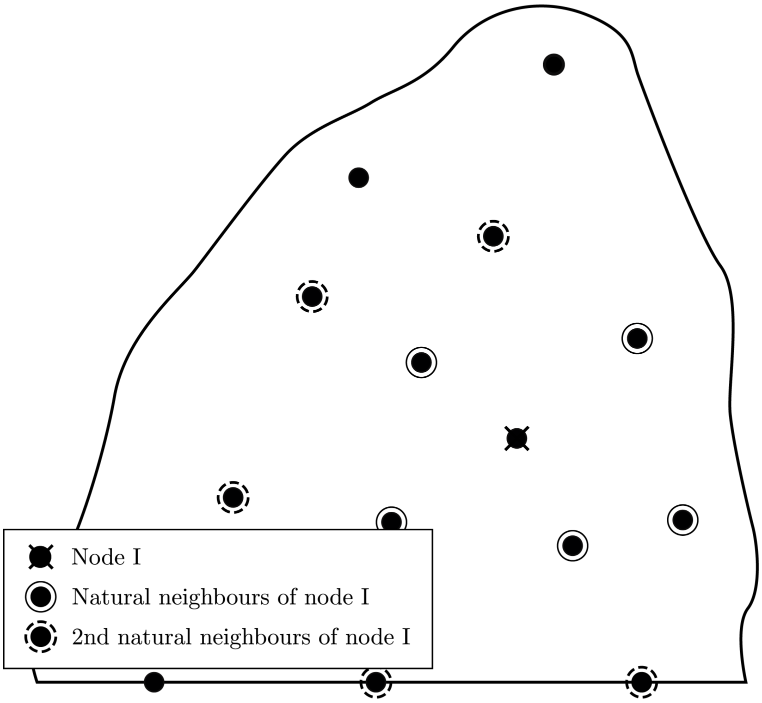

Node I
Natural neighbours of node I
2nd natural neighbours of node I

Figure 3.16: Example of second degree natural neighbours in the middle of the domain.

Chapter 4

Numerical examples and discussion

In this chapter, all the formulations presented in this thesis are applied in numerical examples. Firstly, the provided NNRPIM code is tested. It is necessary to guarantee that the method generates accurate results.

To benchmark all the analysis made in the previous chapters, the fracture mechanics algorithm is applied to practical examples. The obtained results are compared with existing ones from the literature. All the mechanisms presented in 3 are backed up with numerical results.

4.1 Solid mechanics using the NNRPIM

In this section, solid mechanics numerical examples are presented. The following problems are studied:

1. Cantilever beam.

2. Infinite plate with a central crack.

3. Infinite plate with a central hole.

A convergence analysis is made for all examples. In this section a plane stress condition is assumed.

4.1.1 Example 1

In this example, the vertical displacement at the extremity of a cantilever beam is calculated. The accuracy of the NNRPIM in calculating the displacement and stress field is verified. The problem is represented in figure 4.1, and in table 4.1 all the relevant data is displayed.

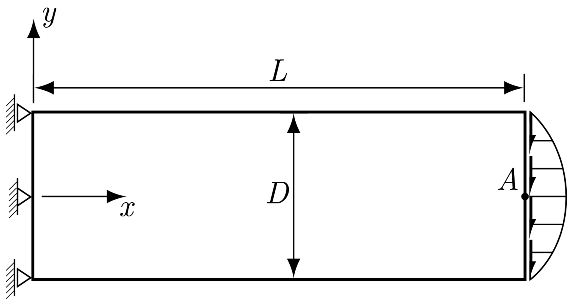


Figure 4.1: Representation of the problem of example 1.

The solution of obtained with the NNRPIM is compared with the analytical solution [37], given by

$$
\begin{aligned}
u &= -\frac{Py}{6EI}\left[(2L-x)3x+(2+v)\left(y^2-\frac{D^2}{4}\right)\right], \\
v &= \frac{P}{6EI}\left[x^2(3L-x)+3v(L-x)y^2+\frac{4+5v}{4}D^2x\right]
\end{aligned}
\tag{4.1}
$$

where I is the moment of inertia $I = D^3/12$.

Table 4.1: Relevant data regarding example 1.

Relevant data (example 1)		
P	1	[Pa]
E	1000	[kPa]
ν	0.3	
Thickness	1	[m]
L	8	[m]
D	2	[m]

In figure 4.2 the relation between the number of nodes used in the analysis, and the relative error of the vertical displacement of point A, is displayed. It is possible to verify that the result converges to a low relative error. The convergence of the results is verified for regular and irregular meshes.

In figure 4.3 the effective stress along the cantilever beam is displayed. It is possible to conclude that the NNRPIM is capable of obtaining a smooth stress field. That result was obtained using a 40 × 20 nodal mesh.

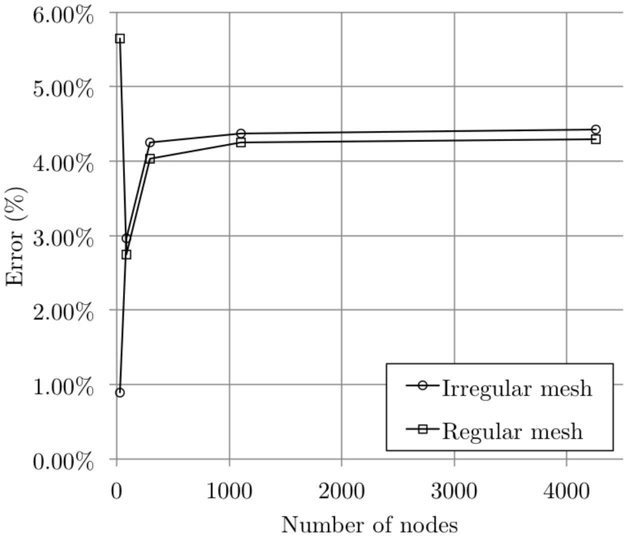


Figure 4.2: Relative error of the vertical displacement of point A.

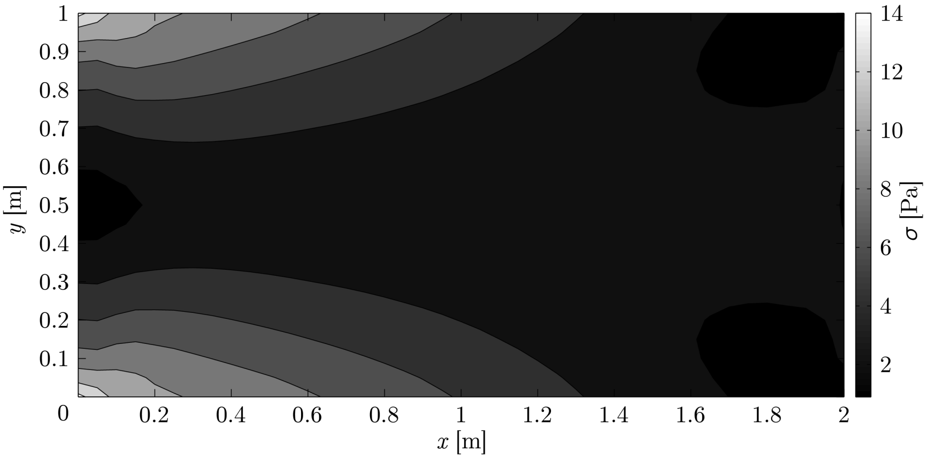


Figure 4.3: Effective stress in the cantilever beam.

4.1.2 Example 2

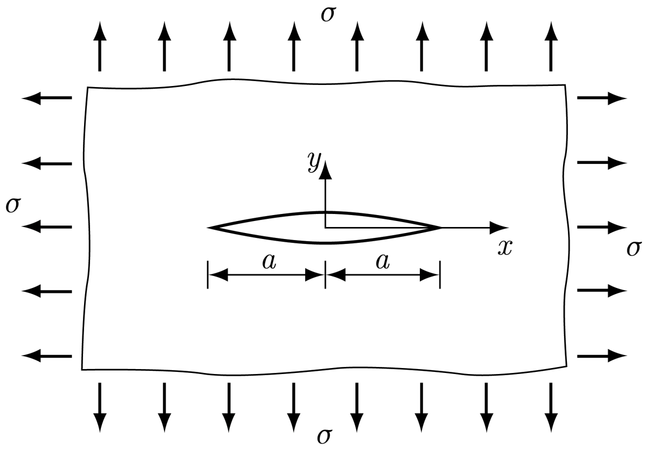


Figure 4.4: Schematic representation of the problem of example 2.

In this example, an infinite plate with a central crack is studied. Even though this is a fracture mechanics problem, geometric simplifications allow for the provided NNRPIM code to solve it, with only the solid mechanics formulation. The problem is represented in figure 4.4. It is possible to model it using only one quarter of the plate, in figure 4.5 there is a representation of how the problem is modelled in this example. Additional relevant data is shown in table 4.2. In order to verify the convergence of the results obtained with the NNRPIM, the numerical solution for the mode I SIF, K_I, is compared with the analytical solution [39], given by

$$K_I = \sigma\sqrt{\pi a}. \tag{4.2}$$

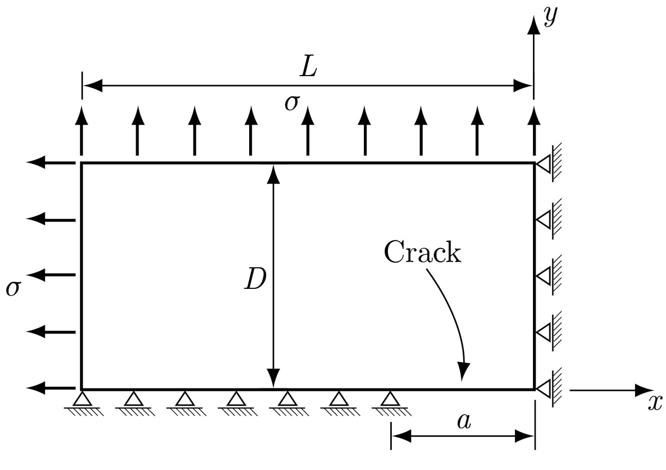


Figure 4.5: Representation of the problem of example 2.

As it is impossible to model an infinite plate, the length of the domain of the discretisation was chosen to be significantly larger than the crack length.

Table 4.2: Relevant data regarding example 2.

Relevant data (example 2)		
σ	20	[Pa]
E	1000	[Pa]
ν	0.25	
Thickness	1	[m]
L	2	[m]
D	1	[m]
a	0.25	[m]

Analysing figure 4.6 it is possible to verify that the solutions converges. Using more than 2000 nodes, the result is similar to the analytical result. The results were obtained using second degree influence domains.

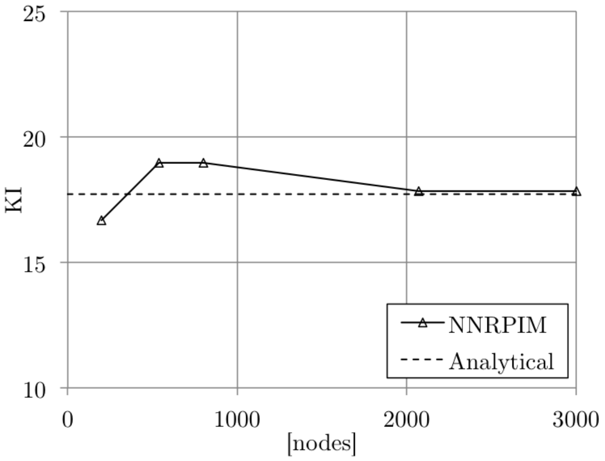


Figure 4.6: Relation of K_I with the number of nodes used in the analysis.

4.1.3 Example 3

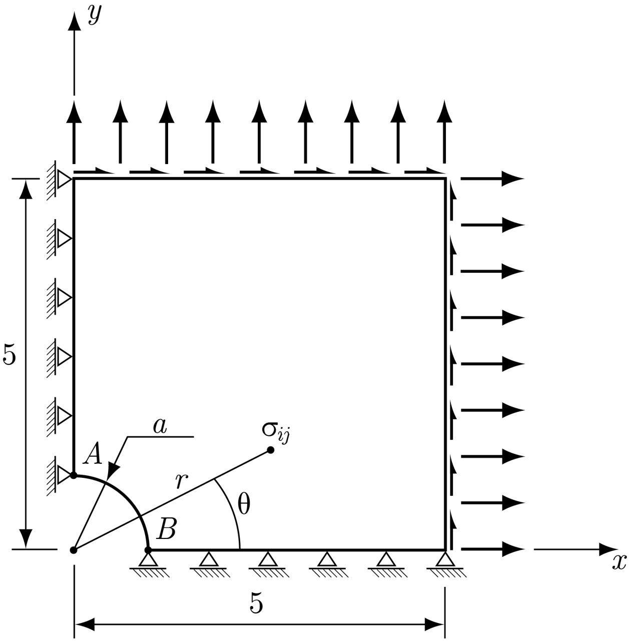


Figure 4.7: Representation of the problem of example 2.

In this example, an infinite plate with a central crack is studied. The problem is represented in figure 4.7. Due to the symmetry only a quarter of the plate was analysed. In this problem, the accuracy of the NNRPIM in analysing a problem with circular holes is tested. In table 4.3 additional data is displayed.

Table 4.3: Relevant data regarding example 3.

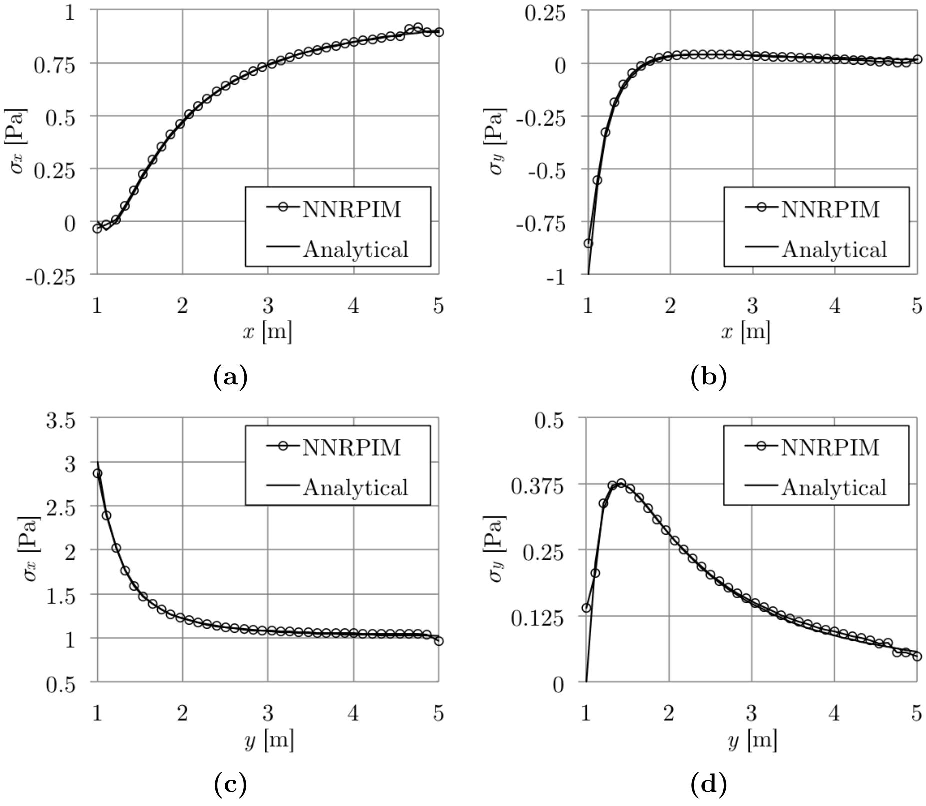
Relevant data (example 3)		
E	1000	[Pa]
ν	0.3	
Thickness	1	[m]
a	1	[m]

In section 4.2 some of the problems contain holes, so it is important to verify the behaviour of the NNRPIM when holes are involved. In order to benchmark the NNRPIM, the obtained results are compared with the analytical solution [3], given by

$$
\begin{aligned}
\sigma_x &= 1 - \frac{a^2}{r^2}\left(\frac{3}{2}\cos 2\theta + \cos 4\theta\right) + \frac{3a^4}{2r^4}\cos 4\theta \\
\sigma_y &= -\frac{a^2}{r^2}\left(\frac{1}{2}\cos 2\theta - \cos 4\theta\right) - \frac{3a^4}{2r^4}\cos 4\theta \\
\tau_{xy} &= -\frac{a^2}{r^2}\left(\frac{1}{2}\sin 2\theta + \sin 4\theta\right) + \frac{3a^4}{2r^4}\sin 4\theta.
\end{aligned}
\tag{4.3}
$$

The loads represented in figure 4.7 are also given by equations (4.3).

In figure 4.8 the stresses along the axis of the problem are displayed, those results were obtained with a nodal mesh with 2225 nodes and second degree influence domains. It is possible to verify that the results obtained with the NNRPIM are similar to the analytical solution. The NNRPIM originates smooth curves in all cases, only near the extremities of the domain the values differ from the analytical result.



(a) **(b)**

(c) **(d)**

Figure 4.8: Stresses along the axis of the problem of example 3: (a) σ_x for $y = 0$; (b) σ_y for $y = 0$; (c) σ_x for $x = 0$; (d) σ_y for $x = 0$.

In figure 4.9 the relation between the stresses in point A and B and the nodal discretisation is displayed. Analysing the figure is possible to verify that the solution converges to the analytical solution.

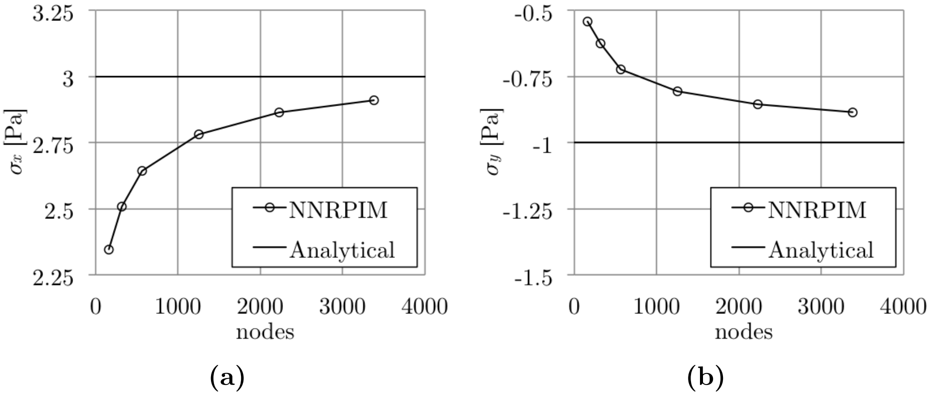


Figure 4.9: Evolution of stress in two critical points: (a) σ_x in point A; (b) σ_y in point B.

