NTNU – Trondheim
Norwegian University of
Science and Technology

NUMERICAL SIMULATION OF FATIGUE CRACK GROWTH

Stine Vethe

Preface and Acknowledgments

This is the master thesis by stud.techn. Stine Vethe at Norwegian University of Science and Technology, Spring 2012 as a partially fulfillment of the Master of Science degree in mechanical engineering with specialization in materials and structural integrity.

Using this opportunity, I would like to thank those who made this thesis possible.

I owe my deepest gratitude to my supervisors, Professor Gunnar Härkegård, for his support and guidance from the start of this thesis.

Ph.D. Candidate Ali Cetin, for his encouragements, guidance and always being available for question.

Several people at 4Subsea AS, for their sharing of knowledge and valuable insights in giving the thesis a industrial touch.

Viggo Røneid, my colleague and "office mate", for his patience, support and interesting discussions.

Lastly, my parents who always have supported and inspired me.

Stine Vethe

Trondheim, 11 June 2012

Contents

Nomenclature

γ_W Curve fitting parameter (Walker model)

$\hat{\theta}$ Crack propagation angle

μ Shear modulus

ν Poisson's ratio

$\overline{\Delta K}$ Equivalent zero-to-tension stress intensity

σ_y Yield strength

C_W, m_W Parameters in the Walker model

E Young's modulus

E_{eff} Effective Young's modulus

G Energy Release Rate

K_c Fracture toughness

K_I, K_{II}, K_{III} mode I, II and III stress intensity factor (SIF)

R Stress ratio

u	Displacement
W	Strain energy density
API	American Petroleum Institute
FCG	Fatigue Crack Growth
FEA	Finite Element Analysis
GUI	Graphical User Interface
LEFM	Linear Elastic Fracture Mechanics
MCS	Maximum Circumferential Stress criterion
MCS	Maximum Circumferential Stress criterion
MERR	Maximum Energy Release Rate criterion
MERR	Maximum Energy Release Rate criterion
MTS	Maximum Tangential Stress criterion
MTS	Maximum Tangential Stress criterion
SIF	Stress Intensity Factor
UTS	Ultimate Tensile Strength
XFEA	Extended Finite Element Analysis
XFEM	Extended Finite Element Method

THE NORWEGIAN UNIVERSITY
OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF ENGINEERING DESIGN
AND MATERIALS

MASTER THESIS SPRING 2012
FOR
STUD.TECHN. STINE VETHE

NUMERICAL SIMULATION OF FATIGUE CRACK GROWTH
Numerisk modellering av utmattingssprekkvekst

Fatigue crack growth is a typical reliability concern in most engineering components under cyclic loading. Due to complex geometry and loading, it is often necessary to simulate crack growth numerically. In this thesis, the focus shall be on advanced non-conventional methods such as XFEM (the extended finite element method) and weight functions (as implemented in the FEA post-processor P•FAT).

The aim of this study should be to explore the possibilities and challenges offered by these methods in fatigue-crack growth (FCG) simulations. Suggested partial tasks are:
- Carry out a (limited) literature search in order to identify practicable FCG models (with mixed-mode capability).
- Develop a procedure for 2D FCG simulations by means of XFEM in Abaqus, preferably by means of scripting. Carry out simulations with XFEM for a (sharply) notched component to be specified, e.g., a wellhead or a threaded connection.
- Carry out limited 3D simulations with XFEM and P•FAT for the component above.

The thesis should include the signed problem text, and be written as a research report with summary both in English and Norwegian, conclusion, literature references, table of contents, etc. During preparation of the text, the candidate should make efforts to create a well arranged and well written report. To ease the evaluation of the thesis, it is important to cross-reference text, tables and figures. For evaluation of the work a thorough discussion of results is appreciated.

Three weeks after start of the thesis work, an A3 sheet illustrating the work is to be handed in. A template for this presentation is available on the IPM's web site under the menu "Under-visning". This sheet should be updated when the Master's thesis is submitted.

The thesis shall be submitted electronically via DAIM, NTNU's system for Digital Archiving and Submission of Master's thesis.

Contact persons are Ali Cetin, IPM, and Vidar Osen, LINKftr.

Torgeir Welo
Head of Division

Gunnar Härkegård
Professor/Supervisor

NTNU
Norges teknisk-
naturvitenskapelige universitet

Institutt for produktutvikling
og materialer

Abstract

Summary (English)

The purpose of this study was to explore the possibilities and challenges with simulating fatigue crack growth (FCG) by the extended finite element method (XFEM). Another aim was to develop a procedure for XFEM FCG simulations in Abaqus by means of scripting. Finally was the procedure used to simulate FCG in an API standard, cone shaped threaded connection.

Different FCG models were reviewed by a limited literature search and a procedure 2D FCG simulations was carried out by a python script. The procedure succeeded with the simulation of FCG when applied to a model with refined mesh around the crack tip.

In the suggested partial tasks of the thesis description were a procedure 3D FCG simulation also suggested, but as this required more computer capacity than available in the study this was not carried out.

Sammendrag (Norwegian)

Målet med denne masteroppgaven var å utforske mulighetene og utfordringene med å simulere utmattingssprekkvekst (FCG) ved hjelp av den utvidede elementmetoden (XFEM). Et av delmålene var deretter å utvikle en prosedyre for FCG simuleringen i Abaqus ved hjelp av et skript. Til

slutt ble prosedyren benyttet til å simulere FCG i en standard API konet gjengekobling .

Et begrenset littersøk ble utført på FCG modeller med mulighet til å modelere kombinasjoner av flere brudd-moder (engelsk: mixed-mode capability), og en prosedyre for 2D simulering av FCG ved hjelp av et python-skript utviklet og testet. Så lenge elementnettet rundt sprekkspissen er kraftig forfinet, fungerer prosedyren som den skal.

En av de foreslåtte deloppgavene i oppgavebeskrivelsen var å i tillegg utvikle en prosedyre for 3D simulering av FCG. På grunn av begrenset tilgang til datakapasitet var ikke dette praktisk mulig å gjennomføre.

Chapter 1

Introduction

In engineering components are fatigue failure a imporatant consideration which may lead to fatal consequences. Fatigue failure has been subject a significant research by engineers in more than 150 years. Nucleation and propagation of cracks caused by repeated cyclic loading below the yield stress of a material may cause catastrophic failure. The increased awareness of fatigue failures the last 50 years is a consequence of many sudden tragic accidents. Some of the most well-known and often mentioned accidents within fatigue failure are probably the disintegration of the two de Havilland Comet jetliners during flights in 1954, and the capsizing of the Alexander L. Kielland oil platform due to a poor weld reducing the fatigue strength of the structure.

Analyzing and designing against fatigue failure may be divided into three major approaches [4]; the stress-based approach, the strain-based approach and the fracture mechanics approach. The stress-based approach consider the nominal (average) stresses in the affected areas in the analysis of the fatigue life. The strain-based approach treats the localized plastic deformations/yielding that may occur in regions with stress raisers as edges and notches. The last approach, the fracture mechanics approach involves analyzing crack growth by using the methods of fracture mechanics. This is the
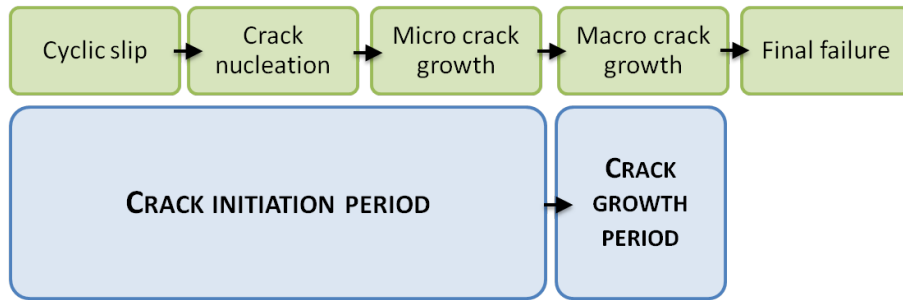
Figure 1.1: Different periods of the fatigue life [2]

approach which this master thesis is based on.

The fracture mechanics approach are further divided into a lifetime approach (number of cycles to failure) or a fatigue crack growth approach (damage tolerance). Fatigue crack growth (FCG) is defined as crack growth caused by repeated cyclic loading [4].

Since the 1950s many researchers have investigated how early cracks may be detected in fatigue life and divided the fatigue life in two periods/phases; the crack initiation period and the crack growth period [2]. Under high-cycle fatigue may the crack initiation period be the largest period of the fatigue life, while for low-cycle fatigue may the crack growth period be the most significant phase. One of the difficulties with this definition may be how to specify the transition between the two phases, from micro to macro crack growth 1.1. To define the transition between crack initiation period and crack growth period (illustrated in 1.1) Paris et al. proposed in 1961 a correlation between crack growth rate, da/dN , and the stress intensity factor range, ΔK.

1.1   Problem statement

There are today many different approaches and methods to perform lifetime prediction calculations on components, but the procedures are either time

consuming, demand high computer capacity or have low accuracy.

