Some Fracture Studies using Digital Photoelasticity

Prataprao Patil

A Thesis Submitted to

Indian Institute of Technology Hyderabad

In Partial Fulfillment of the Requirements for

The Degree of Master of Technology

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Department of Mechanical and Aerospace Engineering

Dec 2015

Declaration

I declare that this written submission represents my ideas in my own words, and where ideas or words of others have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources that have thus not been properly cited, or from whom proper permission has not been taken when needed.

(Signature)

Prataprao Patil

(Prataprao Patil)

ME13M0003

(Roll No.)

;

Approval Sheet

This Thesis entitled Some Fracture Studies using Digital Photoelasticity by Prataprao Patil is approved for the degree of Master of Technology from IIT Hyderabad

Ch Viswanath

(Dr. Viswanath Chinthapenta, Asst. professor) Examiner
Dept. of Mechanical and Aerospace Engineering
IITH

SNKh

(Dr. Syed Nizamuddin Khaderi, Asst. professor) Examiner
Dept. of Mechanical and Aerospace Engineering
IITH

M. 28/04/16

(Dr. M. Ramji, Asso. professor) Adviser
Dept. of Mechanical and Aerospace Engineering
IITH

21.06.2016

(Dr. G. Prabusankar, Asso. professor) Chairman
Dept. of Chemistry
IITH

;

Acknowledgements

I would like to thank my adviser Dr. M. Ramji for his continuous support, motivation, and valuable guidance. This work could not have been achieved if it was his encouragement and solutions to many problems that I faced.

I am also thankful to Dr. S. N. Khaderi for helping me understand, enrich my ideas and valuable suggestions helped me to complete my work. I also thank Dr. Viswanath Chinthapenta and Dr. C. P. Vyasarayani for insightful comments and constructive criticisms at different stages of my research were thought-provoking and they helped me to focus. I would like to acknowledge Mr. RGR Prasath (Guru Sir) for his useful suggestions in learning photoelasticity. I thank my past and present fellow labmates in Engineering Optics lab: Dr. Mohammad Kashfuddoja, Dr. R.Srilakshmi, Vikrant, Vishwajeet, Jabir, Saranath, Sourabh, Yagnik, Milind, Harilal, Naresh for the stimulating discussions and providing their various forms of support. I would thank Mr. K Sathyanarayana of the workshop for letting me use the workshop facilities. I would also like to thank entire workshop staff especially Mr. A. Praveen Kumar and Mr. Pramod Lokhare for their valuable assistance in specimen preparation. Also, the help and support provided by my classmates and friends at IIT Hyderabad were exceptional.

Last but not the least, I would like to thank my family for the support they provided me.

Dedication

Dedicated to

My beloved parents and teachers

Abstract

The fracture analysis of cracks in a structure under complex loading is an important topic to study from safety and cost perspective. This can be studied in different ways like theoretical, numerical or experimental. Theoretical solutions are too ideal for solving the actual structure. For complex cases, numerical or experimental methods are best suited. Photoelasticity has evolved as the most preferred techniques for the evaluation of fracture parameters. The present study attempts to evaluate mixed-mode stress intensity factors (SIF) for different specimen configurations using digital photoelasticity. For estimation of SIF form digital photoelasticity, a new algorithm is proposed to solve multi-parameter stress field equations. The objective function is defined as the square of the error between experimental data and cure fitted data from the multi-parameter equation. The fringe order and principal stress direction around the crack tip have been obtained using digital photoelasticity technique. The fracture parameters and crack tip location are obtained by minimizing the objective function. The linear least square approach is used to solve an optimization problem. In order to ensure the adequate number of parameters, the fringe pattern has been reconstructed theoretically to compare it with the experimentally obtained fringe pattern. To get the fringe order over entire model ten-step method with adaptive quality guided algorithm is used. Using the techniques of digital image processing, data required for evaluation of SIFs has been collected in an automated manner. The presented methodology has been used to extract SIFs for specimen configurations like a single edge notch (SEN) specimen, crack-inclusion and center slant crack (CSC) specimens made out of epoxy resin. The experimental results have been compared with analytical and finite element (FE) results and they are found to be in good agreement, thereby confirming the accuracy of the algorithm.