4.2 Fracture mechanics using the NNRPIM

In this section, fracture mechanics numerical examples are presented. The influence of the following parameters, in the crack path prediction, is studied in detail:

- The crack increment, Δa.
- The radius of selection of integration points, from which the stress vector is used to calculate the crack's propagation direction, r_s.
- The type of influence domain.

The variation of these parameters significantly influences the results. It is expected to determine a range of optimal values that is commom for all numerical examples. To facilitate the comparison between examples, Δa and r_s are expressed as a function of a nodal spacing parameter, h, which is given by

$$h = \frac{L}{divl} \tag{4.4}$$

where L is a dimension of the problem and *divl* is the number in which that dimension is divided, in the discretisation of that problem. The nodal discretisations used in this work have similar nodal spacings vertically and horizontally.

The presented numerical examples are:

1. Infinite plate with central crack.
2. Rectangular plate with an edge crack.
3. Three point bending beam.
4. Four point bending of a beam with a circular hole.
5. Three point bending of a beam with three circular holes – case I.
6. Three point bending of a beam with three circular holes – case II.
7. Plate with a central inclined crack.

The aforementioned examples are classical fracture mechanics problems and are typically used when benchmarking numerical methods. All of them present different challenges to the crack propagation algorithm.

In all the numerical examples a plane strain condition is assumed.

4.2.1 Example 1

The first example is the infinite plate with a central crack. The accuracy of the NNRPIM is evaluated in calculating the stresses near the crack tip (in the form of the SIF) and in predicting the crack path, for regular and irregular nodal meshes. In figure 4.4 there is a representation of the problem and on table 4.4 all the relevant data is shown.

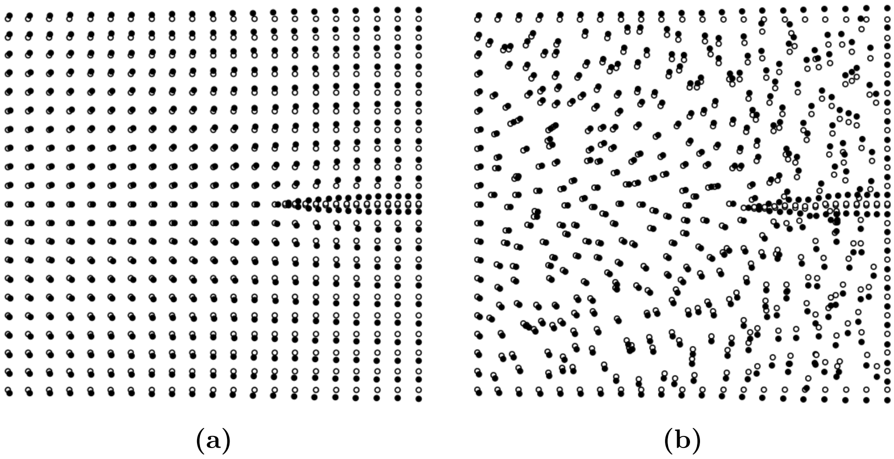


Figure 4.10: Discretisation of the problem: (a) uniformly spaced nodes; (b) irregularly distributed nodes.

This problem was modelled by discretising only half the plate and imposing $u = 0$ in $y = 0$, which would be the left edge in figure 4.10. It is not possible to simulate an infinite plate, for that reason, the length and height of the plate were chosen to be significantly larger than the length of the crack. That way it is possible to consider the plate, infinite.

Table 4.4: Relevant data regarding example 1.

Relevant data (example 1)		
σ	20	[Pa]
E	1000	[Pa]
ν	0.25	
Thickness	1	[m]
L	2	[m]
D	2	[m]
a	0.25	[m]
r_s	0.2	[m]

Stress analysis

The results obtained numerically were compared with the available analytical solution [39]. The stresses near the crack tip are

$$\begin{aligned}\sigma_x &= \frac{K_I}{\sqrt{2\pi r}}\cos\frac{\theta}{2}(1-\sin\frac{\theta}{2}\sin\frac{3\theta}{2})\\ \sigma_y &= \frac{K_I}{\sqrt{2\pi r}}\cos\frac{\theta}{2}(1+\sin\frac{\theta}{2}\sin\frac{3\theta}{2}),\\ \tau_{xy} &= \frac{K_I}{\sqrt{2\pi r}}\sin\frac{\theta}{2}\cos\frac{\theta}{2}\cos\frac{3\theta}{2}\end{aligned} \tag{4.5}$$

where $r \ll a$ and the stress intensity factor, K_I, is given by equation (4.2).

Analysing figure 4.11, it is possible to see that the results converge to the analytical solution when the number of nodes is increased. In relation to the type of the influence domain, there is not a significant difference in the SIFs when using first or second degree natural neighbours. It is interesting to compare the results of figure 4.11 with the results obtained with the provided NNRPIM code, in figure 4.6. It is possible to verify that, in both cases, the results are close to the analytical solution using approximately 2000 nodes.

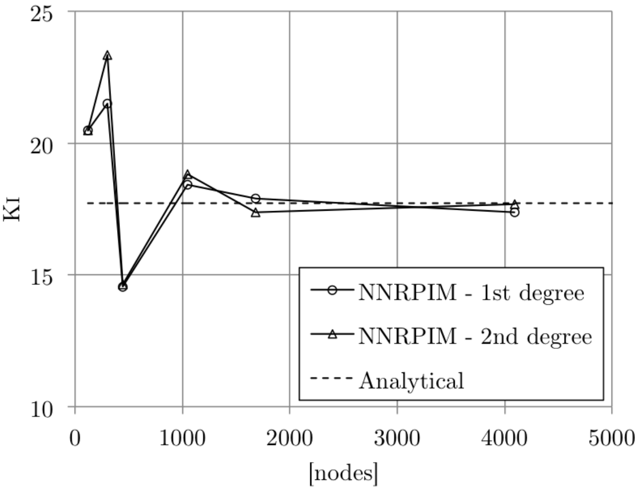


Figure 4.11: Evolution of the mode I SIF with the increase of the number of nodes.

In figure 4.12, σ_y along the xx axis is shown. Those results were obtained using second degree influence domains. σ_y increases substantially near the crack tip, which is expected, given equations (4.5). The curve is smooth and continuos up to that point. The discrete values marked in the graph are the values of the stress at the node in the corresponding position.

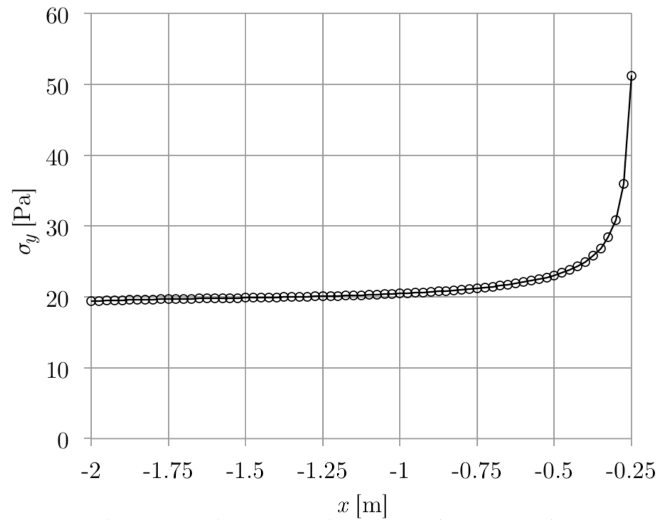


Figure 4.12: σ_y along the xx axis ($y = 0$).

Crack path prediction

In this problem, it is known that the crack propagates in a straight line, given the fact that along the xx axis the shear stress is zero. As this is an introductory example, it is important to check if the crack grows in the predicted direction, and also, inspect the behaviour of the method for irregular meshes. In figure 4.13 there is a detailed comparison between the crack path for regular and irregular nodal meshes. For a regular mesh the crack grows in a straight line, in the case of an irregular mesh there is a slight variation due to the irregular nodal distribution.

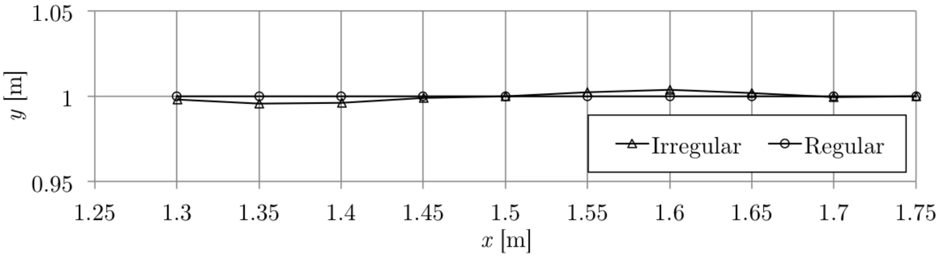


Figure 4.13: Crack path prediction for the central crack in an infinite plate.

4.2.2 Example 2

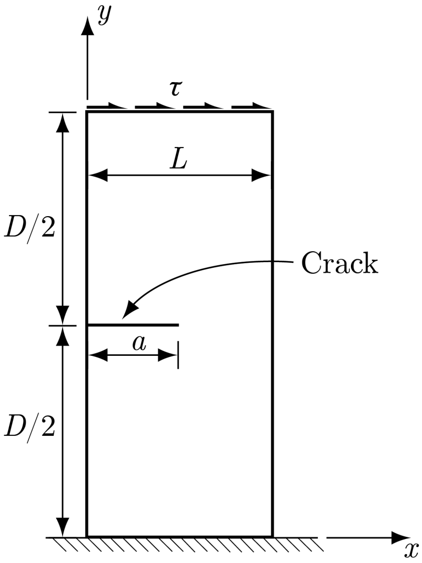


Figure 4.14: Schematic representation of the problem of the example 2.

In this example the accuracy of the NNRPIM in predicting the crack path is tested. The problem is a rectangular plate that contains an edge crack in the middle of its height, the lower edge is fixed and a shear load is applied in the upper edge. The crack is subjected to mixed mode loading, therefore will not propagate in a straight line, making the prediction of the crack path an interesting fracture mechanics problem.

Table 4.5: Relevant data regarding example 2.

Relevant data (example 2)		
Load	0.1	[Pa]
E	1000	[Pa]
ν	0.3	
e	0.5	[m]
L	7	[m]
D	16	[m]
a	3.5	[m]
divl	60	
divd	120	

All the data relevant to this problem is displayed in the table 4.5, and in figure 4.14 there is a schematic representation of the plate. Two types of nodal meshes were used, one refined in the crack area and another one completely uniform, both are shown if figure 4.15. The mesh is refined in crack area in order to improve the quality of the results. In this example, first degree influence domains are used, except when mentioned otherwise. All the results in this example are compared to the solution obtained using FEM by Rao et al. in [32].

Nodal mesh variation

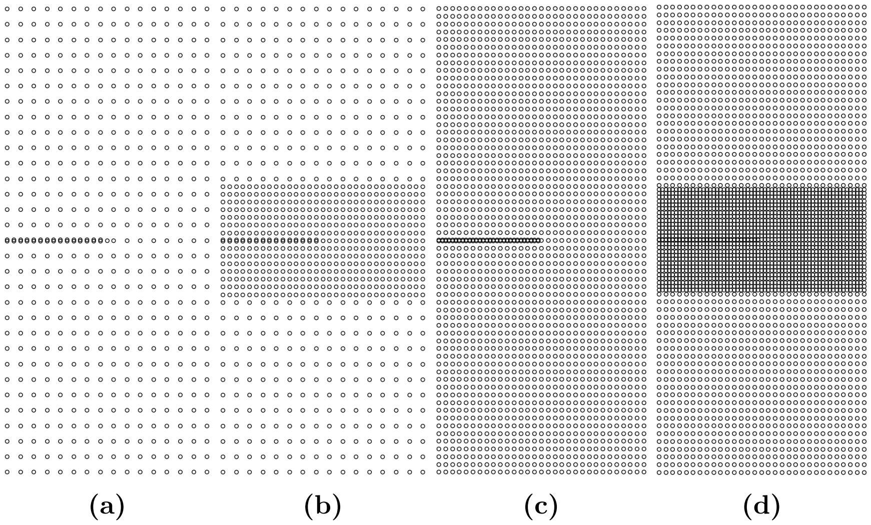


Figure 4.15: Nodal meshes used in example 2: (a) 16 × 31; (b) 16 × 31 (refined); (c) 31 × 61; (d) 31 × 61 (refined).

In order to verify if the results converge and define a nodal discretisation, the number of nodes used in the analysis was sequentially increased. The nodal meshes used in this example are displayed in figure 4.15, and the obtained results are shown in figure 4.16, those where generated using a crack increment equal to the nodal distance in the crack area.

Observing figure 4.16 it is possible to verify that the solution converges and that the NNRPIM gives acceptable and smooth results for 16 × 31 (refined) nodal meshes and superior. Analysing the figure, its possible to see that the results, using a 31 × 61 (refined) and a 61 × 121 meshes, are very similar. For that reason, for the subsequent analysis in this example, a 31 × 61 (refined) nodal mesh was used, since compared with the 61 × 121 nodal mesh, it allows for accurate results with less processing power.

Crack increment variation

To study the influence of the crack increment in the prediction of the crack path, several simulations were performed with different values of that parameter. The results are displayed in figure 4.17.

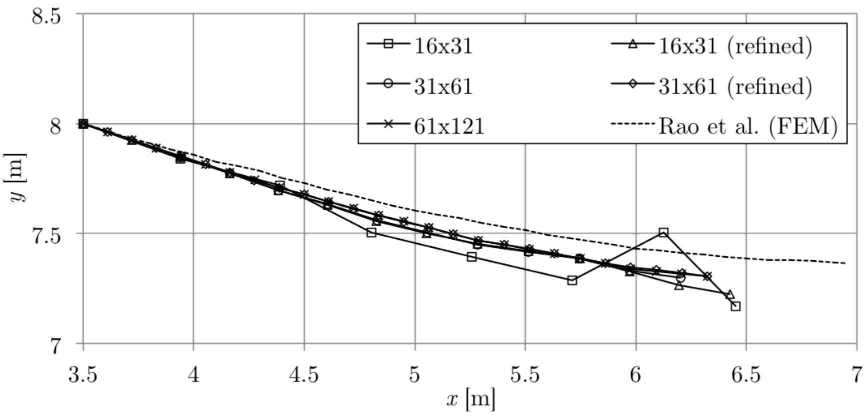


Figure 4.16: Prediction of the crack path for example 2, **for different nodal meshes**. $r_s = 6h$.

Taking the obtained results into consideration, all the studied values yield an acceptable crack path, with the exception of the result that corresponds to the crack increment equal to $2h$. This result is less accurate because there is a big difference, in nodal distance, between the nodes in the crack and the nodes in the rest of the domain. That causes the appearance of inaccurate oscillations in the interpolation functions, which worsens the quality of the results. Taking all the results into consideration, the optimal crack increment is $0.5h$, because the predicted crack path is almost equal to the crack path used as reference.

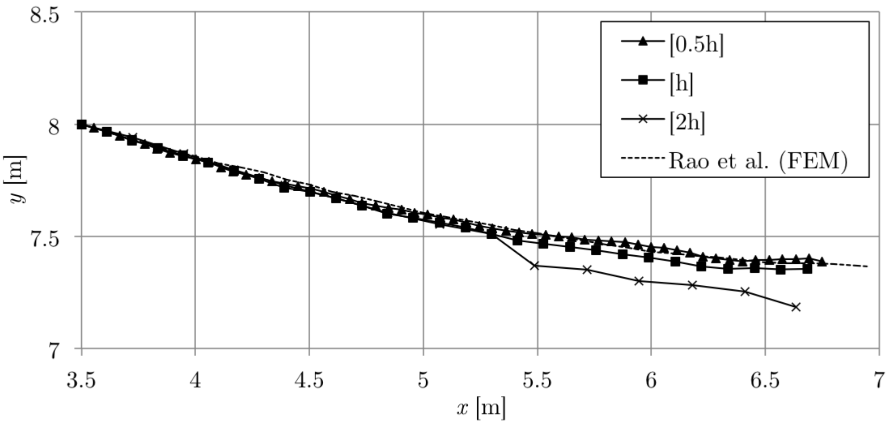


Figure 4.17: Prediction of the crack path for example 2, **for different values of crack increment**. $r_s = 3h$.

Variation of r_s

In figure 4.18 are represented the crack path predictions for different values of r_s. The variation of this parameter has a very significant influence on the crack path prediction. The crack paths for $r_s = 3h$ and $r_s = 1.5h$ are similar between themselves and to the reference path. Those values yield the most accurate results and are considered as optimal in the remaining analysis of this example.

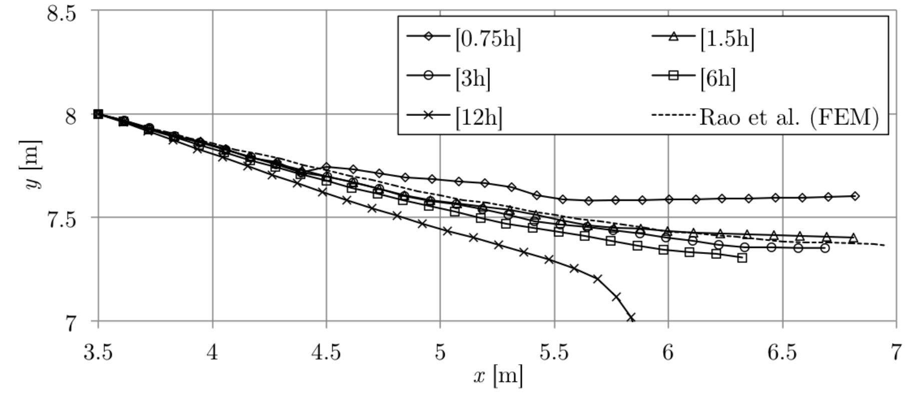


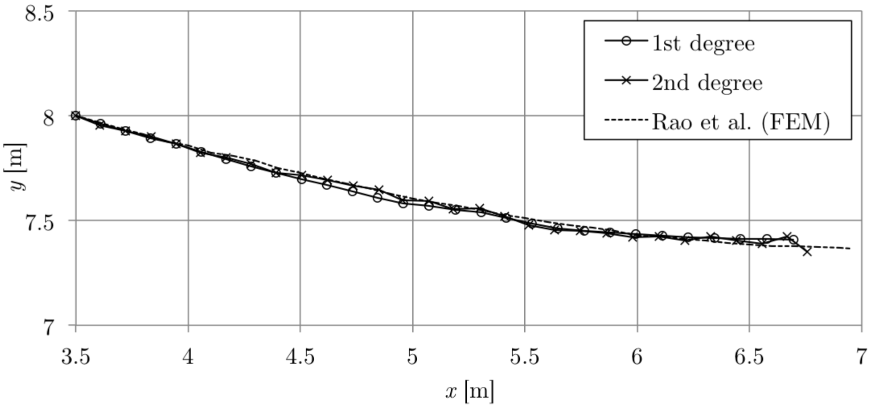
Figure 4.18: Prediction of the crack path for example 2, **for different values of r_s**. Crack increment equal to h

Influence domain variation

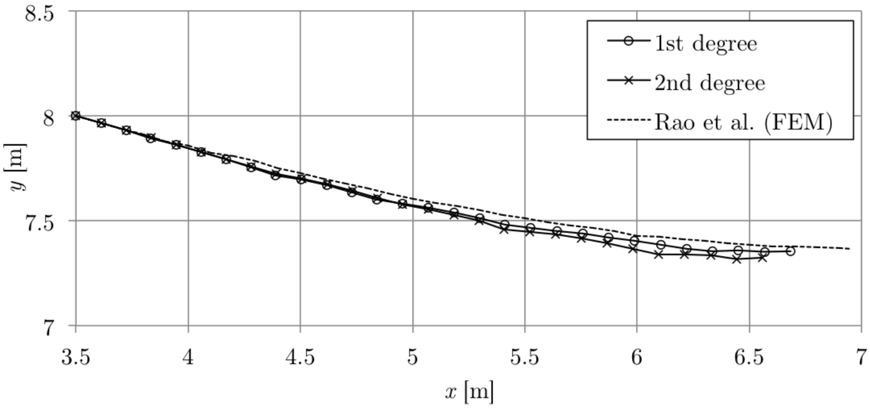
In all of the results above, only first degree influence domains were used. It has been proven that, in the NNRPIM, second degree influence domains provide more accurate results [1]. In this study that affirmation is tested in fracture mechanics problems. The influence of each of the studied parameters, in the results, can not be completely isolated. This is verified, specially when studying the effect of the influence domain in the crack path prediction. In this analysis, various values for other parameters are studied in conjunction with the variation of the influence domain. This methodology may open the possibility of finding a set of optimal parameters and provide a way to generalise the work developed so far.