Inadequate knowledge about fatigue crack growth and the existing procedures are either demanding with a high degree of complexity, i. e. conventional FEA (finite element analysis), or give too conservative results, i.e. the result give a lifetime of two years while the component has already lasted for five years with any indication of being close to failure.

In this thesis a procedure for FCG with a focus on the extended finite element method (XFEM) is developed. The XFEM is a relatively new method which has been developed during the last decade. The method has many possibilities since it does not need remeshing when the crack propagate. With a procedure for using this method in Abaqus, the simulation of FCG may be significantly easier to carry out.

1.2   Limitations

Due to the scope of the thesis to carry out a *limited* literature search are only a few models reviewed in this thesis. In the literature search many different models were studied, but as many of the models are not relevant or rarely used, only the most common are review in this thesis. Beden et al. (2009) have studied a significant amount of models both for constant and variable amplitude loading and written a extensive review of many FCG models [2].

Due to limited access to computer capacity, this thesis cannot provide a procedure for three-dimensional numerical simulations. This is further stated in section 3.4. The procedure is based on the assumption that the results from the preliminary study of the XFEM implementation in ABAQUS version 6.10 also is valid for version 6.11.

The analyses in this study are executed on a computed laptop with an Intel core i5 processor and with only 8GB RAM. Consequently the number of elements in the models could not exceed 300,000 to keep the computation time reasonably low. With this restriction the computation time for the

3

simulations of the threaded connection in chapter 4 was kept to 8-9minutes per increment ($9\,{}^{\mathrm{min}}\!/_{\mathrm{increment}} \cdot 150\,\mathrm{increments} \approx 22\,\mathrm{hours}$). If the number of elements exceed 300,000, it can take hours per increment, and are not possible within the time frame of the thesis.

Chapter 2

Background and literature review

2.1 Fracture mechanics and fatigue crack growth models

FCG models are empirical models generally based on fracture mechanics developed to describe data from experiments by empirical curve fitting parameters on the form[2]:

$$\frac{da}{dN} = f(\Delta K, R) \tag{2.1}$$

Irwin (1958) introduced the stress intensity factor (SIF) K for static fracture analysis 2.2, after his analysis of the stress field around the crack tip in 1957 [2].

$$K = F\sigma\sqrt{\pi a} \tag{2.2}$$

where a is the crack length and F is the geometry factor which depends on the relative crack length $\alpha = a/b$.

In 1961 proposed Paul C. Paris a concept of using a simple empirical equation (2.3) to apply linear elastic fracture mechanics (LEFM) to fatigue [10].

Paris' law describes fatigue crack growth rate, da/dN, and is today the most common model to use for FCG analysis.

$$\frac{\mathrm{d}a}{\mathrm{d}N} = C_P \left(\Delta K\right)^{m_P} \tag{2.3}$$

$$\Delta K = F \Delta S \sqrt{\pi a} \tag{2.4}$$

where C_P and m_P are constants which depend on the material, frequency of the cycles and the environment, and ΔS is the stress range. This model is further reviewed in section 2.3.

2.2 Fatigue Rate Curve

The fatigue rate curve is a $\frac{da}{dN}$ versus ΔK curve as shown in figure 2.1. The curve is often divided into three regions; Region I, Region II and Region III.

In region I is the early development of the fatigue crack represented, and the growth rate is in the order $10^{-6}mm/cycle$ or below. This region is very sensitive to micro structure features like grain size, the means stress of the applied load, the environment and the operating temperature[2]. The most important feature in this region is the FCG threshold, ΔK_{th}. This is the limit for the propagation of fatigue cracks start. For SIF ranges below the ΔK_{th} crack growth will usually not occur [4].

Region II is the intermediate zone for growth rates in the order of 10^{-6} to $10^{-3}mm/cycle$. In this region the crack growth is stable, the data is following a power equation and the plastic zone in front of the crack tip is large compared to the mean grain size [2]. Since the data is following a power equation, the fit will be linear on a log-log plot (figure 2.1) and the use of LEFM concepts are applicable. For region II the mean stress has the highest influence on the results, but the influence is small compared with region I.
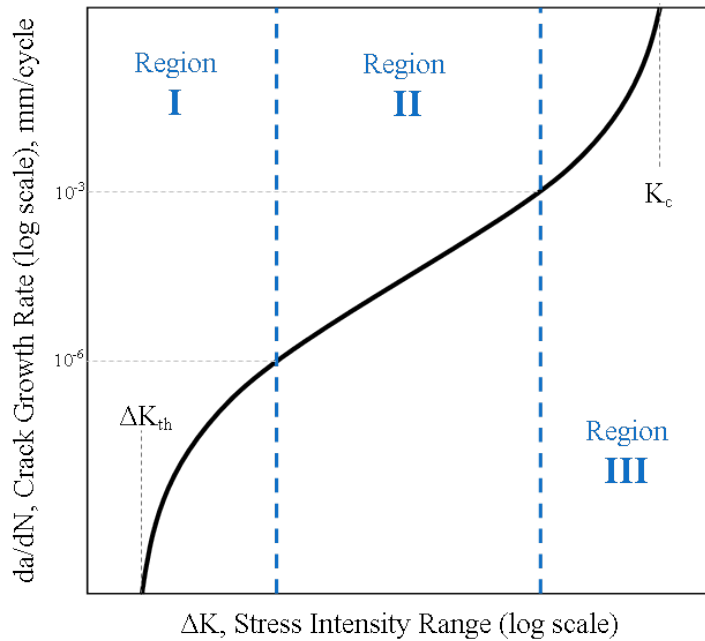
Figure 2.1: Fatigue rate curve

The last region, region III, starts where the curve (figure 2.1) again become steep, and is usually in the order $10^{-3}mm/cycle$ and above. This is high crack growth rates caused by the rapid unstable growth prior to final failure. The curve approaches in this region an asymptote corresponding to the fracture toughness,K_c, for the material. The influence of nonlinear properties cannot be ignored in this region due the present of large scale yielding. This results in that LEFM cannot be used for the data in this region and nonlinear fracture mechanics should be applied instead. FCG analysis for this region is therefore very complex and as the FCG rates are high and little fatigue life involved. This results often in that analysis for this region are ignored.

2.3   Review of fatigue crack growth models

Engineering components are rarely exposed to constant amplitude loading, but rather variable amplitude. To decrease calculation time and simplify the

cases may the variable amplitude loading be modified and simplified in order to use constant amplitude models with fewer parameters. In FCG models are the parameters often empirical.

2.3.1 Paris Model

The FCG model which is the most common used in fatigue analysis today is the power law introduced by Paris and Erdogan (1961)[10] is the Paris model, also known as the Paris law 2.3. The model is simple in use and only needs two curve fitting parameters to be determined. In the descriptions of data in region II (*figure 2.1*)–are Paris law widely used, but region I and region III cannot be sufficiently described by this model. Other limitations of the Paris law are that the model does not account the effect of stress ratio, the results depend upon the material used and most important Paris law are only for pure mode I loading. Paris' law can be modified to be applicable to mixed mode loading by the use of equivalent SIF, further described in section 2.5. In the common used standard BS7910:2005 [1] and the "Wellhead Fatigue Analysis Method" by DNV are Paris' law the suggest model to use in fatigue analysis. For obtaing correct results from Paris' model is it important to collect the fatigue rate data from the related stress ratio.

2.3.2 Walker Model

As a wish to improve the Paris model proposed Walker (1970) [2]a model which includes the effect of stress ratio, R. The model is given by the following relationship:

$$\frac{da}{dN} = C_W \left(\overline{\Delta K}\right)^{m_W} = C_W \left(\frac{\Delta K}{(1-R)^{1-\gamma_W}}\right)^{m_W} \tag{2.5}$$

$$\Delta K = K_{max}\left(1-R\right)$$

8

where the constants C_W and m_W are similar to the constants in Paris model (equation (2.3)) C_P and m_P. The parameter $\overline{\Delta K}$ is an equivalent zero-to-tension stress intensity that causes the same growth rate as the actual K_{mac}, R combination. The third curve fitting parameter, γ_W, is a constant for the material. This parameter may be obtained from data of various R values, linear regression or trial and error. It is possible that there is no value to be found for γ_W, then the Walker model cannot is not valid. If $\gamma_W = 1$ then $\overline{\Delta K}$ equals ΔK and the stress ratio has no effect on the data.