Further, the same algorithm has been extended to rigid line inclusion in an elastic matrix study as well by deriving multi-parameter stress field equations. The stress field and fracture parameter for rigid line inclusion in a matrix can provide more insight to fiber matrix interaction in short fiber composite. The short fiber composites have superior mechanical properties over parent polymeric material. Under the loading, the deformation of matrix transfers stress to the fiber by means of fiber-matrix interface traction. In addition, stress singularities will also arise at the end of fiber. It can be thoroughly understood by studying a problem of rigid line inclusion embedded in an epoxy matrix. The stress field and fracture parameter for rigid line inclusion in an elastic matrix can provide more insight into the fiber-matrix interaction in short fiber composite. The rigid line inclusion will increase the stiffness of material but it also leads to a singularity at the inclusion tip. Therefore, it is necessary to investigate the stress field around the inclusion tip. The elastic stress and strain field for rigid line inclusion embedded in an elastic matrix are calculated using Stroh formulation along with duality principal. Strain intensity factor is defined based on strain field ahead of the rigid line inclusion tip. The reciprocal theorem is used for strain intensity factor estimation involving finite element analysis (FEA). The multi-parameter stress field equation for rigid line inclusion is derived. Steel line inclusion along loading direction in an epoxy matrix with tensile loading is considered. The strain intensity factor is estimated digital photoelasticity and it is compared with finite element analysis (FEA) estimates. The analytical, numerical and experimental SIFs are found to be in coherence with each other.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Literature review

1.1 Introduction

1.1.1 Introduction to fracture mechanics

The design of structures mainly involves an analysis of stress and displacement fields with the prediction of failure. The conventional failure criteria are based on tensile strength, yield strength and bending stress. These criteria are useful for many engineering materials but fail when there is discontinuity (defect) in the material. All material contain initial defects like cracks, voids, inclusions or second phase material which affect the load carrying capacity of the structure. High stress near the defects is often responsible for lowering the strength of the structure and leads to final failure. These defects in a structure are due to second phase particle, debond in composite, fabrication process such as welding, heat treatment and in service life due to fatigue crack, environment assisted or creep crack etc. Many catastrophic structural failures have occurred due to brittle fracture.

The presence of defect like the crack results in redistribution of stress and strain field around the defect as shown in Fig. 1.1. Study of stress and displacement field near crack tip is important to understand the crack growth and fracture behavior. The stress intensity factor (SIF) is used to express the strength of singular elastic stress field and it can be used to predict the failure of cracked plate. SIF depends on flaw size (a), loading conditions and geometrical configuration.

$$\sigma_{ij} = K r^{1-\lambda} f_{ij}(\theta) \tag{1.1}$$

where K is stress intensity factor and it is expressed in (MPa$\sqrt{\text{mm}}$), λ is a singularity and it is 1/2 for crack problems. There are three basic modes of crack extension as shown in Fig. 1.2. Mode-I is the opening mode and the displacement is normal to the crack surface. Mode-II is a sliding mode and the displacement is perpendicular to crack front in the plane of the crack surfaces. Mode-III is tearing mode and the displacement is parallel to the crack front in the plane of the crack surfaces. The mode-I loading plays a dominant role in many engineering problems.