Examining the effect of the influence domain variation, **in conjunction with the variation of r_s**, figure 4.19 shows that the results are similar, therefore, it is not possible to define which one is more accurate.

Studying now the effect of the influence domain **in conjunction with the crack increment**, with the relevant results being in figures 4.19 and 4.20. For the conditions in figure 4.20a it is verified that the second degree influence domain results are more accurate. Comparing figures 4.20b and 4.19b, where the only parameter variation is the crack increment, it is possi-



(a)



(b)

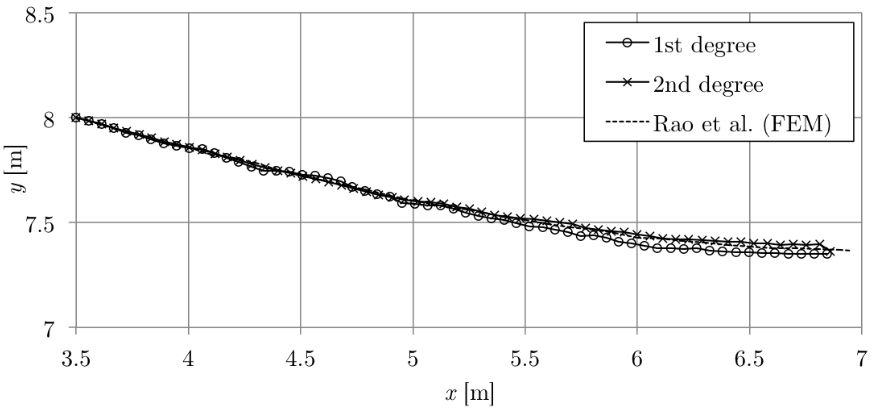
Figure 4.19: Influence domain variation, for different values of r_s; (a) $r_s = 1.5h$; (b) $r_s = 3h$. The crack increment is h

ble to verify that, when the crack increment is shorter, the influence domain causes more variation in the results for the crack path.

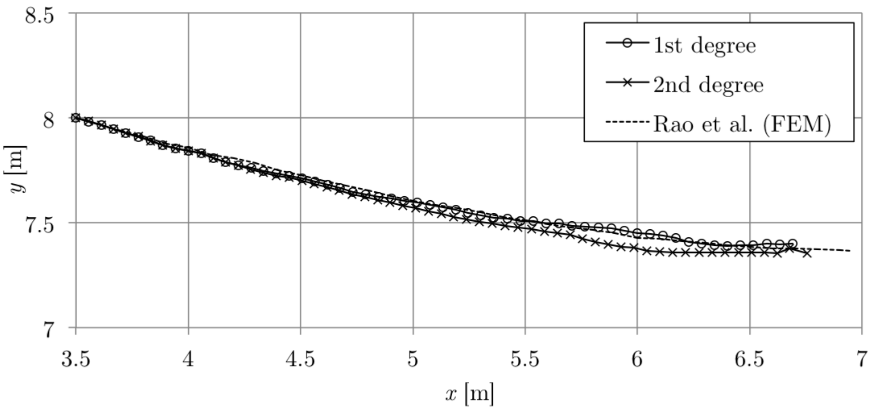
The crack path using a second degree influence domain, from figure 4.20a is considered to be the most accurate result, in table 4.6 the optimal parameters for this example are shown.

Table 4.6: Optimal parameters for example 2.

Optimal parameters (example 2)		
Crack increment	$0.5h$	[m]
r_s	$1.5h$	[m]
Influence domain	Second degree	



(a)



(b)

Figure 4.20: Influence domain variation, for a crack increment equal to $0.5h$: (a) $r_s = 1.5h$; (b) $r_s = 3h$.

4.2.3 Example 3

In example 3 the crack propagation on a beam bent in three points is studied. The problem is represented in figure 4.21 and all relevant data is displayed in table 4.7. In this problem, a beam is loaded with a point force in the middle of its length, it is supported in two points at the same distance from the middle and has an initial vertical crack 50 mm to the left. The results obtained with the NNRPIM are compared with the solution obtained by Geniaut et al. in [48].

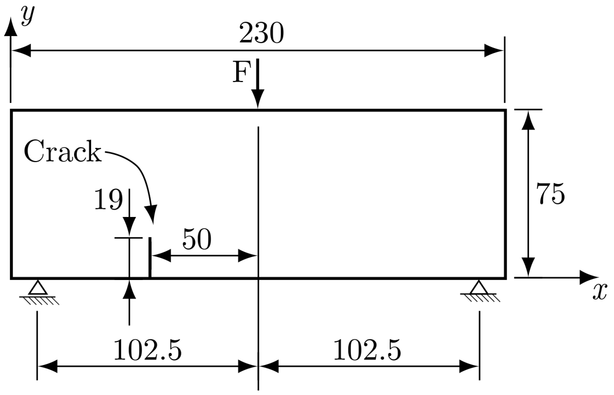
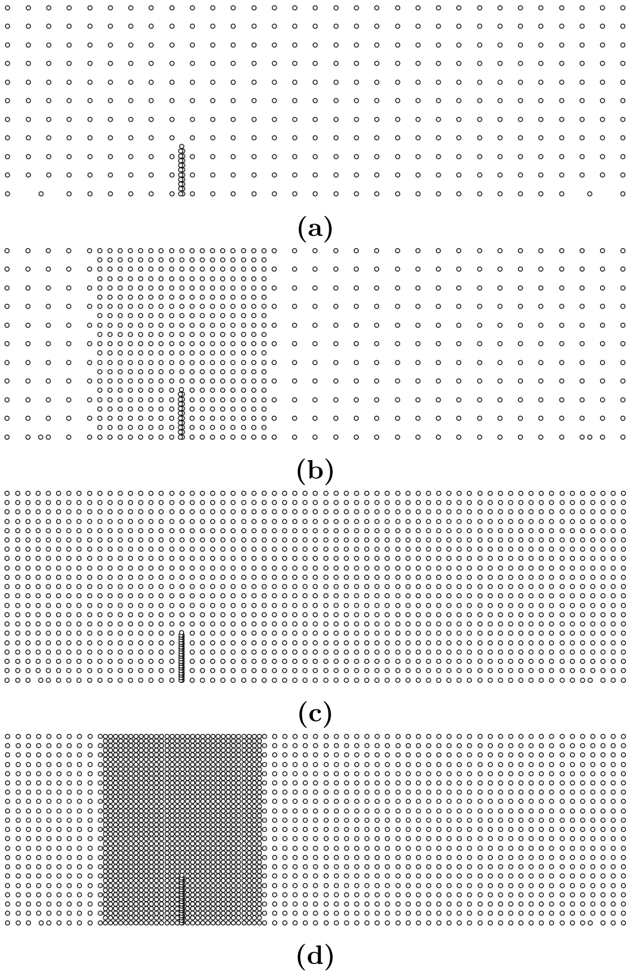


Figure 4.21: Schematic representation of the problem of the example 3. Measurements in [mm].

Two types of nodal mesh were used, one with the nodes uniformly distributed throughout the domain of the problem and another with more nodes in the area of the crack (refined). It is known that the stress field around the crack is complex, therefore it is advantageous to have more nodes in that area. Some of the nodal meshes used in this problem are displayed in figure 4.22.

Table 4.7: Relevant data regarding example 3.

Relevant data (example 3)		
F	0.01	[N]
E	1000	[Pa]
ν	0.3	
Thickness	0.01	[m]
L	0.230	[m]
divl	120	
divd	40	

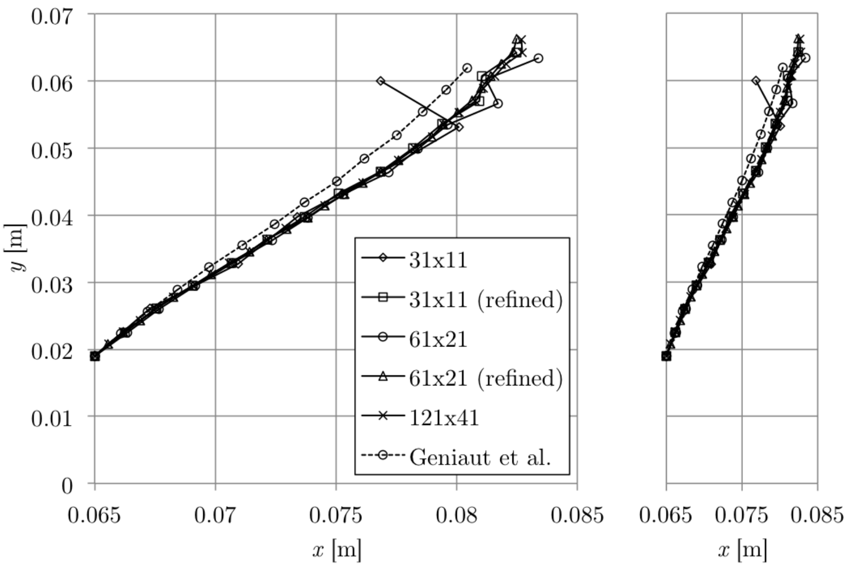

(a)

(b)

(c)

(d)

Figure 4.22: Some of the nodal meshes used in example 3: (a) 31 × 11; (b) 31 × 11 (refined); (c) 61 × 21; (d) 61 × 21 (refined).



(a) **(b)**

Figure 4.23: Crack path prediction for example 3, **for various nodal meshes**: (a) xx axis is elongated in relation to yy; (b) the scale of both axes is the same. $r_s = 3h$.

Nodal mesh variation

The first study performed in this example consists in the gradual refinement of the nodal mesh. The results for this analysis are shown in figure 4.23. The crack increment is equal to the inter-nodal distance in the crack area, in all the simulations performed in this analysis.

Analysing figure 4.23 it is clear that the result converges with the increase of the number of nodes. Observing all the curves, it is possible to conclude that the results with a 61×21 (refined) mesh and the ones with a 121×41 meshes are similar hence, in this example, all the subsequent results were obtained using the 61×21 (refined) nodal mesh, unless otherwise mentioned.

Crack increment variation

The various crack paths for different crack increment values are displayed in figure 4.24. All values provide very similar results, which indicates that, in this case, the crack increment variation does not alter the results. Any result could be an optimal value, the other tests in this example are performed with the crack increment equal to h, except when otherwise mentioned.

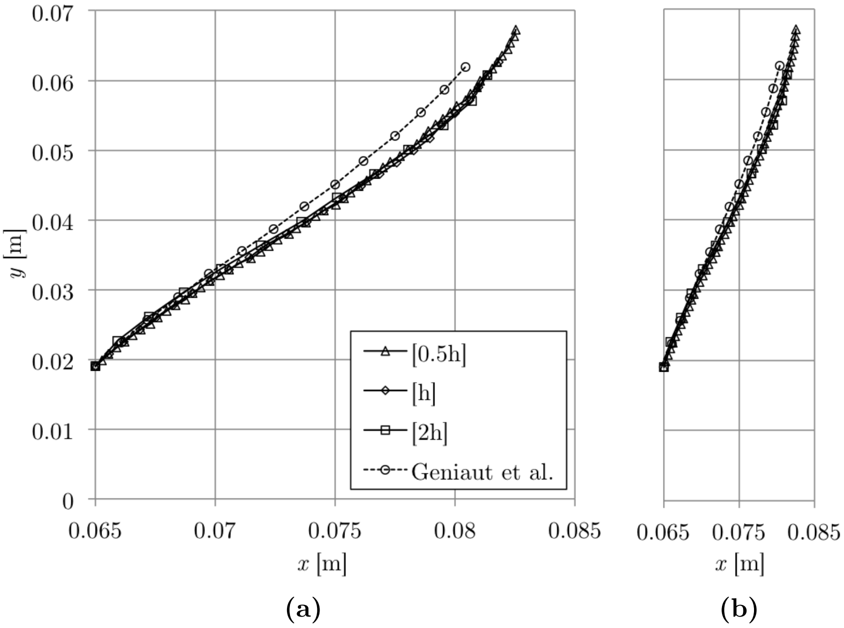


Figure 4.24: Crack path prediction for example 3, **for various values of crack increment**: (a) xx axis is elongated in relation to yy; (b) the scale of both axis is the same. $r_s = 3h$.

r_s variation

In figure 4.25 the results for the crack path prediction for different values of r_s are shown. The variation of this parameter has a significant influence in the crack path and, as its value lowers, the NNRPIM's result approximates the reference solution, as was the case in the previous example.

Considering the information provided so far, the optimal result is either $0.75h$ or $1.5h$, the former provides the most proximity to the reference path, but shows some unwanted oscillations. Using $r_s = 1.5h$, the obtained result is not as close to the reference, however it is not affected by oscillations. In the remaining of this example both values for r_s are studied.

Influence domain variation

In this study, the effect of the influence domain is analysed. In order to define a set of optimal parameters for this problem, the variation of the influence domain is made in conjunction with other parameters.

Varying the influence domain **in conjunction with** r_s the results in figure 4.26 are obtained. For the both values of r_s, there are significant differences in the crack path. Making $r_s = 0.75h$ and using second degree influence domains, the result is completely inaccurate. There is a very significant difference in the results caused by the variation of the influence

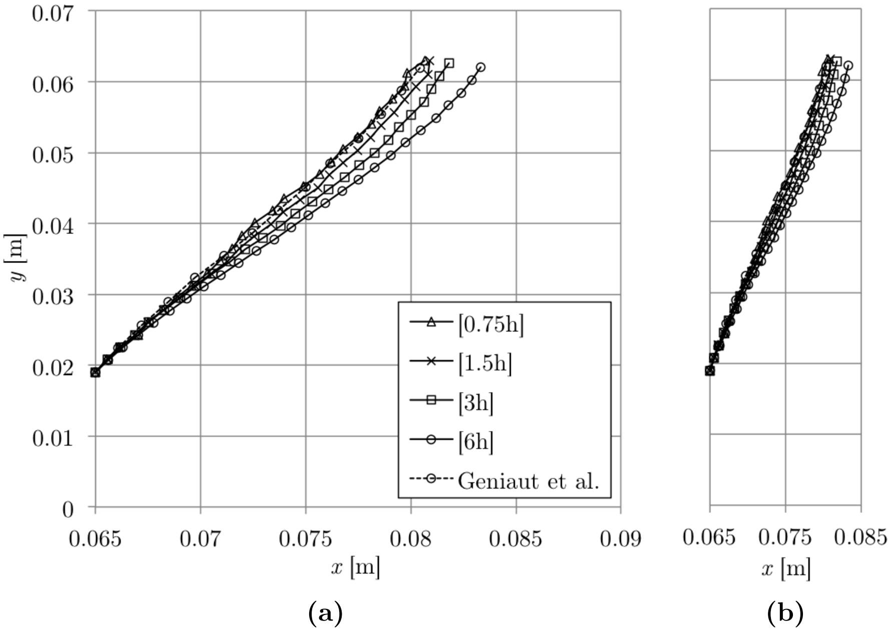
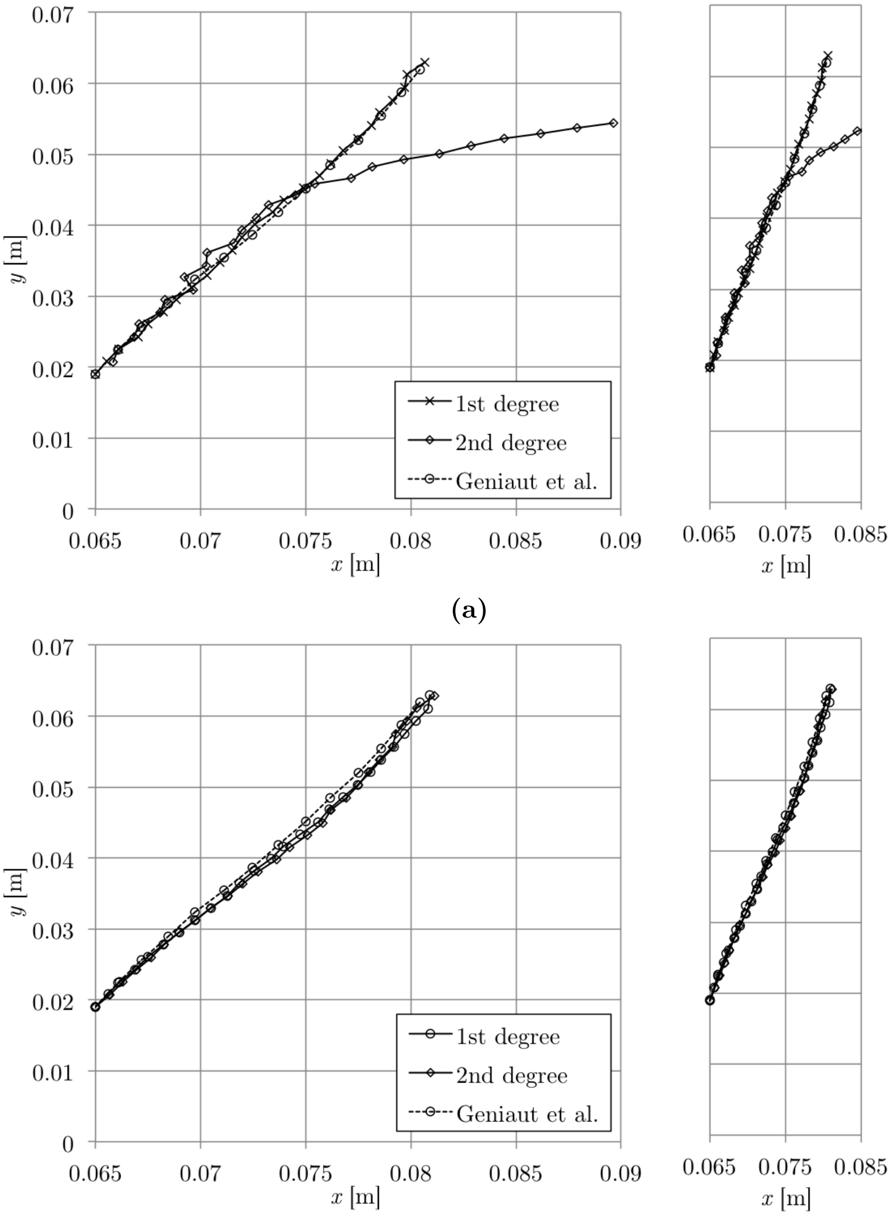


Figure 4.25: Crack path prediction for example 3, **for various values of** r_s: (a) xx axis is elongated in relation to yy; (b) the scale of both axis is the same. The crack increment is equal to h.

domain. The inaccuracy of the result using second degree influence domains may indicate that to few integration points are being selected. With $r_s = 0.75h$ using first degree influence domains, the result is similar to the reference. For $r_s = 1.5h$, only in the final part of the crack path significant differences are verified, with the crack path using second degree influence domains approaching the reference data.