2.3.3   Forman Model

When the SIF approach the critical value will there be a instability of the crack growth (region III, figure2.1), neither Paris' or Walker's models take this into account. The Forman expression is given by the following relationship:

$$\frac{da}{dN} = \frac{C_F \left(\Delta K\right)^{m_y}}{\left(1 - R\right) K_c - \Delta K} = \frac{C_F \left(\Delta K\right)^{m_y}}{\left(1 - R\right) \left(K_c - K_{max}\right)} \tag{2.6}$$

where $K_c$is the fracture toughness. The use of this constant is necessary to model cracks with high growth rates. Also from the Forman model the relationship between K_{max}, K_c and $\frac{da}{dN}$ show that as the maximum SIF approaches the fracture toughness, the crack growth tends to infinity. With this can the Forman equation be used to represent data for both region II and region III (figure2.1). If crack growth data for various stress ratios are available, can these be used by computing the quantity in equation (2.7) for each data point.

$$Q = \frac{da}{dN} \left[\left(1 - R\right) K_c - \Delta K\right] \tag{2.7}$$

If the different combinations of ΔK and R all fall together on a straight line on a log-log plot of Q and ΔK, the Forman equation is assumed to be successful and may be used [4].

9

2.3.4 XFEM crack growth

A preliminary study has been performed under the title *"Modeling fatigue crack growth using XFEM in ABAQUS 6.10" [15]*, where the principle of the XFEM crack model are studied.

A variety of techniques are used to calculate the mixed mode stress intensity factor by the XFEM. Literature reviews performed by Pais [9] and Belytschko [3] identifies the procedures, the advantages and the disadvantages of the different calculation techniques. According to their prior studies is the use of the domain form of interaction integrals extracted from J-integrals the most common technique. This method has high accuracy for different crack conditions when used together with a suitable mesh [9].

In short is the domain form of the interaction integrals an extension of the J-integral where the line integral is converted to an area integral. The J-integral is used to find the energy release rate while the interaction integral is used to obtain the mixed-mode stress intensity factor.

2.3.4.1 Theory of interaction integral

Prior to obtaining the integration integrals the J-integrals need to be established.

On the basis of the relationship between the J-integral, stress intensity factors and the effective Young's modulus, E_{eff}, may the energy released rate G for a general homogenous two-dimensional mixed-mode crack be expressed as [9]:

$$G = J = \frac{K_I^2}{E_{eff}} + \frac{K_{II}^2}{E_{eff}} \tag{2.8}$$

$$E_{eff} = \begin{cases} E & \text{plane stress} \\ \frac{E}{1-\nu^2} & \text{plane strain} \end{cases} \tag{2.9}$$

where E is Young's modulus and ν is Poisson's ratio. Further are the J-integral specified as[9]:

$$J = \int_{\Gamma} \left(W n_1 - \sigma_{jk} n_j \frac{\partial u_k}{\partial x_1} \right) \, \mathrm{d}\Gamma \tag{2.10}$$

where W is the strain energy density, σ is stress and u is displacement. To ease the implementation to a finite element code should equation 2.10 be rewritten by introducing the Dirac delta to the first term of the integral.

$$J = \int_{\Gamma} \left(W \delta_{1j} - \sigma_{jk} \frac{\partial u_k}{\partial x_1} \right) n_j \, \mathrm{d}\Gamma \tag{2.11}$$

To determine the mixed-mode SIFs are auxiliary stress and displacement set superimposed onto the XFEM displacement and stress state. The auxiliary displacement and stress states at the crack tip for a homogenous crack are calculated from the definitions established by Westergaard[16]and Williams[18]:

$$
\begin{aligned}
u_1^{(a)} & \quad \frac{1}{2\mu}\sqrt{\frac{r}{2\pi}} \left[K_I \cos \frac{\theta}{2} (\kappa - \cos\theta) + K_{II} \sin \frac{\theta}{2} (\kappa + 2 + \cos\theta) \right] \\
u_2^{(a)} & = \frac{1}{2\mu}\sqrt{\frac{r}{2\pi}} \left[K_I \sin \frac{\theta}{2} (\kappa - \sin\theta) + K_{II} \cos \frac{\theta}{2} (\kappa + 2 + \cos\theta) \right] \quad (2.12) \\
u_3^{(a)} & \quad \frac{2}{\mu}\sqrt{\frac{r}{2\pi}} K_{III} \sin \frac{\theta}{2}
\end{aligned}
$$

$$\sigma^{(a)} = \begin{bmatrix} \sigma_{11}^{(a)} & \sigma_{12}^{(a)} & \sigma_{13}^{(a)} \\ & \sigma_{22}^{(a)} & \sigma_{23}^{(a)} \\ sym. & & \sigma_{33}^{(a)} \end{bmatrix} \tag{2.13}$$

$$\sigma_{11}^{(a)} = \frac{1}{\sqrt{2\pi r}} \left[K_I \cos\frac{\theta}{2} \left(1 - \sin\frac{\theta}{2} \sin\frac{3\theta}{2} \right) - K_{II} \sin\frac{\theta}{2} \left(2 + \cos\frac{\theta}{2} \cos\frac{3\theta}{2} \right) \right] \tag{2.14}$$

$$\sigma_{22}^{(a)} = \frac{1}{\sqrt{2\pi r}} \left[K_I \cos\frac{\theta}{2} \left(1 + \sin\frac{\theta}{2} \sin\frac{3\theta}{2} \right) + K_{II} \sin\frac{\theta}{2} \cos\frac{\theta}{2} \cos\frac{3\theta}{2} \right] \tag{2.15}$$

$$\sigma_{33}^{(a)} = \nu \left(\sigma_{11} + \sigma_{22} \right) \tag{2.16}$$

$$\sigma_{12}^{(a)} = \frac{1}{\sqrt{2\pi r}} \left[K_I \sin\frac{\theta}{2} \cos\frac{\theta}{2} \cos\frac{3\theta}{2} + K_{II} \cos\frac{\theta}{2} \left(1 - \sin\frac{\theta}{2} \sin\frac{3\theta}{2} \right) \right] \tag{2.17}$$

$$\sigma_{23}^{(a)} = \frac{1}{\sqrt{2\pi r}} K_{III} \cos\frac{\theta}{2} \tag{2.18}$$

$$\sigma_{13}^{(a)} = - \frac{1}{\sqrt{2\pi r}} \sin\frac{\theta}{2} \tag{2.19}$$

$$\kappa = \begin{cases} \frac{3-\nu}{1+\nu} & \text{plane stress} \\ 3 - 4\nu & \text{plane strain} \end{cases} \tag{2.20}$$

where r and θ are the polar coordinates from crack tip, μ is , ν is Poisson's ratio and κ is Kosolov constant. The superposition of the auxiliary displacement, $u_{ij}^{(a)}$, and stress, $\sigma_{ij}^{(a)}$, states and the XFEM displacement, $u_{ij}^{(x)}$, and stress, $\sigma_{ij}^{(x)}$, states into equation 2.11 gives:

$$J^{(a+x)} = \int_\Gamma \left(\frac{1}{2} \left(\sigma_{ij}^{(a)} + \sigma_{ij}^{(x)} \right) \left(\varepsilon_{ij}^{(a)} + \varepsilon_{ij}^{(x)} \right) \delta_{1j} - \left(\sigma_{ij}^{(a)} + \sigma_{ij}^{(x)} \right) \frac{\partial \left(u_i^{(a)} + u_i^{(x)} \right)}{\partial x_1} \right) n_j \, d\Gamma \tag{2.21}$$

On the basis of equation 2.21can the interaction integral be obtained by separating the integral into XFEM state $J^{(x)}$, auxiliary state $J^{(a)}$ and interaction state $I^{(a,x)}$.

$$J^{(a+x)} = J^{(a)} + J^{(x)} + I^{(a,x)} \qquad (2.22)$$

The interaction integral is given by[9]:

$$I^{(a,x)} = \int_{\Gamma} \left[W^{(a,x)} \delta_{1,j} - \sigma_{ij}^{(a)} \frac{\partial u_i^{(x)}}{\partial x_1} - \sigma_{ij}^{(x)} \frac{\partial u_i^{(a)}}{\partial x_1} \right] n_j \, \mathrm{d}\Gamma \qquad (2.23)$$

$$W^{(a,x)} = \sigma_{ij}^{(a)} \varepsilon_{ij}^{(x)} = \sigma_{ij}^{(x)} \varepsilon_{ij}^{(a)} \qquad (2.24)$$

where $W^{(a,x)}$ is the interaction strain energy density. Finally are the divergence theorem used to convert interaction integral from a line integral to an area integral[9]:

$$I^{(a,x)} = \int_{A} \left[\sigma_{ij}^{(x)} \frac{\partial u_i^{(a)}}{\partial x_1} - \sigma_{ij}^{(a)} \frac{\partial u_i^{(x)}}{\partial x_1} - W^{(a,x)} \delta_{1,j} \right] \frac{\partial q_s}{\partial x_j}_{,j} \, \mathrm{d}A \qquad (2.25)$$

where q_s is a smoothing function with a value of 1 inside the line integral and 0 (zero) outside the integral. In order to select the area for the integration integral must path-independency be considered. Normally are path-independence achieved at a radius equal to three elements about crack tip [9].