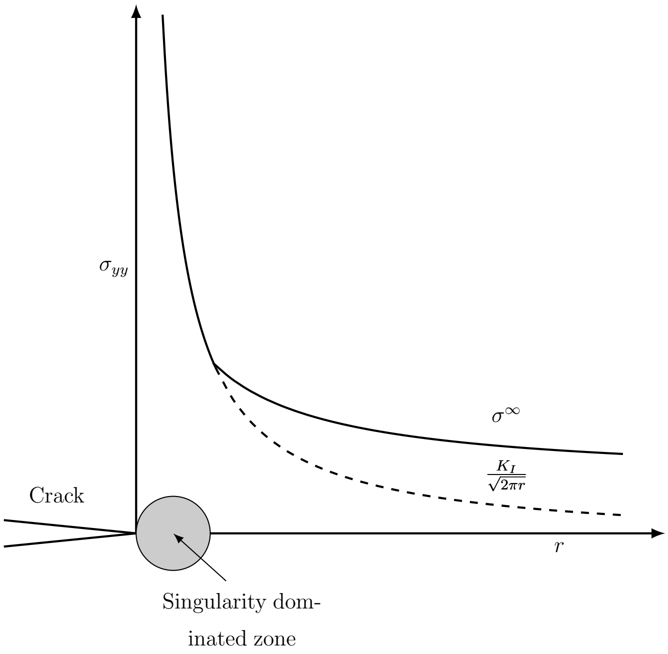


Figure 1.1: Mode-I singular and full-field stress distribution ahead of crack tip normal to crack plane.

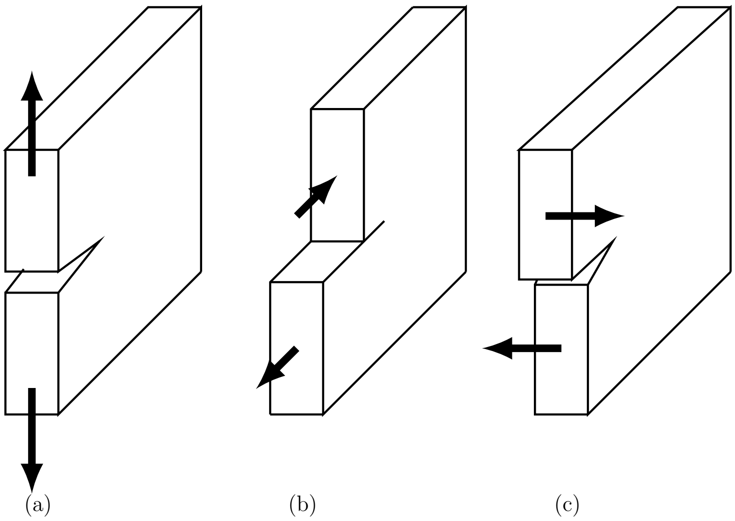


Figure 1.2: Different loading modes for cracked structure (a) mode-I tensile opening (b) mode-II in-plane shear (c) mode-III anti-plane shear.

Fracture of material may occur by different ways as ductile/brittle fracture, fatigue or progressive fracture and delayed fracture. Fatigue fracture is caused by critical localized stress in part subjected to fluctuating stress. The delayed fracture occurs when a material has been statically loaded at an elevated temperature for a long time. The ductile and brittle fracture is classified based on the ability of material to undergo plastic deformation at the crack tip. In ductile fracture, the plastic deformation can be monitored to get an indication of failure of the material. But on the other hand, brittle fracture is crucial as it occurs catastrophically without any warning leading to the spontaneous and rapid crack growth. The failure of cracked material is governed by critical fracture toughness which describes the ability of material containing crack to resist fracture. The fracture toughness is determined from experiments. A crack will start growing once SIF is exceeding the critical fracture toughness.

The fracture parameters are important to predict the growth and sudden failure of the structure. In the past researchers have been estimating SIF with whole field non-contact optical methods such as holographic interferometry [2], Moiré interferometry [3, 4], electronic-speckle-pattern interferometry [3], coherent gradient sensing [5], method of caustics [6], photoelasticity [7], digital image correlation (DIC) [8,9]. Further, contact methods such as resistance strain gauge [**?**] were also used. Methods like holography and other interferometric techniques are very sensitive to vibration and require a coherent light source and complex experimental setup. Among these experimental techniques, photoelasticity provides rich field data for complex geometry and loading with simple optical setup and specimen preparation. Thus, digital photoelasticity is considered for estimation of SIF.

1.1.2 Photoelasticity

Photoelasticity is an optical, non-contact experimental technique for whole field stress analysis. It is based on stress-induced birefringence which yields the information of principal stress difference (isochromatic) and principal stress direction (isoclinic) in the form of fringe contour. This technique has been most widely used in the history of experimental stress analysis. In early days, it was used to study stress concentration factor for complex structural shape for both two and three dimensional. In past days, quantitative isoclinic (θ_c) and isochromatic (N) data were obtained easily only at the fringe contours (see Fig. 1.3).