The results of the variation of the influence domain **in conjunction with the crack increment** are displayed in figures 4.26 and 4.27. Using a smaller crack increment, causes the variation of the influence domain to have a greater effect on the crack path. In this example, it is clear that the crack increment of $0.5h$ results in less accurate crack path predictions than when using $\Delta a = h$. Comparing the two types of influence domains, for the situation in figure 4.27 the second degree influence domains provide a more smooth crack path, contrary to what happened when using the shorter increment.

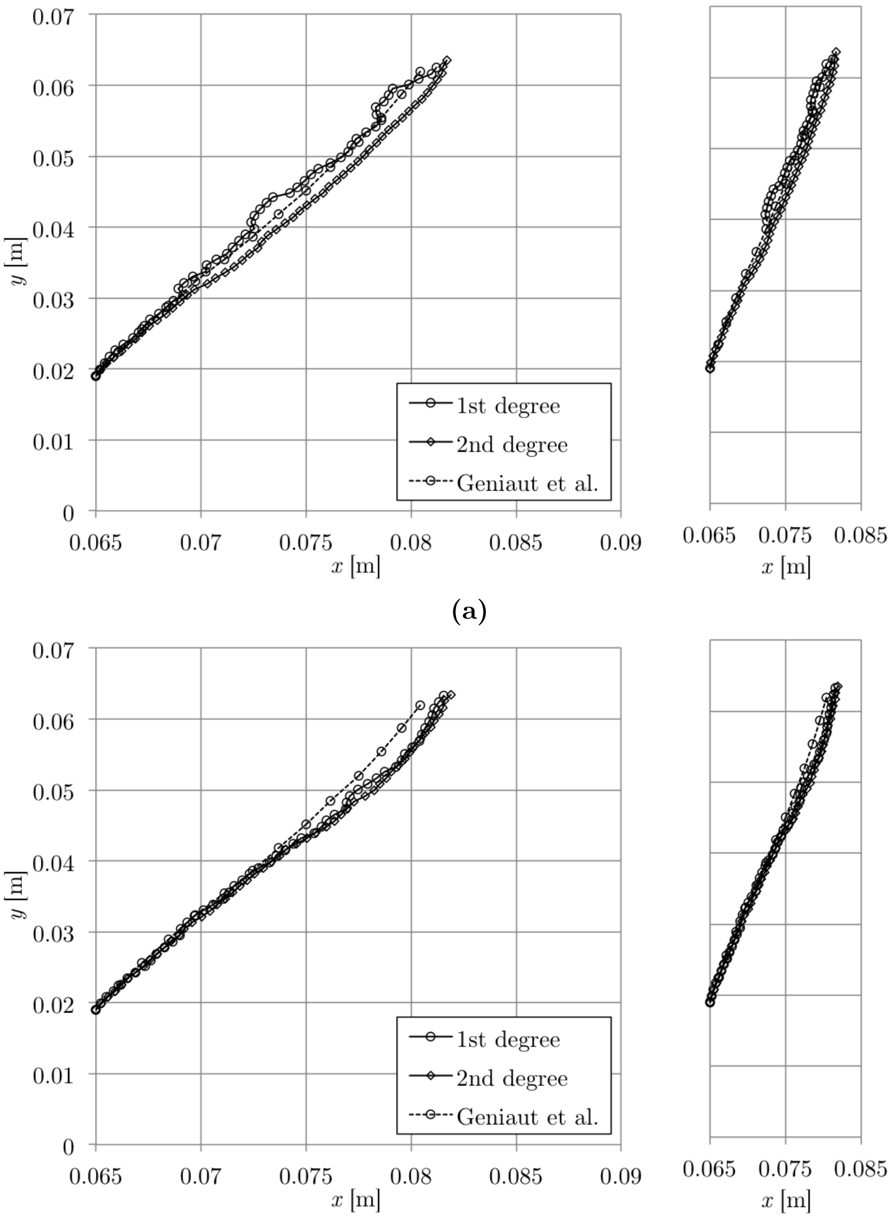
The optimal results found for example 3 are displayed in figure 4.26b. It is clear that first degree influence domains are optimal for this problem. In table 4.8 the optimal values for the studied parameters are shown.



(a)

(b)

Figure 4.26: Crack path prediction for example 3, **for various influence domains**: (a) $r_s = 0.75h$; (b) $r_s = 1.5h$. The crack increment is h.



(a)

(b)

Figure 4.27: Influence domain variation for example 3, **for a crack increment of** $0.5h$: (a) $r_s = 0.75h$; (b) $r_s = 1.5h$.

Table 4.8: Optimal parameters for example 3.

Optimal parameters (example 3)		
Crack increment	h	[m]
r_s	$1.5h$	[m]
Influence domain	First degree	

4.2.4 Example 4

In the present example the crack propagation in a beam bent in 4 points with a circular hole is studied. The beam contains an initial vertical crack in the middle of its length. The circular hole is not in the centre of the beam, thus the crack is subjected to mixed mode loading. The problem is presented in figure 4.28, in table 4.9 additional relevant data is displayed.

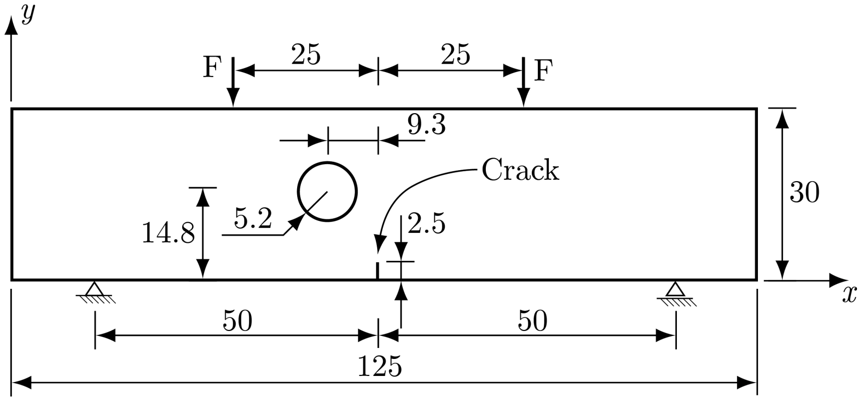


Figure 4.28: Schematic representation of the problem of the example 4. Measurements in [mm].

The problem was discretised in two types of nodal meshes, one with a uniform nodal distribution and another one with more nodes in the area of the crack and hole. Some of the nodal meshes used in this problem are displayed in figure 4.29. The crack path prediction for this problem, by Nguyen-Xuan et al. [30] is used as reference.

Table 4.9: Relevant data regarding example 4.

Relevant data (example 4)		
F	0.0005	[N]
E	1000	[Pa]
ν	0.29	
Thickness	0.01	[m]
L	0.125	[m]
divl	160	
divd	40	

Nodal mesh variation

The results in figure 4.30 show that the solution converges when the number of nodes discretising the problem domain, is increased. The results were obtained using a crack increment equal to the nodal distance near the crack.

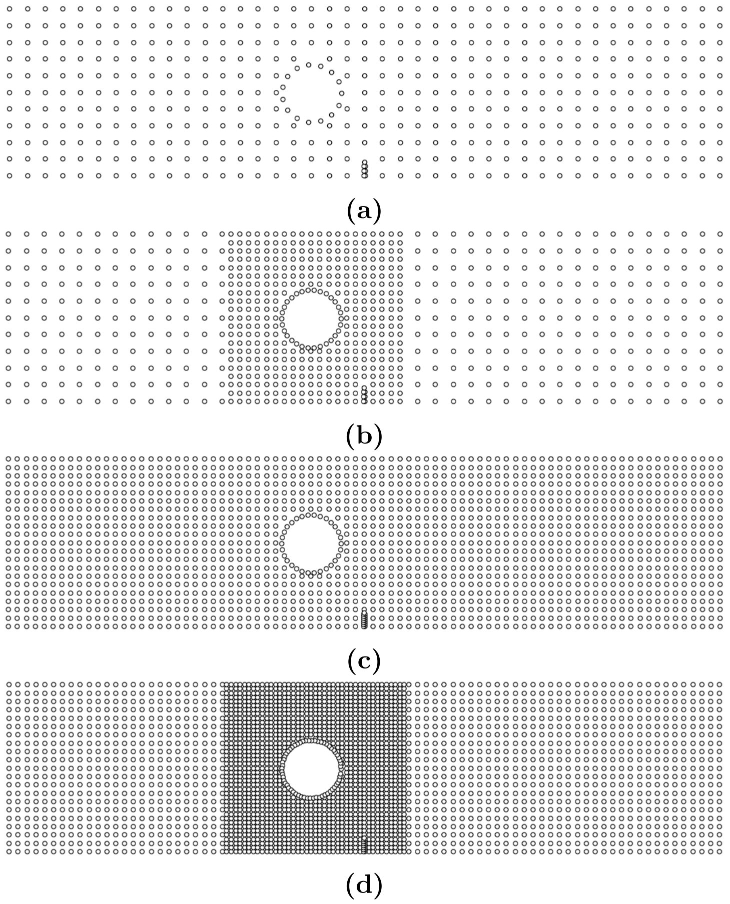


Figure 4.29: Some of the nodal meshes used in example 4: (a) 41 × 11; (b) 41 × 11 (refined); (c) 81 × 21; (d) 81 × 21 (refined).

All the nodal meshes, besides the 40 × 10, yield acceptable results. All the results obtained when using a refined nodal mesh of 80 × 20 or superior, are similar. The subsequent tests of this example are performed with a 80 × 20 refined nodal mesh.

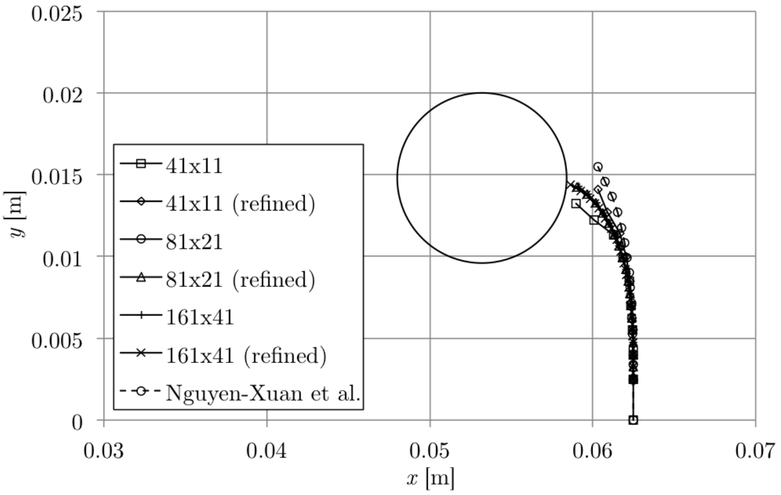


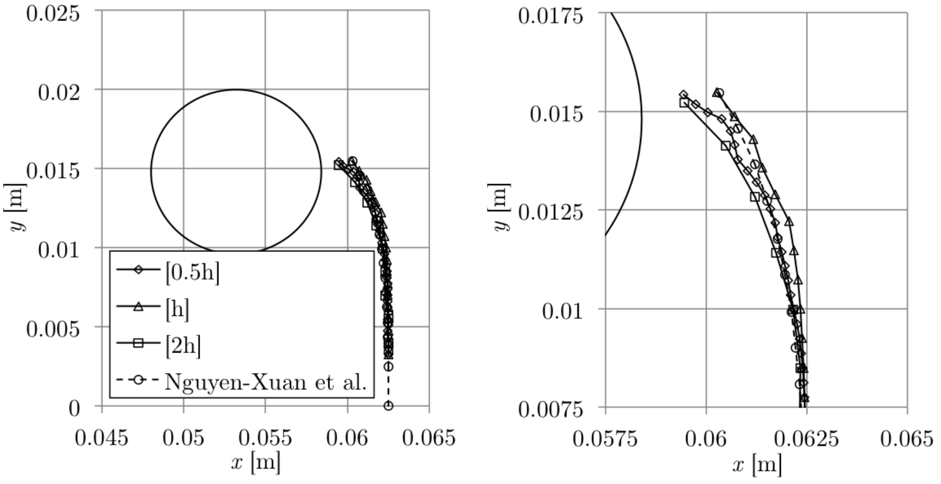
Figure 4.30: Crack path prediction for example 4, **for various nodal meshes**. $r_s = 4h$.

Crack increment variation

The results for the crack increment variation are displayed in figure 4.31. Using a crack increment equal to $2h$, yields a less accurate result. This is caused by the fact that the spacing of the nodes that belong to the crack, is significantly larger than the nodes near it. That results in poor quality interpolation functions. Shortening the crack increment originates more accurate and acceptable results. The optimal value for the crack increment is either $0.5h$ or h. The crack path for $\Delta a = 0.5h$ shows some oscillations in the final part of the crack but is similar to the reference solution in the remaining of the path.

r_s variation

In figure 4.32 the crack paths for different values of r_s are shown. r_s has a significant influence in the crack path. The crack path curves in the direction of the hole at an inferior height for larger values of r_s. That effect is caused by the fact that, when using a larger radius, the stress field near the hole has more influence in the calculation of the crack's propagation direction. When $r_s = 0.5h$ the opposite is verified, the crack starts approaching the hole on a final stage. That happens because the stress field near the hole has



(a) **(b)**

Figure 4.31: Crack path prediction for example 4 **for various crack increments**: (a) general view of the crack path; (b) close up near the crack tip. $r_s = h$.

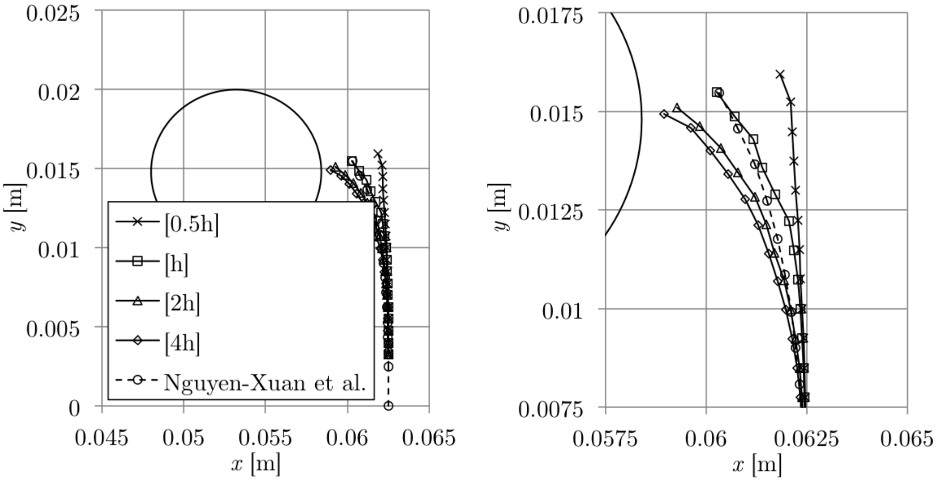
less effect in the calculation of the crack's propagation direction. In the case represented in figure 4.32, the optimal value for the crack path is $r_s = h$.

Influence domain variation

The effect of the influence domain in the crack path is studied in conjunction with the variation of Δa and r_s.

Analysing the effect of the influence domain variation, **in conjunction with** r_s, figure 4.33 shows that the influence domain has a significant effect on the results. The use of second degree influence domains causes less smooth and less accurate results. The pejorative effect of the use of second degree influence domains, is caused by the proximity of the crack tip to a boundary of the domain. The influence domains near the boundaries of the problem are smaller than in the rest of the domain. This size difference is more significant when using second degree influence domains thus, the results when using this type are less accurate. r_s causes less variations in the results when using second degree influence-cells, that is visible in figures 4.33 and 4.34.