By the same principle may an interaction integral also be obtained with the basis in equation 2.8. The J-integral are then superimposed and separated to:

$$J^{(a+x)} = J^{(a)} + J^{(x)} + \frac{2 \left(K_I^{(a)} K_I^{(x)} + K_{II}^{(a)} K_{II}^{(x)} \right)}{E_{eff}} \qquad (2.26)$$

$$I^{(a,x)} = \frac{2 \left(K_I^{(a)} K_I^{(x)} + K_{II}^{(a)} K_{II}^{(x)} \right)}{E_{eff}} \qquad (2.27)$$

13

2.4   Crack growth direction

The direction of the crack propagation is established to be a function of the mixed-mode stress intensity factors at the crack tip. Their are several different criteria too choose from the calculate the direction. Some of the most widely used mixed mode criteria are: the maximum tangential stress criterion, the maximum energy release rate criterion, the zero K_{II} criterion $(K_{II} = 0)$ and maximum circumferential stress criterion.

2.4.1   The maximum tangential stress criterion (MTS)

With this criterion are the deflection angle of the crack growth defined to be perpendicular to the maximum tangential stress at the crack tip. This criterion is based on the work of Erdogan and Sih [5] and are given by:

$$\hat{\theta} = \cos^{-1}\left(\frac{3K_{II}^2 + \sqrt{K_I^4 + 8K_I^2 K_{II}^2}}{K_I^2 + 9K_{II}^2}\right) \tag{2.28}$$

where $\hat{\theta}$ is the and with respect to the crack original plane. If mode II SIF (K_{II}) is positive, the propagation angle is negative and opposite. The maximum tangential stress $\sigma_{\theta m}$ are descibed by the following relationship [17]:

$$\sigma_{\theta m} = \frac{1}{\sqrt{2\pi r}} \cos^2 \frac{\theta_m}{2} \left[K_I \cos \frac{\theta_m}{2} - 3K_{II} \sin \frac{\theta_m}{2}\right] \tag{2.29}$$

One important remark with this criterion is that the crack will not propagate in its own plane (except for pure mode I loading) but in respect to the *original* plane [17].

2.4.2 The maximum energy release rate criterion (MERR)

The MERR criterion is based on the work of Hussain *et al.* (1974) [7]. The criterion assume that the crack propagate in the direction which maximizes the energy release rate.

2.4.3 The zero K_{II} criterion $(K_{II} = 0)$-

The essence of the $K_{II} = 0$ criterion is to let the mode II SIF dissipate in shear mode for microscopic crack extensions.

2.4.4 The maximum circumferential stress criterion (MCS)

The MCS criterion is defined as the angle θ_c, given by the following relationship:

$$\theta_c = -\arccos\left(\frac{2K_{II}^2 + K_I\sqrt{K_I^2 + 8K_{II}^2}}{K_I^2 + 9K_{II}^2}\right) \qquad (2.30)$$

This is a simple and a widely used in the literature for XFEM crack growth, but as this is not an option in the ABAQUS software, is this not considered as an alternative for the FCG simulation in this thesis.

2.5 Crack growth magnitude

To calculate the magnitude of the increment at each iteration for constant amplitude loading are two methods established in the literature [9]. The first method assumes that for all the given iterations will a constant amount of crack growth increment occur. The second method assumes that a FCG law may be used to find the amount of growth for the corresponding iteration (i.e. Paris law(2.3)). While for variable amplitude loading are the methods

for constant amplitude loading no longer valid and the amount of growth need to be calculated separately for each cycle.

For both cases can the SIF ranges be defined by the maximum and minimum SIF for each mode within each cycle:

$$\Delta K_I = K_{I,max} - K_{I,min} \tag{2.31}$$
$$\Delta K_{II} = K_{II,max} - K_{II,min} \tag{2.32}$$

In order to determine the FCG for mixed mode loading by Paris law is an equivalent stress intensity factor required. To calculate the equivalent SIF are several different models proposed.

Tanaka [14] proposed a relationship for the equivalent SIF based on curve fitting data. The is given as

$$\Delta K_{eq} = \sqrt[4]{\Delta K_I^4 + 8\Delta K_{II}^4} \tag{2.33}$$

A relationship based on the energy release rate is the equivalent SIF defined as [9]:
$$\Delta K_{eq} = \sqrt{\Delta K_I^2 + \Delta K_{II}^2} \tag{2.34}$$

From the maximum circumferential stress criterion are relationship is the following expression given as [9]:

$$\Delta K_{eq} = \frac{1}{2}\cos\left(\frac{\theta}{2}\right)[\Delta K_I(1 + \cos\theta) - 3\Delta K_{II}\sin\theta] \tag{2.35}$$

The modified Paris law is given as:

$$\frac{\mathrm{d}a}{\mathrm{d}N} = C_P(\Delta K_{eq})^{m_P} \tag{2.36}$$

By integrating eq (2.36) are a function to determine number of cycles N

needed for å crack to propagate from a initial crack a_0 to a crack length a:

$$N(a) = \frac{1}{C_p} \int_{a_0}^{a} \left(\frac{1}{\Delta K_{eq}} \right)^{m_p} \mathrm{d}a \qquad (\text{for } a \geq a_0) \qquad (2.37)$$

Chapter 3

Procedure for numerical simulation of FCG

The FCG phenomen represents a major concern among engineers and is subject to significant research. Engineering components are often subjected to three-dimensionel stresses, and mixed mode fracture are realistic assumptions.

3.1   Implementation of XFEM in ABAQUS 6.11

The XFEM was first introduced in Abaqus 6.9 in 2009 [12]and the implementation has been significantly developed and improved since that. The implementation give a solution for modelling crack growth without remeshing and without requiring the mesh to match the geometry of the crack.

In this thesis is ABAQUS 6.11 used [13], as this is the latest version of ABAQUS available at NTNU in the beginning of the semster. As there still is a significant amount of limitations in the implementation of XFEM in ABAQUS 6.11 are some limitations necessary to be highlighted. One major limitation is the selection of elements, as only first-order solid con-

tinuum stress/displacement elements and second-order stress/displacement tetrahedron elements can be used in combination with XFEM. Contour integrals found by XFEM can only be evaluated in first-order brick elements, first-order tetrahedron and second-order tetrahedron [13]. Another major limitation is the differences in the implementation for analysing two- and three-dimensional models. For three-dimensional cases are both stationary and propagating crack possible to simulate, while for two-dimensional cases are only propagating cracks implemented.

A limited study of the implementation of XFEM in ABAQUS 6.10 was executed as a preliminary study the autumn before this thesis was started [15]. The results from the study show that the mesh around the crack need to be relatively fine. The mesh size h should not exceed 3% of the crack size to get acceptable results.

Further has there been discovered during the process of making a procedure for FCG in this thesis that ABAQUS gives SIFs twice the analytical answer. So all the SIFs collected by the script have to be divided by a factor 2 to get the correct answers.

3.2 Procedure for 2D FCG simulation

To simulate two-dimensional FCG in ABAQUS is a python script developed (Appendix A).

The buildt-in functions in ABAQUS do not provide the function of simulating a propagating crack and get J-integral or SIF output. ABAQUS 6.11 has the choices either simulating stationary three-dimensional crack to get the output or simulating crack growth in two-dimensions. In order to avoid remeshing is XFEM a prefered choice of crack modeling, but to collect SIFs from ABAQUS is also the contour integral crack functions used. The crack tip is defined as a contour integral while the crack surface is a XFEM crack. The traction separation properties for the material are set to a infinite high maximum

principal stress. As a result of the high maximum principal stress the crack is locked from propagation and mixed-mode SIFs may be obtained from the output database file.

XFEM is developed with the aim to be a mesh independent method, but this is not the reality with the implementation in Abaqus. In order to get accurate results the mesh size has to be below 3% of the crack size [15]. With a coarse mesh the calculated crack propagation directions become too large and the crack will propagate in an oscillating pattern until the crack has grown and the mesh size is below 3% of crack size. The limitation affects the initial crack size as this is dependent of the mesh. This can be partly solved by creating different partitions with different mesh size along the expected crack path. The results of have a higher degree of inaccuracy in the transistion area between the different mesh sizes, but if this is acceptable will this decrease the computation time significantly.

The procedure to simulate two-dimensional FCG are given as:

1. Create (or import) the uncracked model in ABAQUS

2. Prepare the model for standard static analysis with assigning materials, create steps, apply loads and boundary conditions etc.

3. Create mesh with fine mesh in the area for crack growth. It is prefered to use first-order brick elements.

4. Create a set in the assembly called "*crackedPart*" contaning the part to be cracked

5. Complete the script, by filling in the necessary inputs.

6. Make sure the script and the .cae-file are saved in the same folder

7. Execute the script from the Abaqus Command window:

   (a) Open the folder containing the files

   (b) Write "*abaqus cae noGUI=XFEMscript.py*", where *XFEMscript.py*

is the name of the script and *noGUI* indicates running the script
without the graphical user intergace (GUI)

8. Collect the results from *XFEMCrackResults_ modelName_ analysisIdentity.txt*

9. The results may then be further processed in i.e. MATLAB or MS
EXCEL

3.2.1   Dependency to mesh

XFEM is supposed to be a mesh independent method, but this is not the
reality when used in Abaqus. In order to get accurate results the mesh size
has to be below 3% of the crack size [15]. With a coarse mesh the calculated
crack propagation directions become too large and the crack will propagate
in an oscillating pattern until the crack has grown and the mesh size is below
3% of crack size.