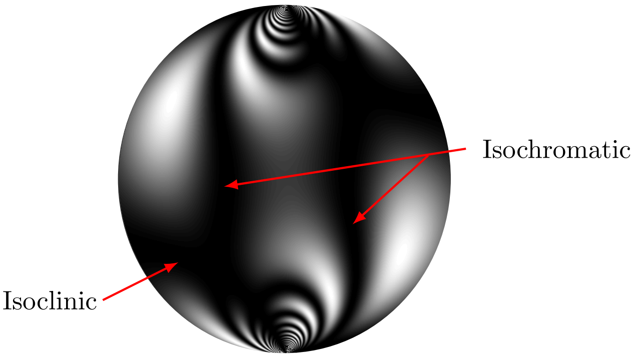


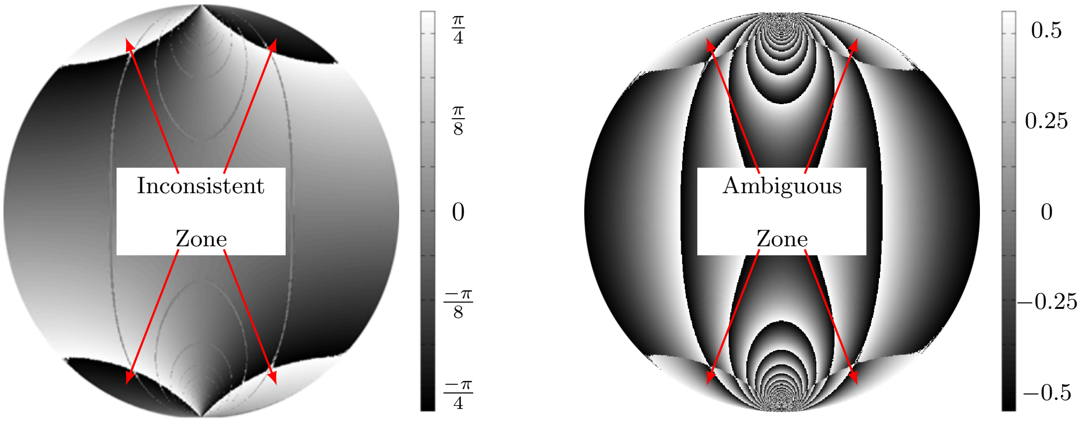
Figure 1.3: Dark field plane polariscope image of a disk under diametric compression showing both isoclinic and isochromatic fringe contours.

With the advent of personal computer-based digital image processing systems, automation of photoelastic parameter estimation has now become simpler. The recent developments in digital image processing have given birth to a separate branch of photoelasticity called 'digital photoelasticity' [10]. A paradigm shift in data acquisition methodologies came into existence with the development of charge-coupled device (CCD) cameras which could record intensity data. Several whole field techniques were also developed. The techniques are broadly classified into spatial domain and frequency domain methods. Phase-shifting techniques (PST), polarization stepping techniques and load stepping come under spatial domain methods. Spatial domain methods require the smaller number of images to be recorded (from three to ten in most cases). Further, they are computationally very fast. Hence, they are considered for whole field isochromatic parameter estimation.

1.2 Literature review

1.2.1 Whole field parameter estimation using digital photoelasticity

Digital photoelasticity provides wrapped phasemap of isoclinic and isochromatic, and then they have to be unwrapped for getting the continuous-phase values. In the case of isochromatic phasemap, ambiguity removal prior to unwrapping is essential. Ramji et al. [11] recently shown that only plane polariscope based methods provide accurate isoclinic values. Use of a ten-step phase shifting method to record accurate values of both isoclinic and isochromatic has now become a standard and a recommended technique in transmission photoelasticity [1]. Recently, Ramji and Prasath have have done an error analysis to find the effectiveness of ten-step methodology for photoelastic parameter estimation and they found to be robust against various error sources [12]. So, in this study ten-step method is considered for the photoelastic parameter estimation. By using the ten-step phase shifting technique (PST), isoclinic data is firstly generated and it is further unwrapped to get the wrapped isochromatic phasemap without any ambiguity. The adaptive quality guided phase unwrapping (AQGPU) [1] algorithm using phase derivative variance is used to unwrap both isoclinic and isochromatic parameters. Figure (1.4) shows the wrapped phasemap of isoclinic and isochromatic which needs to unwrapped to get whole field data.