Analysing the effect of the influence domain variation, **in conjunction the crack increment**, the results in figures 4.33 and 4.34 are compared. For a shorter crack increment, the use of second degree influence domains provides a more smooth crack path. As was the case when using $\Delta a = h$, the use of second degree influence domains causes the crack path to move closer to the hole. The use of the shorter crack increment when the influence-cells are of first degree causes the crack path to be less smooth, for all values of



(a) **(b)**

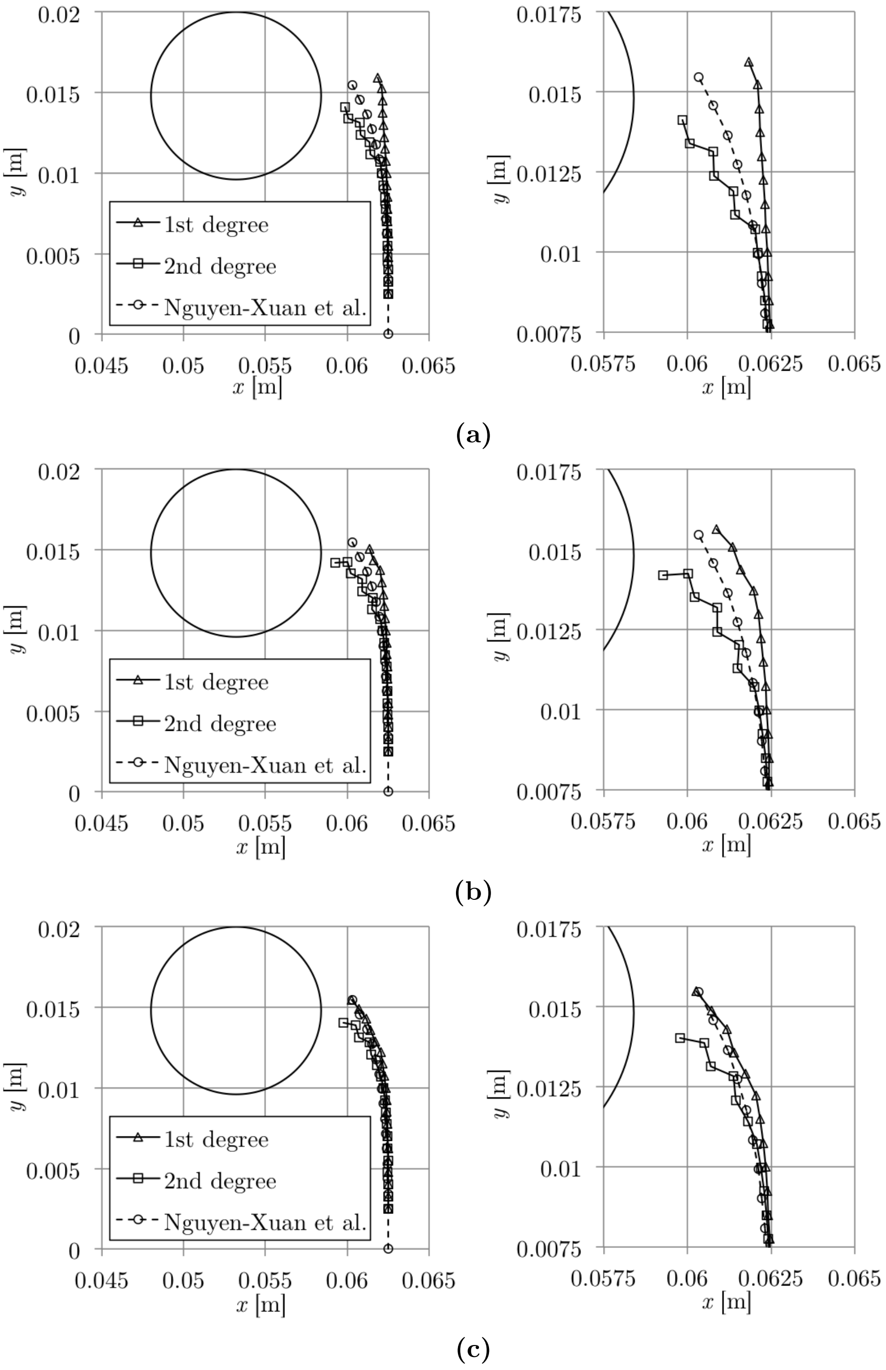
Figure 4.32: Crack path prediction for example 4, **for various** r_s: (a) general view of the crack path; (b) close up near the crack tip. The crack increment is equal to h.

r_s. Making $r_s = 0.75h$ and $\Delta a = 0.5h$ increases the accuracy of the result, when comparing with the crack path using a longer increment.

The optimal crack path is represented in figure 4.34b, with first degree influence domains. The optimal values are presented in table 4.10.

Table 4.10: Optimal parameters for example 4.

Optimal parameters (example 4)		
Crack increment	$0.5h$	[m]
r_s	$0.75h$	[m]
Influence domain	First degree	

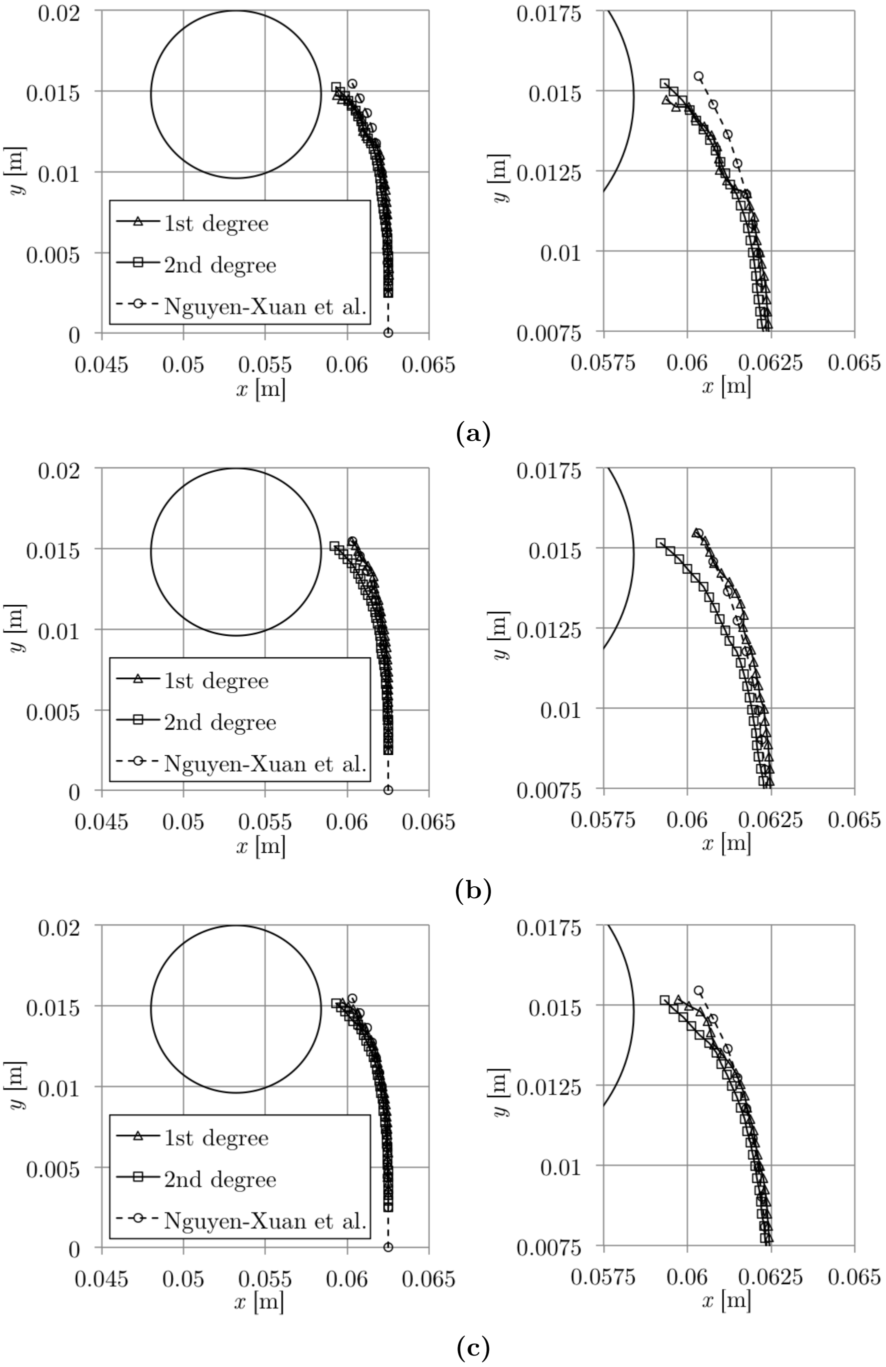


(a)

(b)

(c)

Figure 4.33: Crack path prediction for example 4, **for various values of** r_s: (a) $r_s = 0.5h$; (b) $r_s = 0.75h$; (c) $r_s = h$. Crack increment equal to h.



(a)

(b)

(c)

Figure 4.34: Crack path prediction for example 4, **for a crack increment of** $0.5h$: (a) $r_s = 0.5h$; (b) $r_s = 0.75h$; (c) $r_s = h$.

4.2.5 Example 5

In example 5, the crack path prediction in a beam, that contains three circular holes, bent in three points is made. The problem is presented in figure 4.35. The beam is loaded by the force F and supported in two points positioned at the same distance from the middle of the beam and in the xx axis. The beam contains an initial vertical crack in its lower edge. As the crack is not in the centre of the beam, it is subjected to mixed mode loading. The presence of the circular holes has a significant effect on the stress field along the beam, which will affect the crack path. Additional data relevant to this example is displayed in table 4.11.

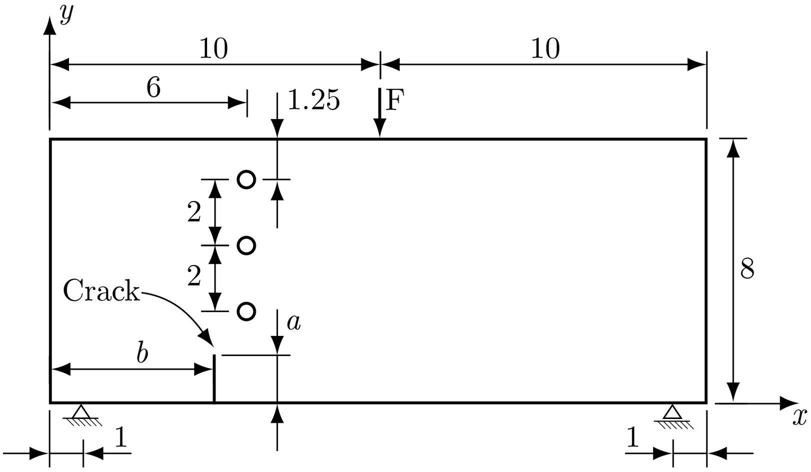


Figure 4.35: Schematic representation of the problem of the example 5 and 6. Measurements in [m].

Some of the nodal meshes used in this example are shown in figure 4.36. Two types can be seen, one with uniformly distributed nodes along the domain and another one with more nodes near the crack (refined). In this example, first degree influence domains are used, except when otherwise mentioned. The results obtained in this work are compared with the results obtained by Bittencourt et al. in [49].

Nodal mesh variation

In order to verify if the solution converges, and to chose the most suitable discretisation for the remaining of the example, the number of nodes used in the analysis was sequentially increased. The obtained crack paths are displayed in figure 4.37. In this study, a crack increment equal to the nodal spacing near the crack, was used.

The resulting path is curved to the middle of the beam and intersects the middle hole. The crack is subjected to mixed mode loading, therefore

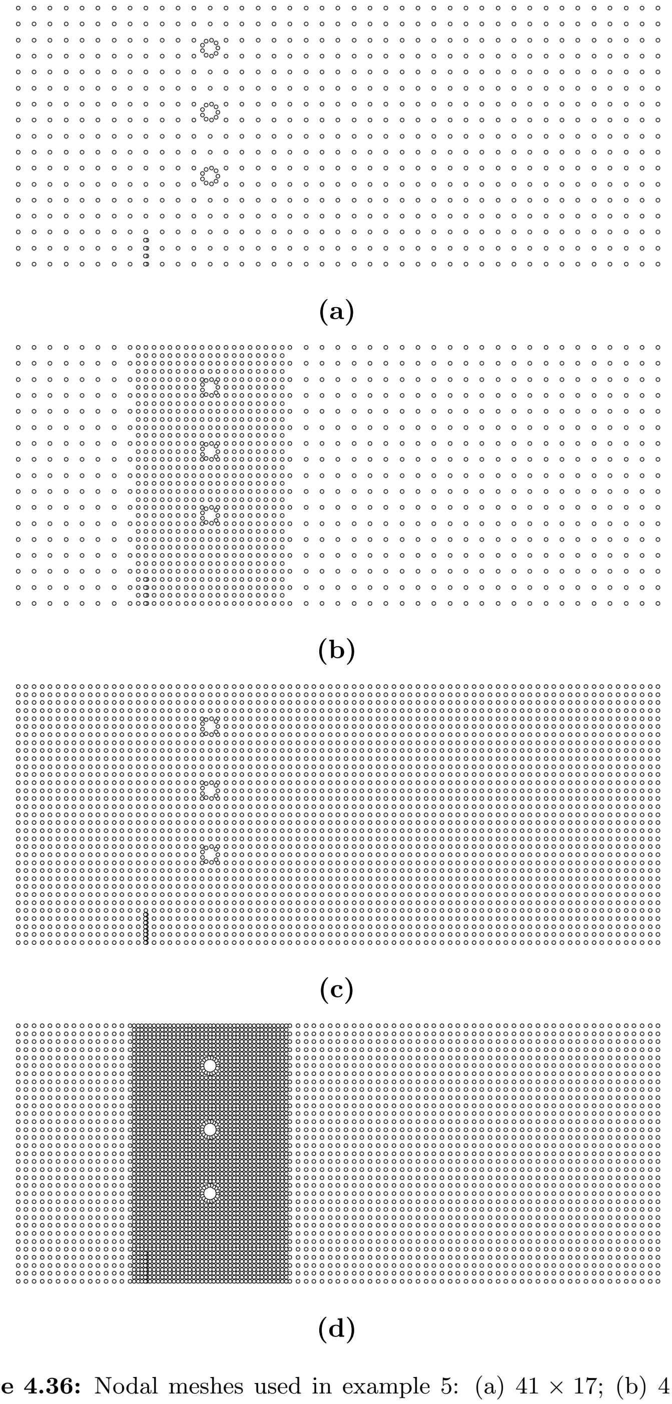


Figure 4.36: Nodal meshes used in example 5: (a) 41 × 17; (b) 41 × 17 (refined); (c) 81 × 33; (d) 81 × 33 (refined).

Table 4.11: Relevant data regarding example 5.

Relevant data (example 5)		
F	10	[N]
E	1000	[Pa]
ν	0.29	
Thickness	0.5	[m]
a	1	[m]
b	4	[m]
L	20	[m]
$divl$	160	
$divd$	64	

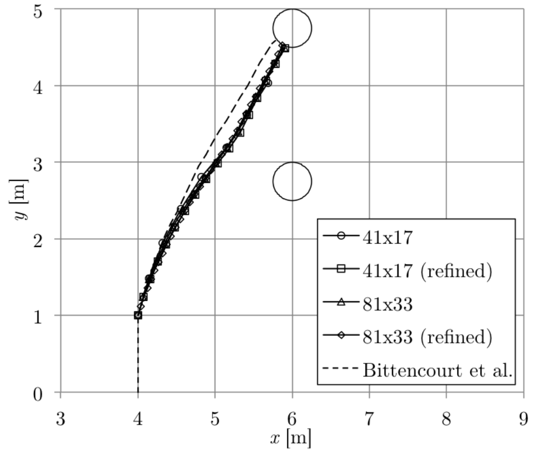


Figure 4.37: Crack path prediction for example 5, **for various nodal meshes**. $r_s = 8h$.

the crack path was expected to be curved. The presence of the holes has a significant effect on the stress distribution along the beam, the fact that the crack path intersects the hole does not refute that claim. Analysing the figure, it can be seen that, for this conditions, the crack paths are not significantly different for the various modal meshes. That may be due to the fact that a relatively large r_s was used in this analysis. In this problem it is necessary to use a dense nodal mesh in order to model the holes properly, a 81×33 refined nodal mesh is used.

Crack increment variation

In order to study the influence of the crack increment in the crack path, several simulations were performed with different values of that parameter. The results are displayed in figure 4.38.

Analysing the figure, it is possible to conclude that the values $0.5h$ and $1h$

yield an accurate crack path prediction. It is clear that the crack increment equal to $2h$ is an excessive value, for this problem. The remaining analysis of this problem is made with a crack increment equal to h, except when otherwise mentioned.

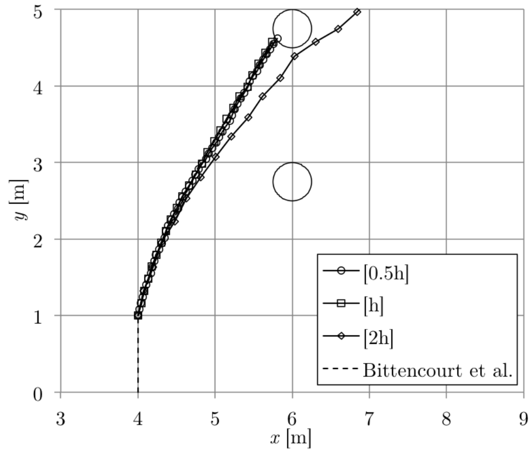


Figure 4.38: Crack path prediction for example 5, **for various crack increments**. $r_s = 2h$.

r_s variation

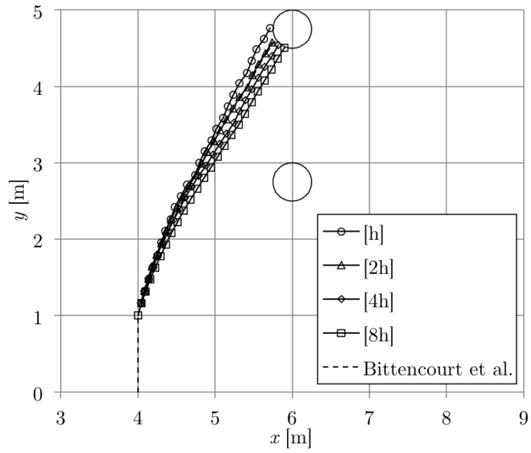


Figure 4.39: Crack path prediction for example 5, **for various values of** r_s. The crack increment is equal to h.

In figure 4.39 the results for the crack path for different values of r_s are shown. The variation of this parameter has a significant effect in the results. The most accurate crack path is obtained with $r_s = 2h$ however (the reference result is under the crack path generated using that value). Making

$r_s = 2h$ and higher, the the path is more inclined to the right side. This is caused by the fact that the stress field near the holes has a more significant effect in the crack's propagation direction. The subsequent analysis of this example will be made with $r_s = h$ and $r_s = 2h$.

Influence domain variation

The effect of influence domain in the crack path is studied in combination with r_s and Δa. The objectives of that approach are, to verify how the various parameters interact and to find a set of optimal values.

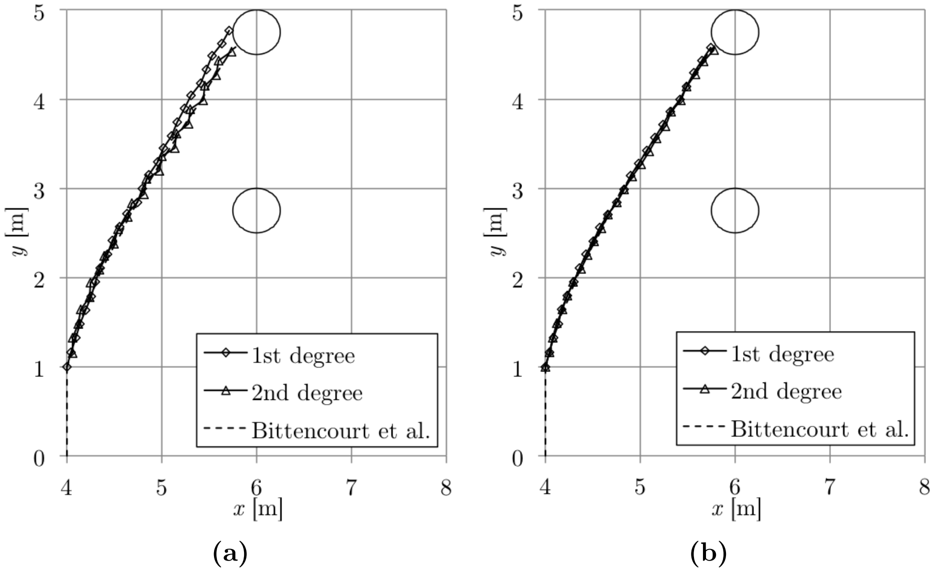


Figure 4.40: Influence domain variation: (a) $r_s = h$; (b) $r_s = 2h$. The crack increment is equal to h.