3.2.2   Unexpected error 193

In the process of creating the procedure a bug in Abaqus were discovered. If
the script are executed from the GUI, often Abaqus crashes and shuts down
with the error message "*Unexpected error 193*". This is probably a bug in
the GUI, as the same error never occur when the script are executed from
Abaqus command window without the GUI. Some search has been done to
find the reason for the error message, however without success. The error
code is neither mentioned in the dosumentation [13] or in any academic online
forums.

3.3   Procedure tested on reference geometries

In order to control the procedure were two tests executed for two of the refer-
ence geometries from the preliminary study [15]; infinite plate with internal

through thickness crack and semi-infinite plate with edge notch.

The test for FCG in a two-dimensional infinite plate with internal through thickness crack gives good results 3.1. The crack propagates as expected and the SIFs obtained by the script have a very high accuracy. From the graph in figure 3.1D are three bumps observed. These are the results from the transition between fine and coarser mesh.

The semi-infinite plate results were compared with analytical SIF from Murakami [8]. The results, given in figure 3.2, differ a little from the handbook solution. In the preliminary study [15]was the transition depth between shallow crack and deep crack calculated to be $a = 1.12$ mm for this type of notch. The results from the script give at the most a deviation of 10% from Murakami in the area around the transition depth. Otherwise the SIFs are in good agreement with a deviation of 3-4%. This is the same results as reported in the preliminary study.

3.4 Procedure for 3D FCG simulation

In the objective of the thesis are one of the suggested partial tasks to establish a procedure for numerical simulation of 3D FCG. To simulate three-dimensional FCG demand much more computer capacity than what is available in this study. For instance simulation of the threaded connection in chapter 4; approx. 250,000 elements were generated for the 2D case. 3D simulations of this threaded connection demands millions of elements generated to get satisfying results. This is one of the drawbacks with XFEM. In conventional FEM symmetry and antisymmetry in models are used to decrease the number of DOFs and increase the accuracy of the analysis. The enrichment functions restrain the possibility of employing this in extended finite element analyses (XFEA). Consequently XFEA often demand many elements. The discussion will be either remeshing and a acceptable number of elements or no remeshing and a high amount of elements.
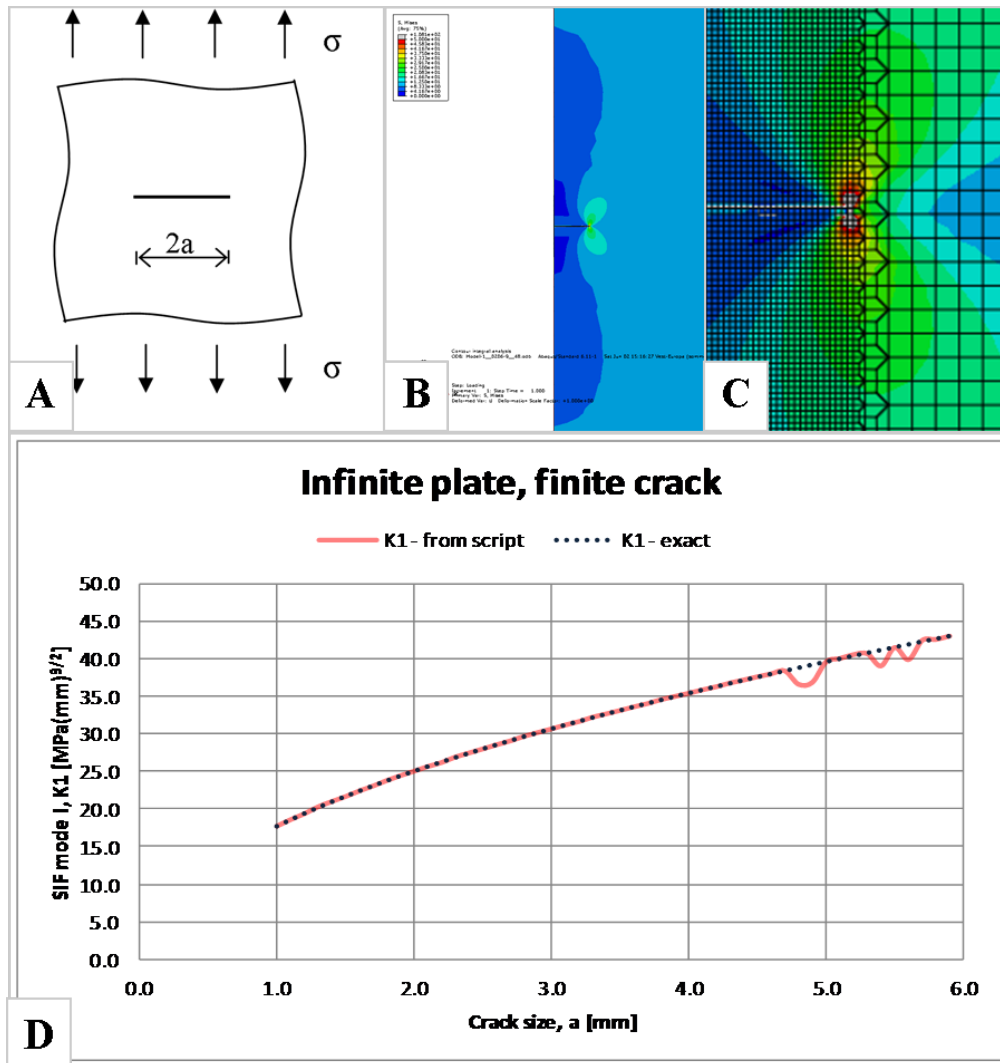
Figure 3.1: Results from using the procedure to simulate FCG in infinite plate with internal crack: A: The infinite plate with an internal through thickness crack; B: The stress distribution around the crack; C: The stress distribution when the crack tip is about to grow into the transistion area between two different mesh sizes; D: Graphical illustration of the results. Red line is the K_I obtained from the script and black dots are the exact K_I.
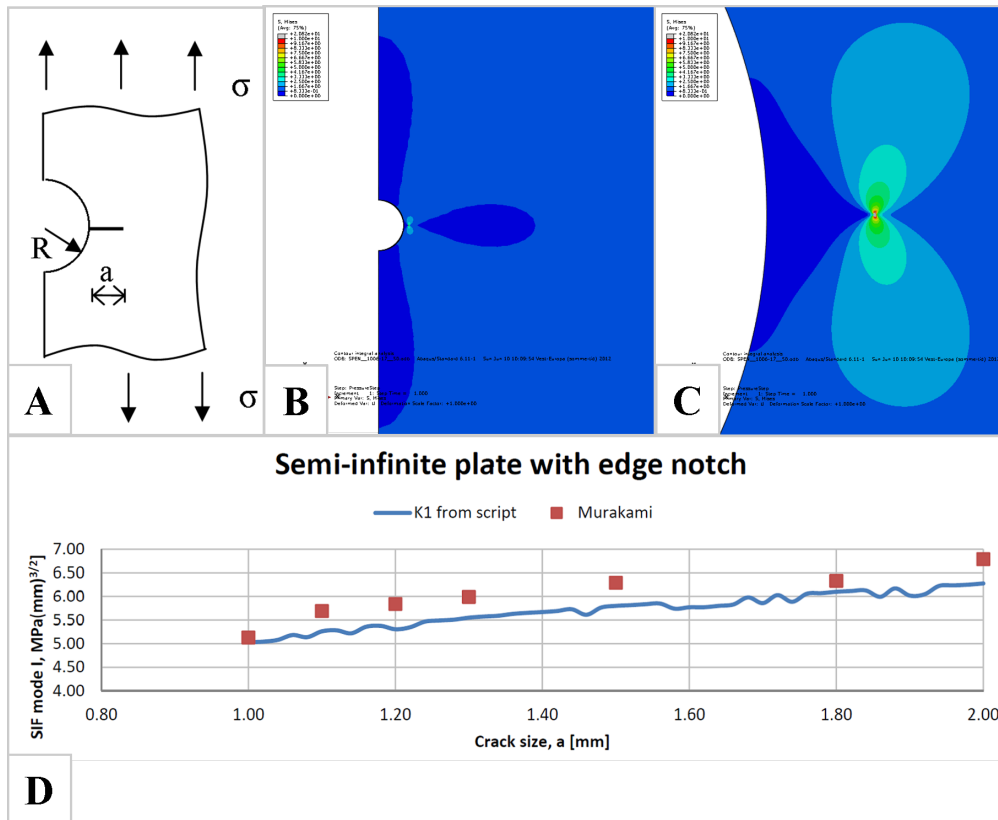
Figure 3.2: Results from using the procedure to simulate FCG in semi-infinite plate with edge notch: A: The semi-infinite plate with an edge notch; B: The stress distribution around the crack; C: The stress distribution around the crack tip; D: Graphical illustration of the results. Red dots is the K_I obtained from Murakami [8] and blue line is the K_I obtained from the script.