(a)

(b)

Figure 1.4: Phasemaps for disc under diametric compression (a) wrapped isoclinic phasemap having an inconsistent zone (b)wrapped isochromatic phasemap having an ambiguous zone.

1.2.2 Evaluation of fracture parameters for cracked configuration using digital photoelasticity

Dally and Sanford [13] used the theoretically constructed isochromatic fringe patterns to study the influence of applied traction on size, shape and orientation of fringes. They classified the stress state at the crack tip based on the isochromatic fringe pattern. Also in 1979, Dally and Sanford developed a nonlinear over-deterministic least square methodology, considering multiple data points around crack tip, to determine mixed-mode SIF and T-stress. They used three parameter equation and shown improvement in results because a use of nonlinear equation derived from modified Westergaard approach. Sanford [14] showed that the linear and non-linear least square method can be used, for any type of optical-stress analysis method which produces a fringe pattern over a field. Nigam and Shukla [15] used both photoelasticity and method of caustics for determining SIFś of the specimens with identical geometry and loading conditions. They found a good agreement between the results obtained by both techniques under static loading conditions.

In all the above approaches data collection zone had to confine to the singularity region near crack tip, which is very small as compared to the specimen [16]. The equations to extract SIFś from whole field data are valid in a region close to the crack tip. Sanford [16] extended the over-deterministic least square algorithm and developed the method of local collocation by including few additional lower order non-singular terms to take advantage of whole field information. Ramesh et al. [17] have shown the equivalence for the various multi-parameter stress field equations such as generalized Westergaard [18] equations proposed by Sanford, William's eigenfunction expansion [19] and Atluri Kobayashi multi-parameter equations [20]. They also evaluated crack tip fracture parameters using multi-point over-deterministic non-linear least square approach. They showed that the multi-parameter equation allowed data collection over a larger zone around the crack tip. Guagliano et al. [21] used the same methodology to analyze the effect of adding up to twenty terms in the multi-parameter equation while collecting the data from the isochromatic fringe pattern spread over a wider zone around the crack-tip. Till date, only the isochromatic data has been used for the crack tip fracture parameters estimate but the real advantage of digital photoelasticity has not been explored. To evaluate SIF from an experiment above mentioned method uses non-linear approach but this approach fails for complex configuration in terms of initial value and convergence of solution leading to wrong results. Recently, Harilal et al. [9] proposed a linear approach to solve multi-parameter displacement field equation and validated the approach with DIC experiment for a mixed mode crack problem. We extend this method for SIF determination using digital photoelasticity. To evaluate SIF from photoelasticity, the fracture parameters are evaluated by finding coefficients of the curve fitted multi-parameter stress field equation over the experimental isochromatic data surrounding crack tip. The objective function is defined as the square of the error between experimental and reconstructed fringe order obtained from a multi-parameter equation. This objective function is minimised to obtain the coefficient values using non-linear over deterministic technique where always an initial guess of the coefficient is a must. Hence, a solution is not straight forward and sometimes doesn't converge easily especially in a case of mixed mode problems.

1.2.3 Fracture study of rigid line inclusion in a elastic matrix

The rigid line inclusion is a basic building block for analysis of short fiber composite. They are used for structural applications instead of plane polymeric material for electrical, packaging and automobile applications [22]. In short fiber composites, both fiber and matrix share the applied load, and the load transfer between the matrix and fiber happens via the fiber/matrix interface. As a consequence, the short fiber composites have superior strength and elastic stiffness over the parent polymeric material [23]. However, the fibers could also lead to singular stress field in the matrix near the tip of the fiber. To understand the mechanics of short fiber composites, the problem of a rigid line inclusion embedded in an elastic matrix is usually studied. As a first step towards understanding the mechanics of short fiber composites, the problem of a rigid line inclusion embedded in an elastic matrix is usually studied. Figure 3.4 shows schematic of rigid line inclusion problem. Analysis of the stress field and fracture parameters of a rigid line inclusion in an elastic matrix could provide interesting insights on the fiber-matrix interaction in short fiber composites.