The results that represent the variation of the type of influence domain **in conjunction with** r_s, are shown in figure 4.40. In figure 4.40a the use of second degree influence domains causes the crack path to approximate the reference solution. Also, some oscillations appear in the result, which may indicate that an excessively small number of integration points is being selected. Making $r_s = 2h$, in figure 4.40b, it is possible to verify that there is not a significant difference in the solution whether first or second degree influence domains are used. Both solutions are on top of the reference crack path.

To study the variation of the influence domain **in combination with the crack increment** the results in figures 4.40 and 4.41 are taken into consideration. It is possible to verify in figure 4.41a that, the use of shorter increments in conjunction with first degree influence domains, results in oscillations in the crack path. Comparing figures 4.40a and 4.41a, it is possible

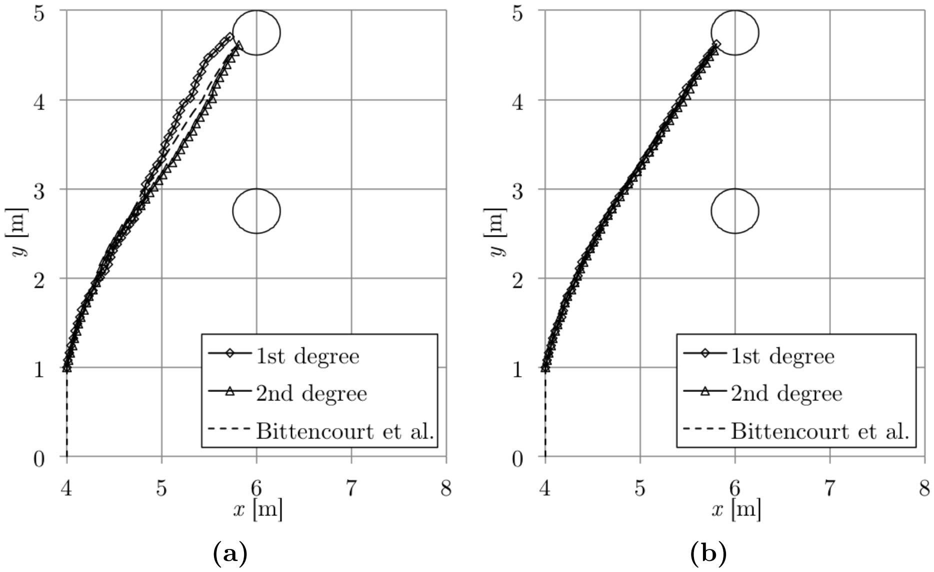


Figure 4.41: Influence domain variation, the crack increment is equal to $0.5h$: (a) $r_s = h$; (b) $r_s = 2h$.

to verify that the oscillations when using $\Delta a = h$ and second degree influence domains, disappear when using a shorter increment. As is displayed in figure 4.41b, the use of a shorter increment and $r_s = 2h$ leads to similar and accurate crack path predictions.

The optimal crack path is obtained using first degree influence domains and is represented in figure 4.40b. The first degree influence domains were chosen due to the lower computational cost when using then. The optimal values for the studied parameters are displayed in table 4.12.

Table 4.12: Optimal parameters for example 5.

Optimal parameters (example 5)		
Crack increment	h	[m]
r_s	$2h$	[m]
Influence domain	First degree	

4.2.6 Example 6

This example is similar to the previous example, figure 4.35, the difference being the crack's length and position (identified as a and b in table 4.13). The variation of those dimensions has a significant impact in the crack path prediction.

In this problem, the crack path is more complex than in previous examples. Special care was taken in refining the areas through which the crack propagates. The nodal meshes used in this problem are shown in figure 4.47. First degree influence domains are used, except when otherwise mentioned. The results obtained by the NNRPIM are compared with numerical results obtained by Geniaut et al. [48] and with experimental results obtained by Ingraffea et al. [50]. The experimental results were obtained with crack mouth opening displacement control in a brittle material.

Table 4.13: Relevant data regarding example 6.

Relevant data (example 6)		
F	10	[N]
E	1000	[Pa]
ν	0.29	
Thickness	0.5	[m]
a	1.5	[m]
b	5	[m]
divl	240	
divd	96	

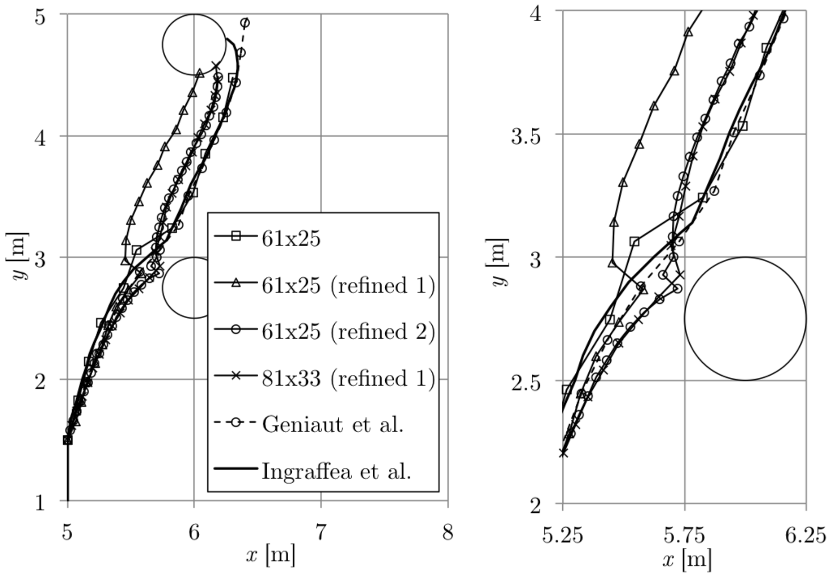
Nodal mesh variation

In order to verify if the results of this example converge, the number of nodes used in the analysis was gradually increased. The relevant results are shown in figure 4.42. It is possible to verify that there is convergence in the results. Using the 81 × 33 (refined 1) and 61 × 25 (refined 2) the results are similar. The remaining of the numerical simulations in this example are performed with a 61 × 25 (refined 2) nodal mesh. The complexity of the crack path requires a fine mesh in the crack area.

Crack increment variation

In order to study the influence of the crack increment in the crack path, several simulations were performed with different values of that parameter. The results are displayed in figure 4.43.

Analysing the figure, it is possible to conclude that making $r_s = 2h$ does not generate an acceptable result. The crack path ends in the lower hole



(a) (b)

Figure 4.42: Crack path prediction for example 6, **for various nodal meshes**: (a) general view of the crack path; (b) close up near the bottom hole. $r_s = 2h$.

instead of the middle one. It was verified in previous examples that the use of this value generates inaccurate results. In this example making $\Delta a = 0.5h$ also results in an inaccurate crack path, with the crack intercepting the bottom hole instead of the middle one. The remaining analysis of this problem is made with a crack increment equal to h, except when otherwise mentioned.

r_s variation

In figure 4.44 the results relevant to the analysis of the effect of r_s are shown. It can be verified that the r_s variation has a significant influence in the crack path prediction. In this example, the most demanding zone for the algorithm, is near the bottom hole. The crack propagates close to it thus, in that area, a smaller number of integration points may be selected. A high accuracy from the method is therefore required. Analysing the figures, it is possible to verify that, in the areas near the holes, the crack path prediction is less accurate, in general. Even though significant inaccuracies are present, all the results end in the middle hole, as was studied experimentally. It should be noted that, using the values in figure 4.44b, the crack path near the holes is not correctly calculated, because the program tries to find integration points in the holes (due to the high values of r_s). Hence, the calculation of the crack's propagation direction is inaccurate. In

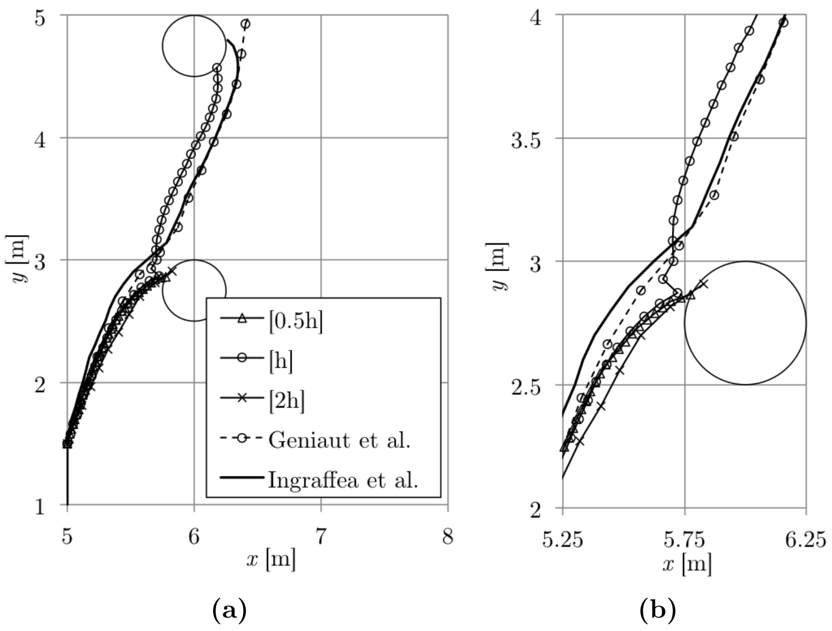


Figure 4.43: Crack path prediction for example 6, **for various crack increments**: (a) general view of the crack path; (b) close up near the bottom hole. $r_s = 2h$.

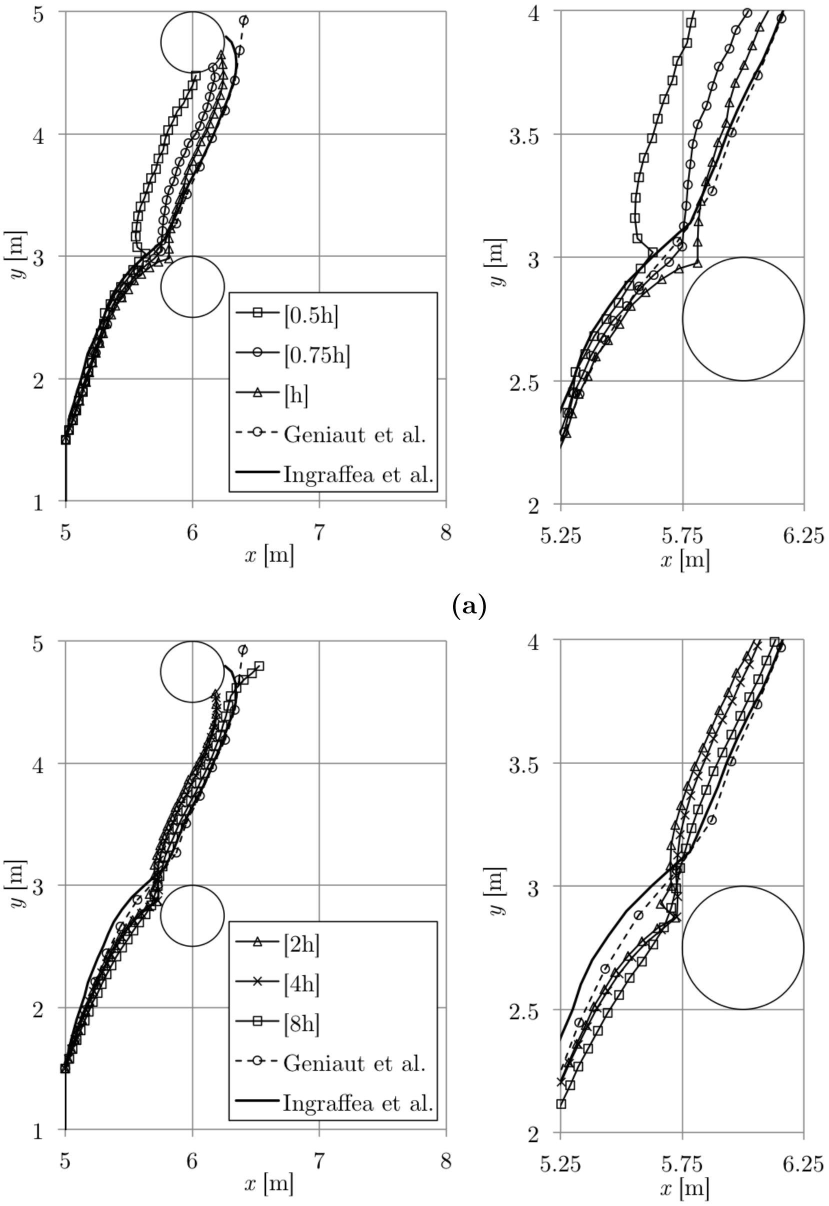
almost all the results, the curve near the middle hole is excessively inclined to the right side. When $r_s = 0.5h$, an excessively low number of integration points is selected, resulting in an inaccurate crack path. Making $r_s = 0.75h$ and $r_s = h$ generates the most accurate crack predictions. These two values are studied in the remaining of this example.

Influence domain variation

The effect of influence domain in the crack path is studied in combination with r_s and Δa. The objectives of that approach are, to verify how the various parameters interact and to find a set of optimal values.

The results that represent the variation of the type of influence domain **in conjunction with** r_s, are shown in figure 4.45. For both values of r_s, using second degree influence domains, yields significantly less accurate results. When the crack approximates the middle hole, the influence domains in that area are smaller. Using second degree influence domains causes that difference to be bigger, resulting in the inaccuracies seen in the figure.

To study the variation of the influence domain **in combination with the crack increment** the results in figures 4.45 and 4.46 are taken into consideration. The use of shorter crack increment does not result in a more accurate crack path prediction. As was the case using $\Delta a = h$, with $\Delta a = 0.5h$, the use of second degree influence domains degrades the quality of the



(a)

(b)

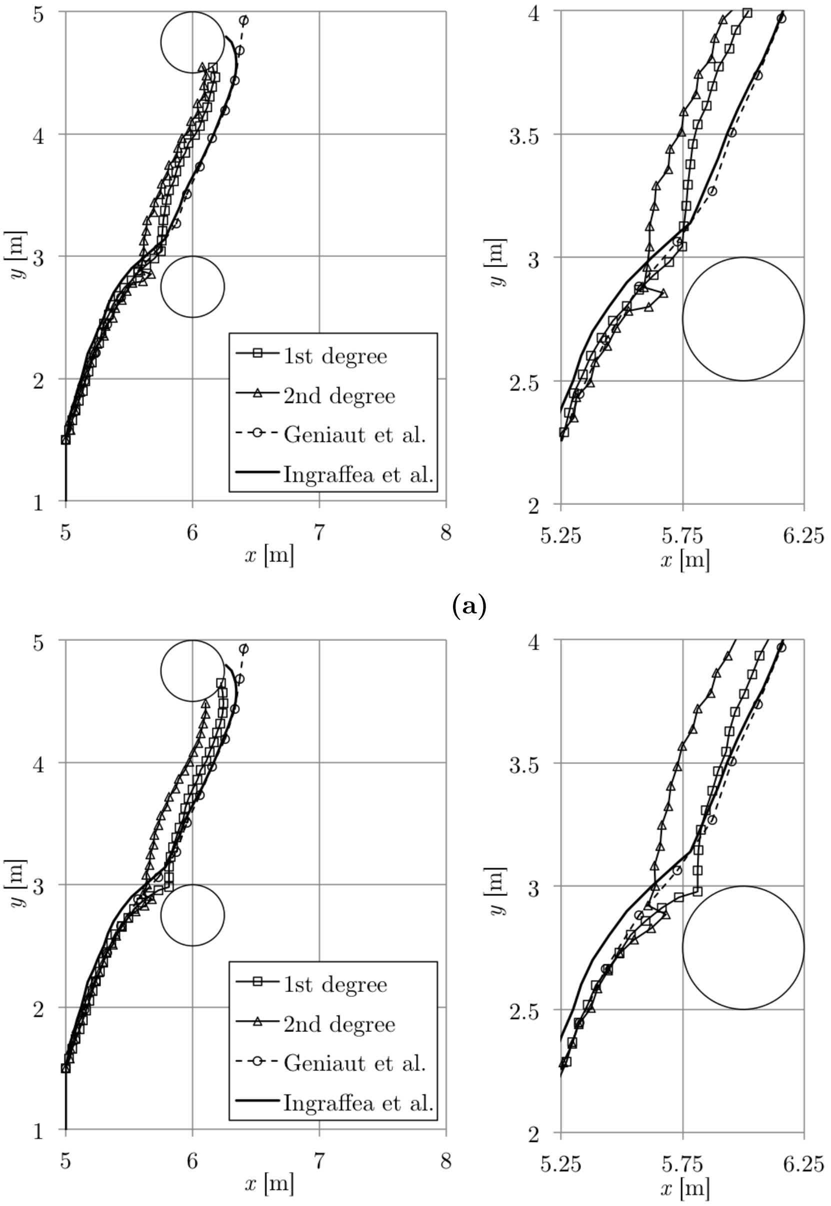
Figure 4.44: Crack path prediction for example 6, **for various** r_s: (a) three lower values; (b) three higher values. $\Delta a = h$.

results. In this case, the crack ends in the lower hole and not in the middle one.

Analysing the results of this example in general, it is possible to verify that the proximity of the lower hole to the crack, creates a difficulty to the developed algorithm. Two possible causes are identified. Firstly, the proximity of the crack to the hole requires for a low value of r_s to be used. As a low number of integration points are used, the method is required to be more accurate than in the other studied numerical examples. Another cause is related to the fact that the crack is "almost tangent" in relation to the hole. This creates an area where the influence domains have to be much smaller than in the rest of the domain. As was referred in section 2.1.2, the variation of the size of the influence domains affects the performance of the method. The optimal crack path is obtained using first influence domains and is represented in figure 4.45b. The optimal values for the studied parameters are displayed in table 4.14.

Table 4.14: Optimal parameters for example 6.