Previous studies have reported small differences in results from two- and three-dimensional FCG simulation.

Chapter 4

FCG Simulation of threaded connection

4.1 Threaded connections in subsea installations

Threaded connections are a reliable and cost efficient method for joining pipes compared to welded joints, and are extensively used in risers and drill pipe string in offshore applications [11]. The threads are cone shaped API (American Petroleum Institute) standard threads consisting of a pin (male part) and a box (female part). Because of the cone shape, the coupling are easily perform with only a minor twist rotation.

Threaded connections in risers and drill pipe strings are often subjected to cyclic loading due to waves and movements of the connected oil platforms. Consequently is fatigue one of the most important failure modes in such components. The design of the threaded connections is therefore a compromise between sealing capacity and resistance to fatigue failure.

4.2 Simulation of FCG in a threaded connection

The simulation of FCG for a threaded connection is here described with a independent analysis report in order to make it easy to follow.

The analysis is based on constants given in the standards BS7910:2005 [1] and chapter 6 of "*Wellhead Fatigue Analysis Method*" [6]. From BS7910 are the FCG threshold value ΔK_{th} given as:

$$\Delta K_{th} = 63 \text{N/mm}^{3/2} \tag{4.1}$$

The Paris' constants C_p and m_p may also be obtained from the same standards.

4.2.1 Geometry

The geometry of the model used in the analysis is based on the design of the machine drawings provided by 4Subsea AS. The measurements had to be modified and some assumptions to be taken to make it fit when sketched in ABAQUS.

Previous studies have shown few contradictions in the results between two-dimensional and three-dimensional FCG simulations of threaded connections. Therefore a simplification from a 3D- to 2D-model is made to save computer capacity. The simplification should have been to a 2D axis symmetric model, however this is not implemented for XFEM in ABAQUS and could not be done.

The geometry consists of two parts, a box and a pin. The threads has a cone shape as illustrated in figure 4.1.
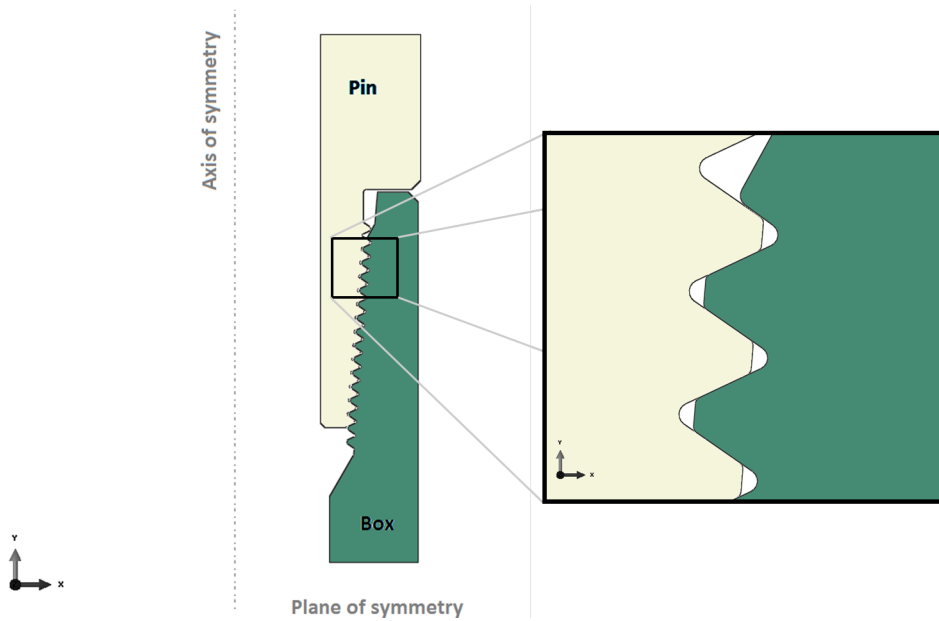
Figure 4.1: Geometry of threaded connection

4.2.2 Material

The material used in the analysis are the API steel grade B [11]. The material properties for this material are listed in table 4.1.

Yield strength, σ_y	UTS (23% elongation)	Young's modulus, E	Poisson's ratio, ν
356 MPa	575MPa	208 GPa	0.3

Table 4.1: Material properties for API steel grade B

4.2.3 Boundary conditions

The model is locked in y-direction at the bottom and the whole model is locked in x-direction to obtain a axis symmetric geometry. The boundary conditions are shown in figure 4.2 with an orange color.
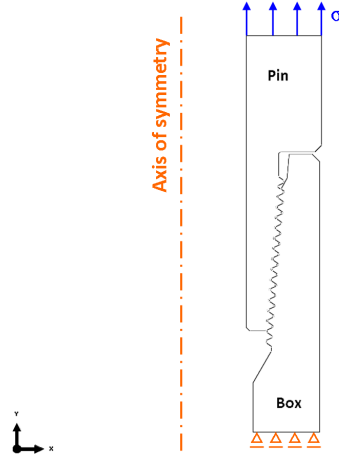
Figure 4.2: Boundary conditions (orange) and loadings (blue)

4.2.4 Loads

A constant tension load F_N and a table of bending moments M_B as a function of significant wave height and period was given by 4Subsea AS. The bending moments are all in the interval 75-80 kNm, however the differences in the moments are small and can be described with only one simple value. The loads is given as:

$$
\begin{aligned}
F_N &= 75kN \\
M_B &= 80kNm
\end{aligned}
\tag{4.2}
$$

The applied stress is superposed two calculated as the maximum applied stress σ by super positioning maximum bending stress σ_b and the tensile stress σ_N:

$$
\sigma = \sigma_N + \sigma_B = \frac{F_N}{A} + \frac{M_B r}{I} = 127.1 MPa
\tag{4.3}
$$

where r is the outer radius of the box, A is the cross-sectional area of the box and I is the second moment of inertia of the cross-section of the box.
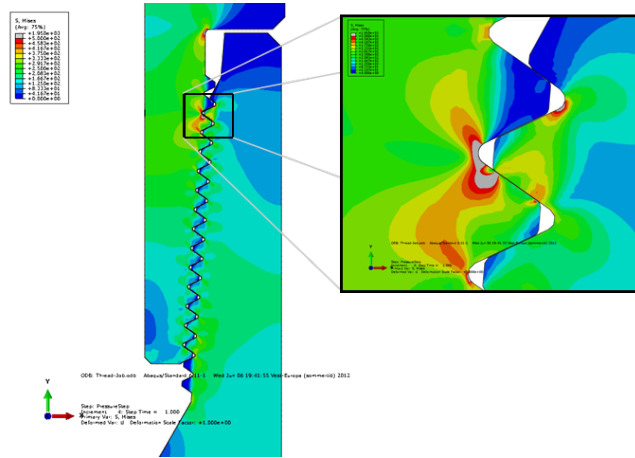
Figure 4.3: Stresses from a static analysis of threaded connection

4.2.5   Analysis

The analysis are performed according to the procedure described in chapter 3. In order to get accurate results the mesh is divided into many partitions, the section around the crack has a mesh size of 3% of the initial crack.

The initiation of the crack was located in the area with highest stress. The knowledge about the location of fatigue crack initiation and propagation is inadequate. The initiation is *often* located where the highest stress concentration occurs; at the root of the last threads with full connection (Figure 4.3). In the analysis a initial crack of 0.5mm was initiated at the root of the last thread on the pin. This is close to where the highest stress occurred in the static analysis without crack.