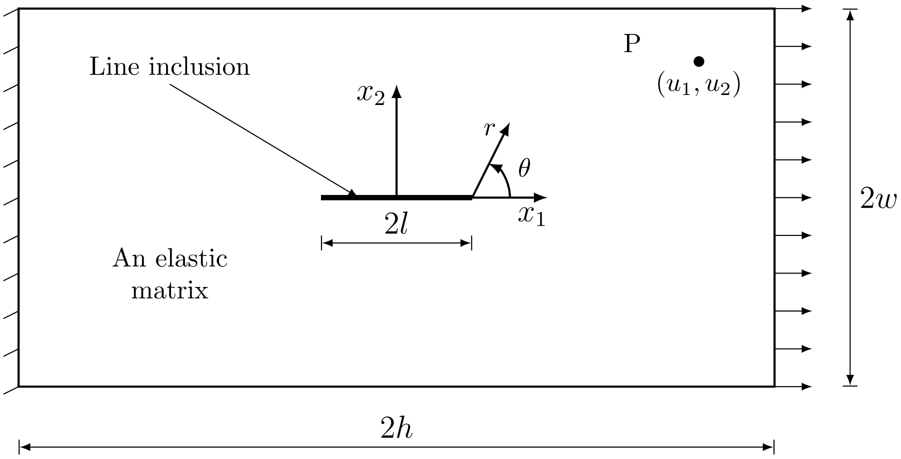


Figure 1.5: A schematic of a rigid line inclusion embedded in an infinite elastic matrix subjected to a loading.

Although the analytical solution for the rigid inclusion problem is known since [24], only few studies exist on quantifying the associated fracture parameters. Atkinson [25] has provided displacement and stress field solution for ribbon like inclusion using complex variable approach. He has studied both rigid and elastic ribbon inclusion. Wang et al. [26] have derived the stress field at the tip of rigid line inclusion and attempted to quantify the elastic singularity using various stress intensity factors (SIF). The SIF for a rigid line inclusion, when the applied remote stress is transverse to the inclusion, was derived by Ballarini [27] using a singular integral and complex potential approach of Muskhelishvili [24]. Hasebe et al. [28, 29] have considered a problem of kinked crack, which started from the tip of a rigid line inclusion. They also investigated the stress fields, SIF and resultant moment acting on the inclusion. Hurtado et al. [30] have solved for stress field of lamellar inhomogeneities such as crack, rigid line inclusion and elastic line inclusion using Eshelbyś equivalent inclusion method. The problem of a rigid line inclusion embedded in a general anisotropic elastic solid was studied by Li and Ting using Stroh formulation [31]. Ni and Nasser [32] showed that the solution for an inclusion can be derived directly from the solution of crack by using duality principle. Recently, the stress field and SIF for a line inclusion embedded in a matrix have been studied using photoelasticity by Noselli et al. [33]. The above mentioned studies focused only on analyzing the elastic fields and SIF for the rigid line inclusion in an infinite matrix.

1.3 Scope and motivation

The SIF can be evaluated theoretically, numerically or experimentally. Analytical methods are restricted to simple configurations and boundary condition. For complex configurations, SIF needs to be extracted using either numerical or by the experimental method. Researchers have developed and implemented several optical method like Moire interferometry, speckle interferometry, holographic interferometry, photoelasticity, digital image correlation for SIF estimation. Most of the interferometric technique measure relative deformation and they are sensitive to vibration. Among these experimental techniques, photoelasticity provides rich field data for complex geometry and loading with a simple set up and specimen preparation. Hence, photoelasticity is considered here for the SIF estimation.