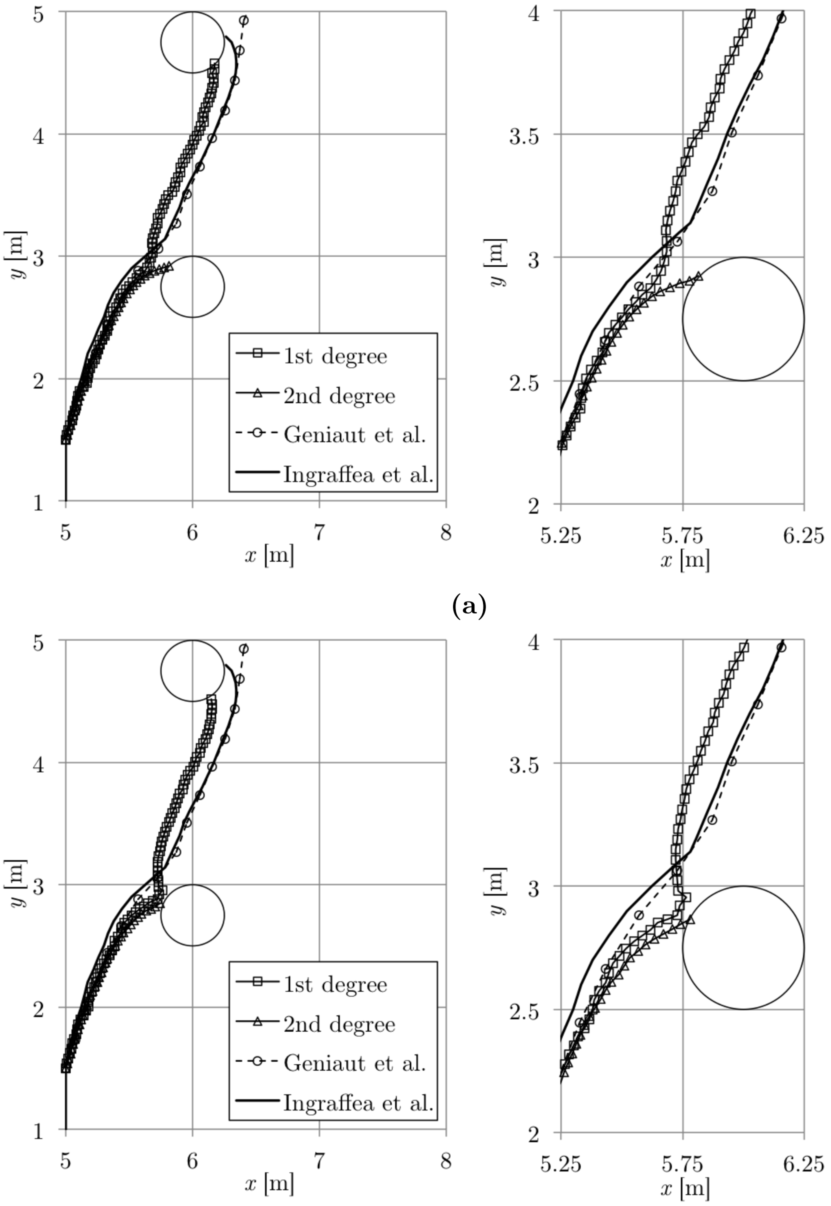
Optimal parameters (example 6)		
Crack increment	h	[m]
r_s	h	[m]
Influence domain	First degree	



(a)

(b)

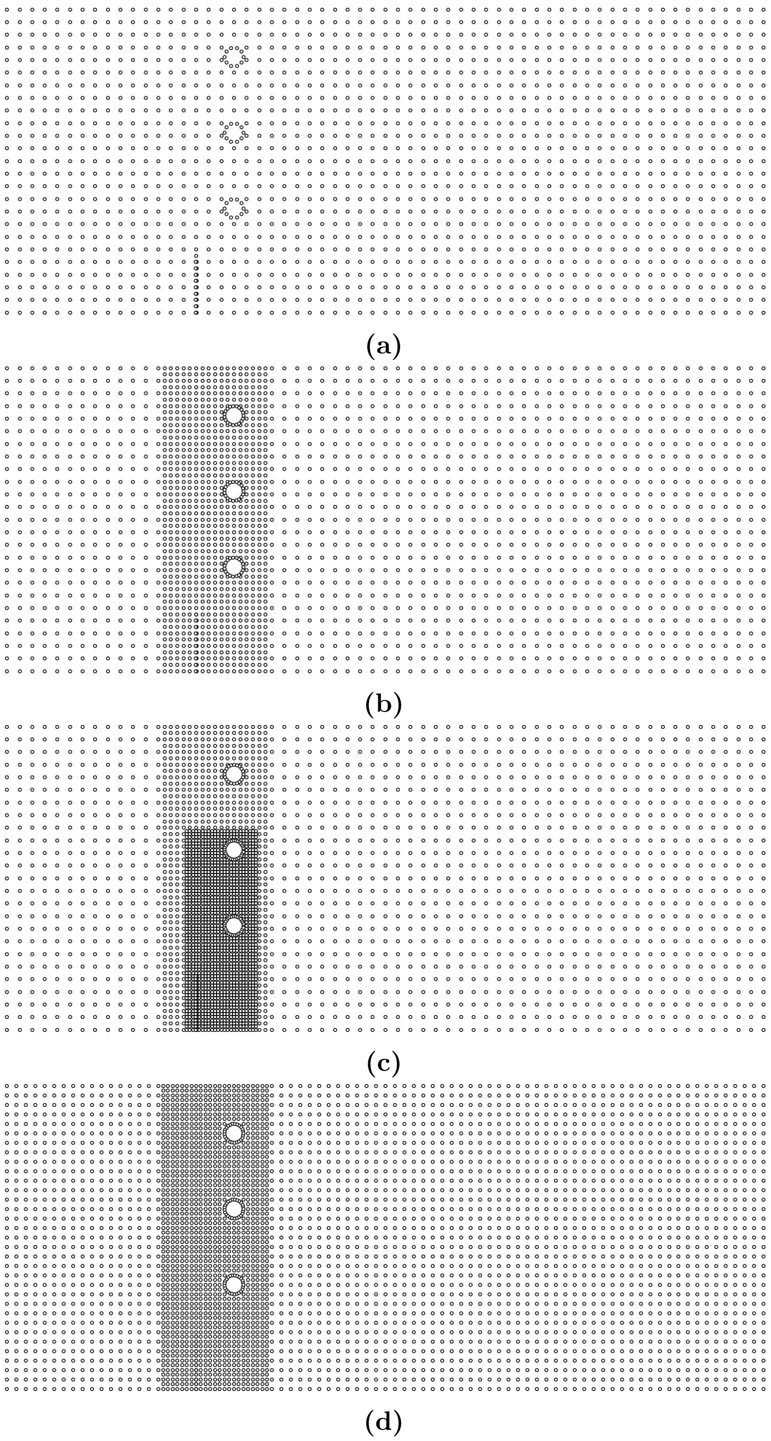
Figure 4.45: Influence domain variation: (a) $r_s = 0.75h$; (b) $r_s = h$. $\Delta a = h$.



(a)

(b)

Figure 4.46: Influence domain variation: (a) $r_s = 0.75h$; (b) $r_s = h$. $\Delta a = 0.5h$.


(a)

(b)

(c)

(d)

Figure 4.47: Nodal meshes used in example 6: (a) 61 × 25; (b) 61 × 25 (refined 1); (c) 61 × 25 (refined 2); (d) 81 × 33 (refined 1).

4.2.7 Example 7

In example 7 the crack propagation in a plate with a central inclined crack is studied. The plate is subjected to tension in the upper and lower edges. The crack is inclined in relation to the load therefore, it is subjected to mixed mode loading. In this example the crack propagates in two directions simultaneously. The problem is represented in figure 4.48 and in table 4.15 the additional relevant data is shown.

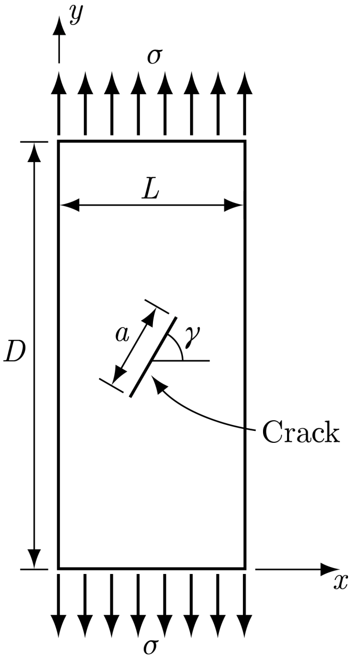


Figure 4.48: Schematic representation of the problem of the example 7.

The nodal meshes used in this example are shown in figure 4.54. This is a complex problem and the nodal discretisation in the area of the crack is crucial to the accuracy of the results. In this problem, the nodal mesh in figure 4.54c has four times the nodal density near the crack than in the rest of the domain. In this example, second degree influence domains are used, except when otherwise mentioned. The results obtained in this work are compared with the results obtained experimentally by Pustejovsky in [51]. The results were obtained with fatigue loading in Titanium. The crack path predicted using static loading and LEFM is comparable to the experimental results, Rao et al. [32] made the same simplifications in solving this problem. The crack path predicted by Rao et al. is similar to the experimentally determined one hence, is not shown in the figures.

Table 4.15: Relevant data regarding example 7.

Relevant data (example 7)		
σ	5	[Pa]
E	1000	[Pa]
ν	0.29	
L	0.3048	[m]
D	0.0762	[m]
Thickness	0.05	[m]
γ	60	[°]
a	0.0142	[m]
divl	96	
divd	320	

Nodal mesh variation

In order to verify the convergence of the results the number of nodes used in the analysis was sequentially increased. The various crack path predictions are displayed in figure 4.49. The results were obtained with a crack increment equal to the nodal distance near the crack.

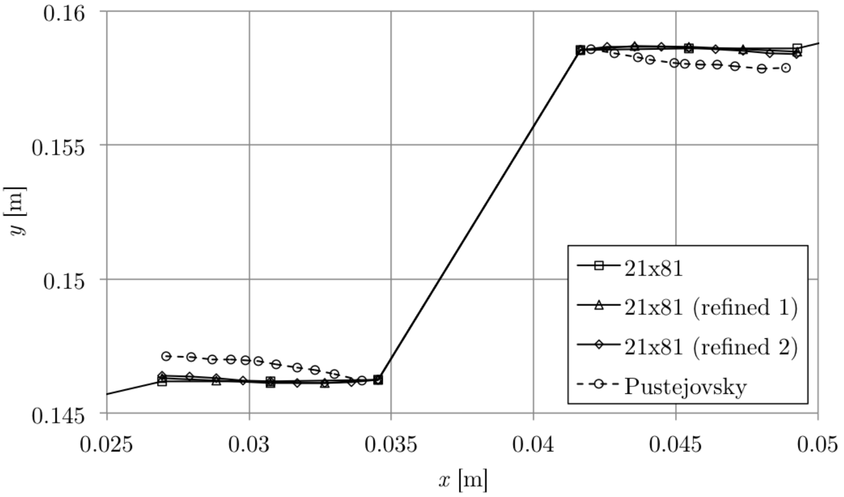


Figure 4.49: Crack path prediction for example 7, **for various nodal meshes**. $r_s = 2h$.

As expected, the less refined mesh yielded the less accurate results. Using the 81×21 (refined 1) and the 81×21 (refined 2) nodal meshes, similar results are obtained. In the remaining of this example the 81 × 21 (refined 2) is used. As this is a complex problem, the most refined mesh possible is used.

Crack increment variation

In order to study the influence of the crack increment in the crack path, several simulations were performed with different values of that parameter. The results are displayed in figure 4.50. Analysing the figure, it is possible to conclude that all values yield an accurate crack path prediction. In the case of the conditions in figure 4.50, the crack path is not sensitive to the value of the crack increment. That may be explained by the fact that the nodal mesh is quite fine near the crack, and r_s is relatively large in this case.

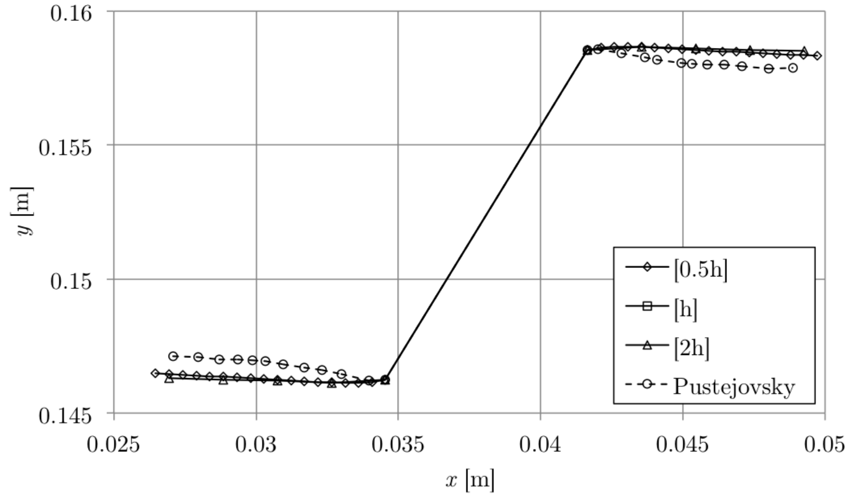


Figure 4.50: Crack path prediction for example 7, **for various crack increments**. $r_s = 2h$.

r_s variation

In figure 4.51 the crack paths for different values of r_s are shown. r_s has a significant influence in the crack path prediction. It can be verified that the obtained crack path approximates the reference crack path as r_s lowers. When $r_s = 0.375h$, that tendency is no longer verified. That may be caused by the fact that not enough integration points are selected. The optimal value for this example is $r_s = 0.5h$.

Influence domain variation

The effect of the influence domain in the crack path is studied in conjunction with the variation of Δa and r_s.

Analysing the effect of the influence domain variation, **in conjunction with** r_s, figure 4.53 shows that the influence domain has a significant effect on the results. The variation of the type of influence domain has a more

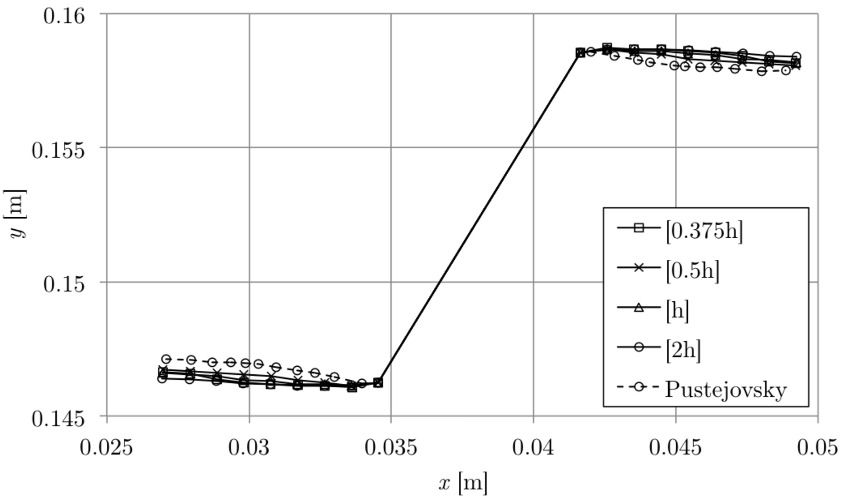


Figure 4.51: Crack path prediction for example 7, **for various values of** r_s. The crack increment is h.

significant effect in the results when r_s is shorter. This was verified in previous examples. For $r_s = 0.5h$, using first degree influence domains significantly degrades the accuracy of the results.

Analysing the effect of the influence domain variation, **in conjunction the crack increment**, the results in figures 4.53a and 4.52 are compared. For a shorter crack increment, the variation of the type of influence domain is less significant. Analysing those figures it is also possible to conclude that the optimal value for the crack increment is h.

The optimal crack path is represented in figure 4.53a (using second degree influence domains). The optimal values are summarised in table 4.16.

Table 4.16: Optimal parameters for example 7.

Optimal parameters (example 7)		
Crack increment	h	[m]
r_s	$0.5h$	[m]
Influence domain	Second degree	

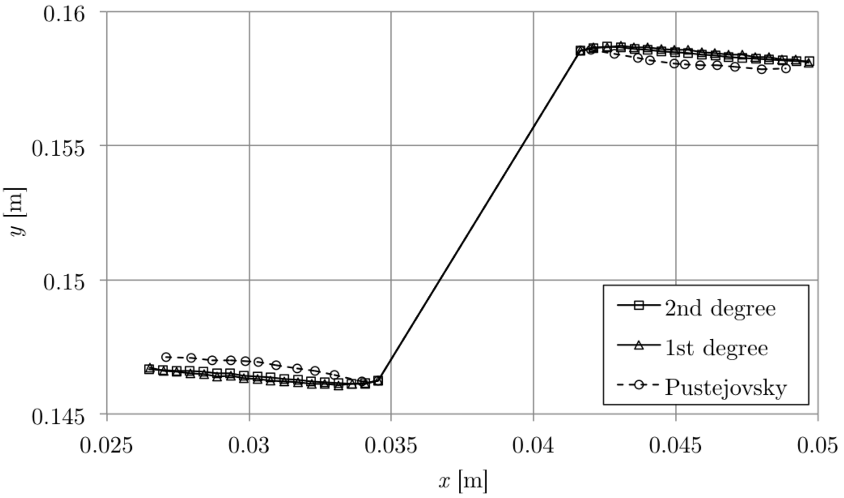
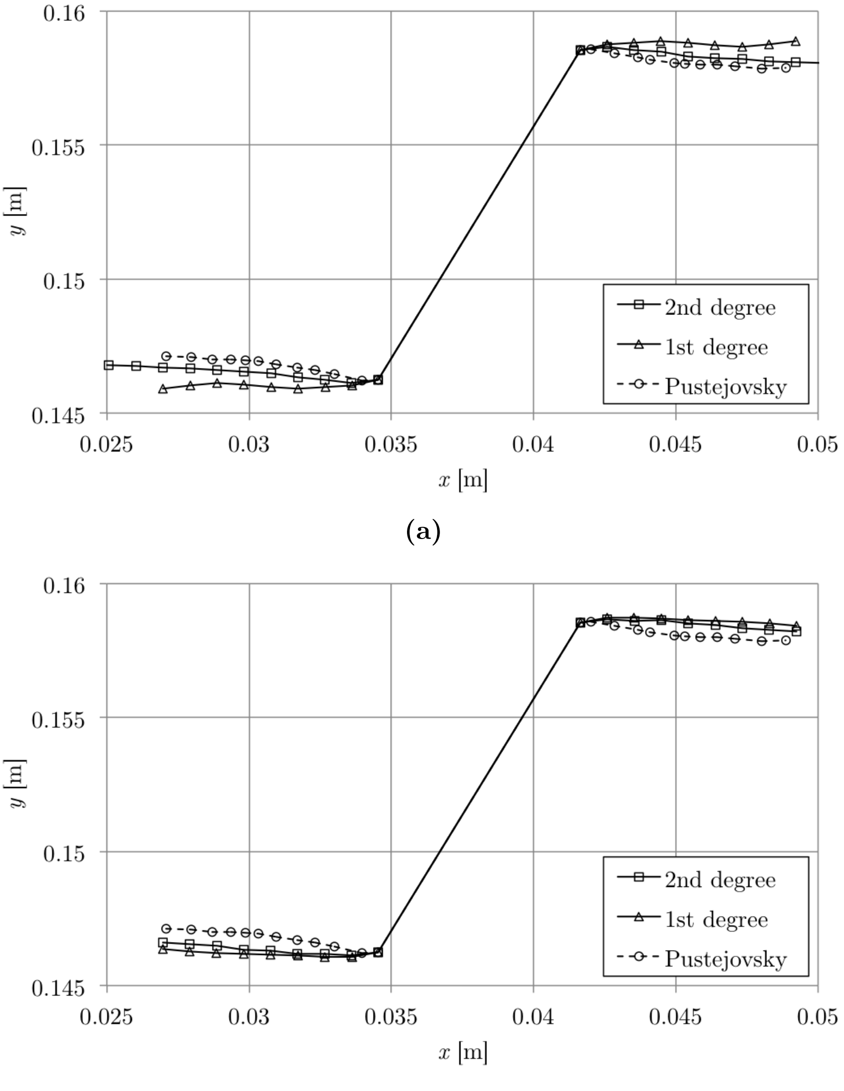


Figure 4.52: **Influence domain variation**, the crack increment is equal to $0.5h$ and $r_s = 0.5h$.