A second analysis was also executed with a initial crack of 0.1mm. This had a very limited crack path as a consequence of the limits for number of elements.
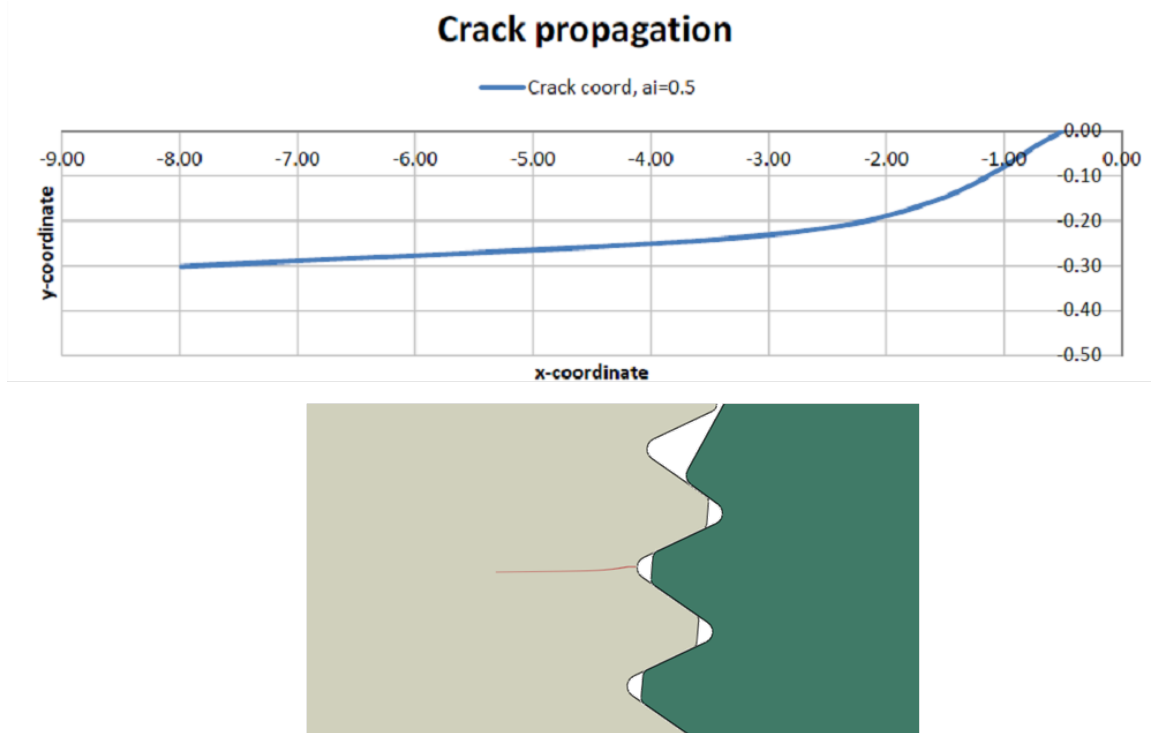
Figure 4.4: Propagated crack in threaded connection

4.2.6 Results

Running the script on the model of threaded connection gave results in the form of the text file, but Abaqus was not able to show the stress distribution around the crack in the GUI. However the crack propagated as illustrated in figure 4.4. As the highest stress occur a little lower (figure 4.3) than where the crack was initiated, the crack propagates in the direction of the highest stress. The limited analysis with initial crack of 0.1mm follows the same path as the initial crack of 0.5mm except it starts to change direction at a crack length of 0.3mm.

The graph of the SIFs 4.5 shows decreasing SIFs as the stress also decrease and then stabilize. The threshold are in BS7910 [1] given as $63^{N}/mm^{3/2}$. The simulation show that the combination of the applied stress and the geometrical stress concentration cause SIFs above the threshold value for cracks up

31

Figure 4.5: SIF mode I vs crack size

to $a = 0.69\,\text{mm}$.

4.2.7 Conclusion

The simulation obtained a crack path as expected, and the crack propagated in the direction of the highest stress. These results show that the required accuracy of the initial crack location is low as long as it is close to the highest stress.

Chapter 5

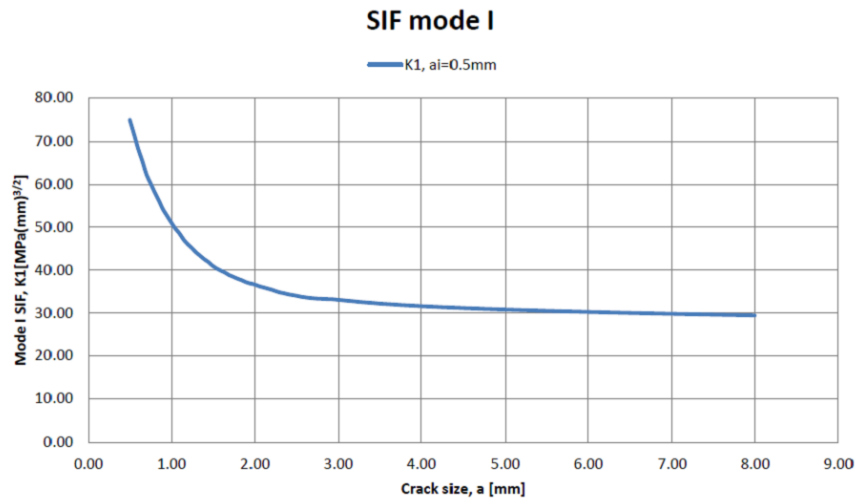# Conclusion

Using XFEM to simulate FCG give new posibilities to the execution of the analysis, but are also challenging. The method are still a relative new method, consequently few people know and trust the method and there is still a long way to go on the implementation in commercial software. On the other hand the method make the meshing less challenging and the computation time for each icrement is reduced significantly.

The procedure developed are easy to use and with few limitations. One of the biggest drawbacks with the procedure is to get the mesh correct. The requirement of a very refined mesh around the crack tip make the partitioning and meshing challenging. There is a wish to have the refined mesh area as small as possible, but without making an influence on the crack path.

5.1   Future work

In the future work the last partial task should be fulfilled by developing a procedure for three-dimensional FCG. The procedure should be developed in the same context as the one developed in this thesis.

Further should the procedure for the 2D simulations be modified to be more

robust. One of the tasks could be to include the refining of the mesh in the beginning of the script.

Bibliography

[1] BS7910 : 2005 Guide to methods for assessing the acceptability of flaws metallic structures.

[2] S. M. Beden, S. Abdullah, and A. K. Ariffin. Review of Fatigue Crack Propagation Models for Metallic Components. *European Journal of Scientific Research*, 28(3):364–397, 2009.

[3] T. Belytschko, R. Gracie, and G. Ventura. A review of extended/generalized finite element methods for material modeling. *Modelling and Simulation in Materials Science and Engineering*, 17(4):1–24, 2009.

[4] Norman E. Dowling. *Mechanical Behavior of Materials*. Pearson Education (US), 3rd edition, 2006.

[5] F. Erdogan and G. C. Sih. On the crack extension in plates under plane loading and transverse shear. *ASME, Transactions, Journal of Basic Engineering*, Series D, 1963.

[6] Torfinn Hørte. Wellhead Fatigue Analysis Method. Report for JIP Structural Well Integrity, Det Norske Veritas, 2011.

[7] M. A. Hussain, S. L. Pu, and J. H. Underwood. Strain Energy Release Rate for a Crack Under Combined Mode I and Mode II. *ASTM STP*, 560:2–28, 1974.

[8] Y. Murakami. *Stress intensity factors handbook*, volume 1. Pergamon Press, 1987.

[9] Matthew J. Pais. *Variable amplitude fatigue analysis using surrogate models and exact XFEM reanalysis*. PhD thesis, University of Florida, December 2011.

[10] P. C. Paris, M.P. Gomez, and W.P Anderson. A Rational Analytic Theory og Fatigue. *The Trend in Engineering*, 13:9–14, 1961.

[11] J. Seys, K. Roeygens, J. Van Wittenberghe, T. Galle, P. De Baets, and W. De Waele. Failure behaviour of preloaded API line pipe threaded connections. *Sustainable Construction and Design*, 2(3):407–415, 2011.

[12] Simulia. Dassault Systèmes Announces New Release of Abaqus FEA from SIMULIA, May 2009.

[13] Simulia. *Abaqus 6.11 Online Documentation*, April 2011.

[14] Keisuke Tanaka. Fatigue crack propagation from a crack inclined to the cyclic tensile axis. *Engineering Fracture Mechanics*, 6(3):493 – 507, 1974.

[15] S. Vethe. Modeling fatigue crack growth using XFEM in ABAQUS 6.10. Preliminary study, 2011.

[16] H. M. Westergaard. Bearing Pressures and Cracks. *Journal of Applied Mechanics*, 6(61):49–53, 1939.

[17] B.N. Whittaker, R.N. Singh, and G. Sun. *Rock fracture mechanics: principles, design, and applications*. Developments in geotechnical engineering. Elsevier, 1992.

[18] M. L. Williams. On the stress distribution at the base of a stationary crack. *Journal of Applied Mechanics*, 24(1):109–114, 1957.

Appendix A

Python script for simulation of 2D FCG in ABAQUS

Below are the developed python script given.