(a)

(b)

Figure 4.53: Influence domain variation, the crack increment is equal to h: (a) $r_s = 0.5h$; (b) $r_s = h$.

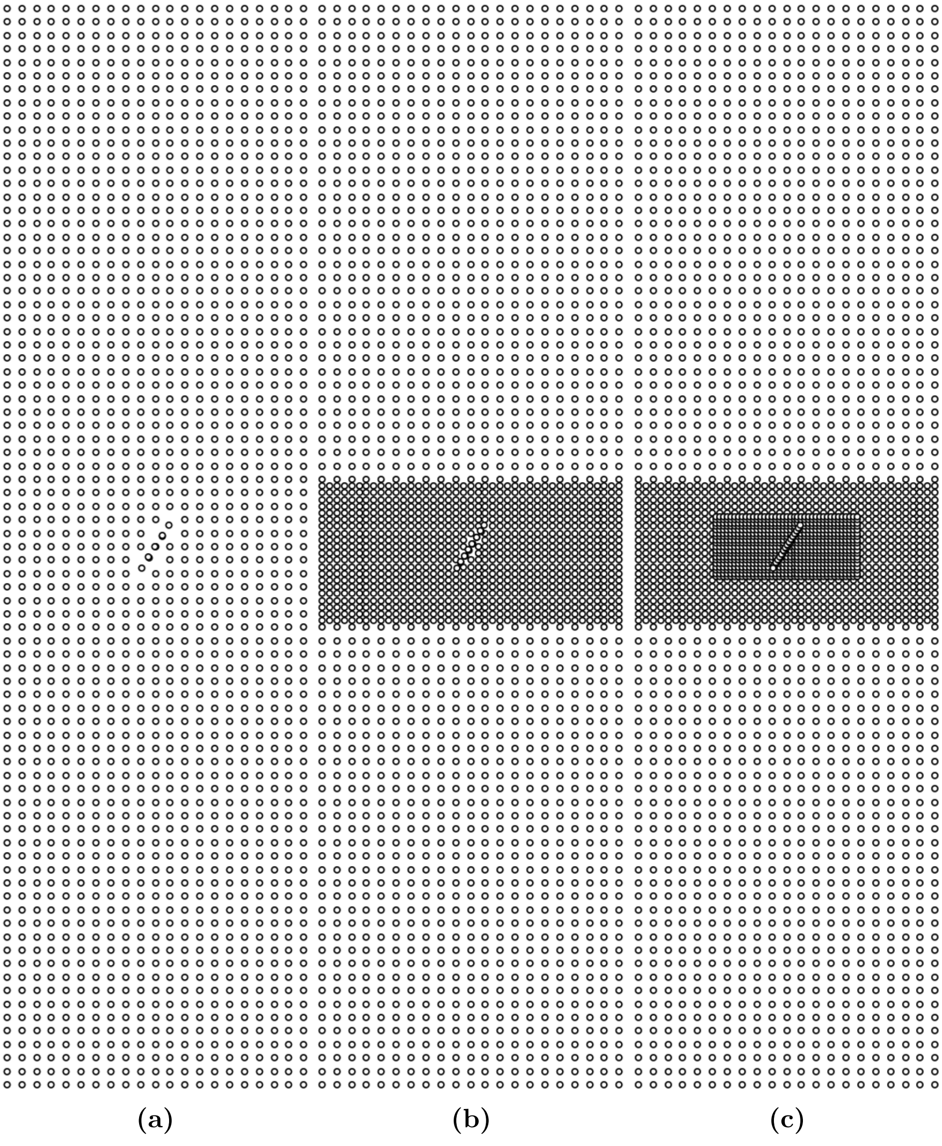


Figure 4.54: Nodal meshes used in example 6: (a) 81 × 21 (1702 nodes); (b) 81 × 21 (refined 1) (2357 nodes); (c) 81 × 21 (refined 2) (2935 nodes).

Chapter 5

Conclusion and future work

In this work, an automatic way of simulating crack growth using a meshless method was developed. The developed program uses the stress field to calculate the direction of crack propagation, which is perpendicular to the principal stress direction. The program sequentially increases the crack length to obtain the final crack path. At each iteration of the code, the crack's propagation direction is calculated and the crack is extended in straight line segments.

The most relevant conclusions are:

1. The concept of the natural neighbour indeed provides an organic and simple form of implementing discontinuities in a continuous domain.

2. The NNRPIM is a solid platform that can be used to solve fracture mechanics problems. The existence to the Kronecker delta property, the integration mesh being completely nodal dependent and the fact that the NNRPIM provides a standardised nodal dependent way of creating the influence domains are some of the most important characteristics relevant to the development of the fracture mechanics algorithm.

3. The fact that meshless methods only use nodes to discretise the problem's domain, makes the remeshing to accommodate the crack's extension a simple and computationally economic task.

4. In the crack path prediction problem, three parameters have a significant influence in the results, the crack increment Δa, the radius of selection of integration points, r_s and the type of influence domain. The effect of these parameters in the crack path was analysed extensively in this thesis, the main conclusions, regarding each one, are:

   - The crack increment affects the crack path because it determines the number of nodes that discretise the crack. Nodal discretisation has a significant effect in the computation of the test functions. If the crack increment is overly short or long, the exaggerated nodal distance variations are responsible for low quality test functions, resulting in inaccurate results. In this work it was found that the value of crack increment should be between a certain range. It was also found that, as long as the crack increment does not cause inaccurate test functions, its value should be minimised in order to improve the accuracy of the results (improving the discretisation of the crack).

- r_s is a parameter that ultimately defines the number of integration points used to calculate the crack's propagation direction. If this value is too large, the obtained values for the stresses at the crack tip are not an accurate representation of the actual stress state in that area. If r_s is too small, inaccurate oscillations appear in the crack path. This is caused by the fact that not enough integration points are selected to correctly compute the stresses near the crack tip. In general, a value close to the nodal spacing provided the most accurate results.
- Dinis et al. [1] stated that using second, rather than first, degree influence domains in the NNRPIM provided more accurate results. In fracture mechanics problems the use of second degree influence domains did not always provide the most accurate result. It was verified that, when the crack propagates near the boundaries of the problem, the use of second degree influence domains leads to less accurate results. It was the case of example 4 and 6 where the crack path is near a hole in the solid. That occurrence is caused by a the variation of the size of the influence domains. Near the boundaries of the problem, the influence domains are smaller. When using second degree influence domains, that difference is more significant, causing less accurate results. When the crack path is not near any other boundary, the use of second degree influence domains resulted in a more accurate crack path prediction.

5. The influence of the above mentioned parameters in the results cannot be considered to be independent, i. e. the variation of a given parameter, originates different variations of the results, for different values of other parameters.

One of the aims of this work was to find a set of optimal parameters that would allow for an uniform fracture mechanics analysis of a given component or structure. The optimal values for each example are displayed in table 5.1. Considering the obtained results and all de corresponding analysis, it is possible to conclude that the optimal value for the crack increment is in the range of $0.5h$ to h, r_s is in the range of $0.5h$ to $1.5h$, it is not possible to choose one type of influence domain.

Table 5.1: Optimal values for the studied parameters.

Example	Δa	r_s	**Influence domain**
2	$0.5h$	$1.5h$	Second degree
3	h	$1.5h$	First degree
4	$0.5h$	$0.75h$	First degree
5	h	$2h$	First degree
6	h	h	First degree
7	h	$0.5h$	Second degree

The present work is an introductory application of the NNRPIM to fracture mechanics. One major development of this work would be to eliminate completely the influence of the crack increment and r_s in the crack path prediction. The following areas of application represent possible and interesting developments for the work done in this thesis:

- Consider material non-linearity (plasticity): a significant part of the materials used in engineering exhibit plastic behaviour. To extend the developed algorithm beyond Linear-elastic fracture mechanics, would create the possibility of studying such materials.

- Analyse three-dimensional problems: The extension of the code developed in this thesis to three dimensions would create the possibility of studying the crack path on any structure or component.

- Include new fracture criterions: besides the maximum tangential stress criterion, there are others, more complex, that provide a more general analysis. It would be interesting to verify if the increase in complexity is justified by an increase in the accuracy of the results.

- Consider dynamics and fatigue loading: considering the effects of dynamics and variable loads would create the possibility of predicting the life cycle of a given solid. It would also be interesting to verify what is the effect of dynamic loads in the crack path.

Ultimately, a numerical tool capable of accurately predicting the life cycle of a given component with a wide range of possible loading conditions and engineering materials, could be developed from this work.

Bibliography

[1] L.M.J.S. Dinis, R.M.N. Jorge, and J. Belinha. Analysis of 3d solids using the natural neighbour radial point interpolation method. *Computer Methods in Applied Mechanics and Engineering*, 196(13-16):2009 – 28, 2007.

[2] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. Meshless methods: an overview and recent developments. *Computer methods in applied mechanics and engineering*, 139(1):3–47, 1996.

[3] Vinh Phu Nguyen, Timon Rabczuk, Stephane Bordas, and Marc Duflot. Meshless methods: A review and computer implementation aspects. *Mathematics and Computers in Simulation*, 79(3):763 – 813, 2008.

[4] T.J.R. Hughes. *The finite element method: linear static and dynamic finite element analysis.* Dover Civil and Mechanical Engineering Series. Dover Publications, 2000.

[5] J.G. Wang and G.R. Liu. A point interpolation meshless method based on radial basis functions. *International Journal for Numerical Methods in Engineering*, 54(11):1623–1648, 2002.

[6] L.B. Lucy. A numerical approach to the testing of the fission hypothesis. *The astronomical journal*, 82:1013–1024, 1977.

[7] Y.Y. Lu, T. Belytschko, and L. Gu. A new implementation of the element free galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 113(3):397–414, 1994.

[8] B. Nayroles, G. Touzot, and P. Villon. Generalizing the finite element method: diffuse approximation and diffuse elements. *Computational mechanics*, 10(5):307–318, 1992.

[9] S.N.A.S. Shen. The meshless local petrov-galerkin (mlpg) method: a simple & less-costly alternative to the finite element and boundary element methods. 2002.

[10] G.R. Liu and Y.T. Gu. A point interpolation method for two-dimensional solids. *International Journal for Numerical Methods in Engineering*, 50(4):937–951, 2001.

[11] H. Lin and S.N. Atluri. Meshless local petrov-galerkin(mlpg) method for convection diffusion problems. *CMES(Computer Modelling in Engineering & Sciences)*, 1(2):45–60, 2000.

[12] G. Voronoï. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. deuxième mémoire. recherches sur les parallélloèdres primitifs. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 1908(134):198–287, 1908.

[13] B. Delaunay. Sur la sphere vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, 7(793-800):1–2, 1934.

[14] K.M. Liew, X. Zhao, and A.J.M. Ferreira. A review of meshless methods for laminated and functionally graded plates and shells. *Composite Structures*, 93(8):2031–2041, 2011.

[15] A. Ortiz, M.A. Puso, and N. Sukumar. Maximum-entropy meshfree method for compressible and near-incompressible elasticity. *Computer Methods in Applied Mechanics and Engineering*, 199(25):1859–1871, 2010.

[16] C.T. Wu, C.K. Park, and J.S. Chen. A generalized approximation for the meshfree analysis of solids. *International journal for numerical methods in engineering*, 85(6):693–722, 2011.

[17] Jorge Américo Oliveira Pinto Belinha. *The natural neighbour radial point interpolation method : solid mechanics and mechanobiology applications.* PhD in mechanical engineering, Universidade do Porto. Faculdade de Engenharia, 2010.

[18] L. Dinis, R.M. Natal Jorge, and J. Belinha. Analysis of plates and laminates using the natural neighbour radial point interpolation method. *Engineering analysis with boundary elements*, 32(3):267–279, 2008.

[19] L. Dinis, R.M. Jorge, and J. Belinha. The natural neighbour radial point interpolation method: dynamic applications. *Engineering Computations: Int J for Computer-Aided Engineering*, 26(8):911–949, 2009.

[20] L. Dinis, R.N. Jorge, and J. Belinha. Radial natural neighbours interpolators: 2d and 3d elastic and elastoplastic applications. *Progress on Meshless Methods*, 2008.

[21] L. Dinis, R.M. Natal Jorge, and J. Belinha. Large deformation applications with the radial natural neighbours interpolators. *Computer Modeling in Engineering and Sciences (CMES)*, 44(1):1, 2009.

[22] M.H. Aliabadi and D.P. Rooke. *Numerical fracture mechanics*, volume 8. Springer, 1991.

[23] Y. Sumi. Computational crack path prediction. *Theoretical and applied fracture mechanics*, 4(2):149–156, 1985.

[24] S. Boljanović and S. Maksimović. Analysis of the crack growth propagation process under mixed-mode loading. *Engineering Fracture Mechanics*, 78(8):1565–1576, 2011.

[25] J. Dolbow and T. Belytschko. A finite element method for crack growth without remeshing. *International journal for numerical methods in engineering*, 46(1):131–150, 1999.

[26] C.T. Wu, Y. Guo, and E. Askari. Numerical modeling of composite solids using an immersed meshfree galerkin method. *Composites Part B: Engineering*, 2012.

[27] Samuel Geniaut and Erwan Galenne. A simple method for crack growth in mixed mode with x-fem. *International Journal of Solids and Structures*, 49:2094–2106, 2012.

[28] T.P. Fries and M. Baydoun. Crack propagation with the extended finite element method and a hybrid explicit–implicit crack description. *International Journal for Numerical Methods in Engineering*, 2012.

[29] S. Mariani and U. Perego. Extended finite element method for quasi-brittle fracture. *International Journal for Numerical Methods in Engineering*, 58(1):103–126, 2003.

[30] H. Nguyen-Xuan, G.R. Liu, N. Nourbakhshnia, and L. Chen. A novel singular es-fem for crack growth simulation. *Engineering Fracture Mechanics*, 2012.

[31] G. Ventura, J.X. Xu, and T. Belytschko. A vector level set method and new discontinuity approximations for crack growth by efg. *International Journal for Numerical Methods in Engineering*, 54(6):923–944, 2002.

[32] B.N. Rao and S. Rahman. A coupled meshless-finite element method for fracture analysis of cracks. *International Journal of Pressure Vessels and Piping*, 78:647–657, 2001.

[33] M. Duflot and H. Nguyen-Dang. A meshless method with enriched weight functions for fatigue crack growth. *International Journal for Numerical Methods in Engineering*, 59(14):1945–1961, 2004.

[34] M. Duflot. A meshless method with enriched weight functions for three-dimensional crack propagation. *International Journal for Numerical Methods in Engineering*, 65(12):1970–2006, 2006.

[35] S.C. Li and Y.M. Cheng. Enriched meshless manifold method for two-dimensional crack modeling. *Theoretical and applied fracture mechanics*, 44(3):234–248, 2005.

[36] F. Erdogan and G.C. Sih. On the crack extension in plates under plane loading and transverse shear. *Journal of basic engineering*, 85:519, 1963.

[37] S. Timoshenko and J.N. Goodier. *Theory of elasticity*. McGraw-Hill classic textbook reissue series. McGraw-Hill, 1969.

[38] J.F.S. Gomes. *Mecânica dos Sólidos e Resistência dos Materiais*. Edições INEGI, 2004.

[39] H.L. Ewalds and R.J.H. Wanhill. *Fracture mechanics*. Edward Arnold, 1984.

[40] S.A. Meguid. *Engineering fracture mechanics*. Elsevier Applied Science, 1989.

[41] G.R. Irwin. Analysis of stresses and strains near the end of a crack traversing a plate. *J. appl. Mech.*, 1957.

[42] G.C. Sih. Strain-energy-density factor applied to mixed mode crack problems. *International Journal of Fracture*, 10(3):305–321, 1974.

[43] C.H. Wu. Fracture under combined loads by maximum-energy-release-rate criterion. *ASME, Transactions, Journal of Applied Mechanics*, 45:553–558, 1978.

[44] P. Krysl and T. Belytschko. Element-free galerkin method: Convergence of the continuous and discontinuous shape functions. *Computer methods in applied mechanics and engineering*, 148(3):257–277, 1997.

[45] D. Organ, M. Fleming, T. Terry, and T. Belytschko. Continuous meshless approximations for nonconvex bodies by diffraction and transparency. *Computational Mechanics*, 18(3):225–235, 1996.

[46] M. Fleming, Y.A. Chu, B. Moran, T. Belytschko, Y.Y. Lu, and L. Gu. Enriched element-free galerkin methods for crack tip fields. *International Journal for Numerical Methods in Engineering*, 40(8):1483–1504, 1997.

[47] T. Belytschko, Y.Y. Lu, and L. Gu. Crack propagation by element-free galerkin methods. *Engineering Fracture Mechanics*, 51(2):295–315, 1995.

[48] S. Geniaut and E. Galenne. A simple method for crack growth in mixed mode with x-fem. *International Journal of Solids and Structures*, 2012.

[49] T.N. Bittencourt, P.A. Wawrzynek, A.R. Ingraffea, and J.L. Sousa. Quasi-automatic simulation of crack propagation for 2d lefm problems. *Engineering Fracture Mechanics*, 55(2):321–334, 1996.

[50] A.R. Ingraffea and M. Grigoriu. Probabilistic fracture mechanics: a validation of predictive capability. Technical report, DTIC Document, 1990.

[51] Micheal A. Pustejovsky. Fatigue crack propagation in titanium under general in-plane loading-i: Experiments. *Engineering Fracture Mechanics*, 11:9–15, 1979.