The .py file is also submitted together with the thesis in a zip-file

```python
from abaqus import*
from abaqusConstants import*
import part, material, section, assembly, step, interaction
import regionToolset, displayGroupMdbToolset as dgm, mesh, load, job

##=========================================================
## INPUT

##===  IMPORTANT  ===###
# The part with initial crack need to have a set of the whole crack part named "crackedPart"

    ## MODEL INPUT
analysisIdentity= 'myIdentity'  # Analysis indentity
caeFile= 'myCae'                # Name of the .cae file
modelName = 'myModel'           # Name of model
partName = 'myPart'             # Name of part (in assembly) containing crack
crackStep='MyCrackStep'         # Name of Step with crack initiation


NameOfMaterial='myMaterial'     # Name of material the material of the part containing crack


    ## CRACK INPUT
crackSize = 1.0                 # Initial crack size – IMPORTANT: the mesh size around the
crack cannot exceed 3% of crack size
radiusOfCrackTipNode = 0.01     # Radius of the area containg the crack tip – IMPORTANT:
Radius of the crack tip node cannot be smaller than crack size

intitialGlobalCrackCoordinates = (0.,0.,0.) # Initial global coordinates of crack initiation

    ## CALCULATION INPUT

contours=20                     # Number of contours for calculation of SIF
contourAverage = 5              # Number of contours to include for calculation of the
average of contours – IMPORTANT: has to be a positive integer


increment=20                    # Number of increments of the crack growth

incrementSize_temp =0.05        # Size of increments – extension of the crack

# END INPUT

##=========================================================
# Create tuples
crackPropagationDirection =[]
K1=[]
K2=[]
outputRes = open('XFEMCrackResults'+"_"+modelName+"_"+analysisIdentity+'.txt','a') #
Initiate report file
firstLine=['nIncrement'," ",'crackPropagationDirection',"  ",'cracktipX',"  ",'cracktipY',
"    ",'K1',"     ",'K2',"    ","\n"]
outputRes.writelines(firstLine)
outputRes.close()

mdb = openMdb(caeFile+'.cae')
myModel = mdb.models[modelName]
myAssembly = myModel.rootAssembly
```

```python
myPartInstance = myAssembly.instances[partName]


# XFEM Material properties

myMaterial=myModel.materials[NameOfMaterial]
myMaterial.MaxpsDamageInitiation(table=((1e+30, ), ))
myMaterial.maxpsDamageInitiation.DamageEvolution(table=((1.0, ), ), type=DISPLACEMENT)


#========================================================================

crackLocation=[(0, 0.),(crackSize/2, 0.),(crackSize, 0.)] # Initial crack coordinate(spline
points)
localCrack_temp = crackLocation

nIncrement= 0              #temp start

for nIncrement in range(increment):

    localCrack = tuple(localCrack_temp)

    ### Create initial crack ###


    mySketch2 = myModel.ConstrainedSketch(name='CrackSketch',sheetSize=200.0)
    mySketch2.sketchOptions.setValues(viewStyle=REGULAR)
    mySketch2.setPrimaryObject(option=STANDALONE)
    mySketch2.Spline(localCrack)

    myCrack = myModel.Part(name='CrackPart',
        dimensionality=TWO_D_PLANAR, type=DEFORMABLE_BODY)
    myCrack.BaseWire(sketch=mySketch2)
    mySketch2.unsetPrimaryObject()
    del myModel.sketches['CrackSketch']

    #========================================================================

    ## Create assembly
    myAssembly.Instance(name='myCrack-1', part=myCrack, dependent=OFF)
    myCrackInstance = myAssembly.instances['myCrack-1']
    myCrackInstance.translate(intitialGlobalCrackCoordinates)


    # Create a set for the crack
    edges = myCrackInstance.edges
    myAssembly.Set(edges=edges, name='crackFace')


    #========================================================================
    ## CREATE SETS

    # # Determine the closest node to crack tip

    crackTipCoordinates= ((localCrack[len(localCrack)-1][0]+intitialGlobalCrackCoordinates[0
    ]),
        (localCrack[len(localCrack)-1][1]+intitialGlobalCrackCoordinates[1]), .0)
```

```python
crackTipNodes = myPartInstance.nodes.getByBoundingSphere((crackTipCoordinates),
radiusOfCrackTipNode)

closestindex = 0 # Assume the first index in the crackTipNodes is closest node
closestX = crackTipNodes[closestindex].coordinates[0]-crackTipCoordinates[0]
closestY = crackTipNodes[closestindex].coordinates[1]-crackTipCoordinates[1]
closestdistance = sqrt(pow(closestX,2)+pow(closestY,2))

for index in range(len(crackTipNodes)):
    distanceX = crackTipNodes[index].coordinates[0]-crackTipCoordinates[0]
    distanceY = crackTipNodes[index].coordinates[1]-crackTipCoordinates[1]
    distance = sqrt(pow(distanceX,2)+pow(distanceY,2))
    if distance < closestdistance:
        closestdistance = distance
        closestindex = index

# The closest node as crack tip
crackTipNode = myPartInstance.nodes.sequenceFromLabels((crackTipNodes[closestindex].label
,))
myAssembly.Set(nodes=crackTipNode, name='crackTip')
crackFront = crackTip = myAssembly.sets['crackTip']

#===============================================================================

### Define crack ##

## XFEM crack ##
myAssembly.engineeringFeatures.XFEMCrack(
    crackDomain=myAssembly.sets['crackedPart'],
    crackLocation= myAssembly.sets['crackFace'],
    name='XFEMcrack')

## Crack tip (CI crack) ##
myAssembly.engineeringFeatures.ContourIntegral(name='ContourCrack',
    crackFront=crackFront, crackTip=crackTip,
    extensionDirectionMethod=Q_VECTORS, qVectors=(((localCrack[len(localCrack)-2][0],
    localCrack[len(localCrack)-2][1],0.0),
    (localCrack[len(localCrack)-1][0], localCrack[len(localCrack)-1][1], 0.0)), ),
    midNodePosition=0.5)

#===============================================================================

# Request history output for the crack
myModel.HistoryOutputRequest(name='SIF_history',
    createStepName=crackStep, contourIntegral='ContourCrack',
    numberOfContours=contours, contourType=K_FACTORS, kFactorDirection=KII0, rebar=
    EXCLUDE, sectionPoints=DEFAULT)

#===============================================================================

# Create and submit job
myAssembly.regenerate()

myJob = mdb.Job(name=modelName+'_'+analysisIdentity+'_'+str(nIncrement), model=modelName,
    description='Contour integral analysis')
mdb.saveAs(pathName=modelName+"_"+analysisIdentity+'_'+str(nIncrement))
```

```python
myJob.submit(consistencyChecking=OFF)
myJob.waitForCompletion()



#**************************************************************************

#_____ OUTPUT _____ OUTPUT _____ OUTPUT _____

#**************************************************************************



# Read from history output

import odbAccess
crackODB=session.openOdb(name=modelName+'_'+analysisIdentity+'_'+str(nIncrement), path=
modelName+'_'+analysisIdentity+'_'+str(nIncrement)+'.odb', readOnly=True)
histRegion=crackODB.steps[crackStep].historyRegions['ElementSet  PIBATCH']

crackPropagationDirectionTemp = 0

for i in range(1*contours-contourAverage,1*contours):
    crackPropagationDirectionTemp = crackPropagationDirectionTemp + histRegion.
    historyOutputs[histRegion.historyOutputs.keys()[i]].data[0][1]

crackPropagationDirection.append(crackPropagationDirectionTemp/(contourAverage))



## Calculate average K1 ##
K1_temp = 0
for i in range(3*contours-contourAverage,3*contours):
    K1_temp = K1_temp + histRegion.historyOutputs[histRegion.historyOutputs.keys()[i]].
    data[0][1]

K1.append(K1_temp/(2*contourAverage))  #Divided by 2 as Abaqus over estimate by a factor 2

## Calculate average K2 ##
K2_temp = 0
for i in range(4*contours-contourAverage,4*contours):
    K2_temp = K2_temp + histRegion.historyOutputs[histRegion.historyOutputs.keys()[i]].
    data[0][1]

K2.append(K2_temp/(2*contourAverage))   #Divided by 2 as Abaqus over estimate by a
factor 2

cracktipX= localCrack_temp[len(localCrack_temp)-1][0]+intitialGlobalCrackCoordinates[0]
cracktipY= localCrack_temp[len(localCrack_temp)-1][1]+intitialGlobalCrackCoordinates[1]

crackODB.close()
#=========================================================================

# Calculate the next crack tip from crackPropagationDirection
q1 = [localCrack[len(localCrack)-1][0]-localCrack[len(localCrack)-2][0], localCrack[len(
localCrack)-1][1]-localCrack[len(localCrack)-2][1], 0.0]
q1len = sqrt(pow(q1[0],2)+pow(q1[1],2))
q1[0] = q1[0]/q1len
q1[1] = q1[1]/q1len
```

```python
    q2 = [0.,0.,0.]
    q2[0] = q1[0]*cos(radians(crackPropagationDirection[nIncrement]))-q1[1]*sin(radians(
    crackPropagationDirection[nIncrement]))
    q2[1] = q1[0]*sin(radians(crackPropagationDirection[nIncrement]))+q1[1]*cos(radians(
    crackPropagationDirection[nIncrement]))


    #==============================================================================
    # Save the results
    # nIncrement crackPropagationDirection cracktipX cracktipY K1 K2

    outputRes = open('XFEMCrackResults'+"_"+modelName+"_"+analysisIdentity+'.txt','a')
    lines = [str(nIncrement),"    ",
        str(crackPropagationDirection[len(crackPropagationDirection)-1]),"    ",
        str(cracktipX),"    ",
        str(cracktipY),"    ",
        str(K1[len(K1)-1]),"    ",
        str(K2[len(K2)-1]),"    ",
        "\n"]
    outputRes.writelines(lines)
    outputRes.close()

    localCrack_temp.append((localCrack_temp[len(localCrack_temp)-1][0]+incrementSize_temp*q2[
    0],localCrack_temp[len(localCrack_temp)-1][1]+incrementSize_temp*q2[1]))

print ('Simulation completed!')
```