The mixed-mode stress intensity factors (SIF) for different specimen configurations are estimated involving digital photoelasticity technique. Existing methods use non-linear over-deterministic approach to solving multi-parameter stress field equation for crack tip SIF estimation. This approach is sensitive to initial values of crack tip location and fracture parameters, and may lead to a local minimum with poor curve fit of the experimental data. Hence, in order to estimate SIF using the digital photoelastic technique, a new algorithm is proposed for solving the multi-parameter stress field equation. The objective function is defined as the square of error between experimental fringe order and estimated fringe order based on solving the multi-parameter stress field equation. The fringe order and isoclinic around the crack tip has been obtained using ten-step phase shifting technique. To get the fringe order and isoclinic data over the entire model adaptive quality guided phase unwrapping algorithm has been used. The fracture parameters and crack tip location are obtained by minimizing the objective function. The linear least square approach is used to solve the optimization problem. In order to ensure the adequate number of parameters, the fringe pattern has been reconstructed theoretically to compare it against the experimental fringe pattern. Using the techniques of digital image processing, photoelastic data required for the evaluation of SIFs has been collected in an automated manner. The presented methodology has been used to extract SIFs for specimen configurations like a single edge notch (SEN) specimen, center slant crack (CSC) and crack-inclusion specimens made out of epoxy resin.

Further, the same methodology is extended to rigid line inclusion problem by deriving the multi-parameter equation. The elastic stress and strain field for a rigid line inclusion embedded in an elastic matrix are derived using the Stroh formulation along with duality principal. Strain intensity factor is defined based on strain field ahead of the rigid line inclusion tip. The reciprocal theorem is used for strain intensity factor estimation involving finite element analysis (FEA). Later, the multi-parameter stress field equation for a rigid line inclusion is derived using a standard procedure in elasticity. A steel line inclusion along the loading direction embedded in an epoxy matrix under tensile load is considered. The strain intensity factor is estimated digital photoelasticity and is compared with the FEA estimates and they are found to be in good agreement.

1.4 Objectives

- Implementation AQGPU algorithm along with ten-step PST for getting the whole field fringe order from digital photoelasticity
- Develop and implementation of an efficient method for accurate prediction of fracture parameters for cracked configuration using digital photoelasticity. An over deterministic linear least square approach for solving multi-parameter stress field equation is proposed
- Define material independent fracture parameter for rigid line inclusion based on analytical solution
- Develop numerical and experimental tool to estimate fracture parameters at tip of rigid line inclusion embedded in an elastic matrix

1.5 Thesis layout

Chapter 1 deals with the introduction to fracture mechanics and the digital photoelasticity. Also, a brief literature review on fracture parameter estimation from digital photoelasticity and its extension to rigid line inclusion problem is presented. Finally, scope and motivation are defined.

Chapter 2 describes the implementation of a ten-step phase shifting method with AQGPU algorithm to get the whole field parameter using digital photoelasticity. Later unwrapped isochromatic phasemaps are obtained for standard problems like disc, ring, and plate with hole specimens.

Chapter 3 deals with implementation of a linear approach to estimate the SIF using isoclinic and isochromatic data from digital photoelasticity. The methodology is validated with finite element analysis for cracked samples.

Chapter 4 explains the estimation of strain intensity factor for a rigid line inclusion in a finite elastic matrix from the numerical and experimental analysis. The multi-parameter equation for a rigid line inclusion is derived and used for fracture parameter estimation using digital photoelasticity.

Chapter 5 is the conclusion and recommendation for future work.

Chapter 2

Whole field isochromatic parameter estimation using digital photoelasticity

2.1 Introduction

Photoelasticity is an optical, whole field technique for experimental stress analysis. The photoelastic technique gives physical insight to the engineering design. The method is mainly used for studying two- dimensional plane problem, also it can be extended to three-dimensional study as well. This method based on the birefringence property exhibited by many transparent materials. Upon the application of stress, the photoelastic material shows fringe contours. These fringe contours are related to the principal stress difference in a plane normal to the direction of light propagation. In early days, the fringe order data is obtained manually. With the advent of PC-based digital image processing systems to obtain the intensity data, a quantitative and automatic estimation of the whole field photoelastic parameters has now become possible. Because of the advances in digital photoelasticity, photoelastic analysis has become more efficient and reliable technique for understanding the complex structural behavior.

This chapter elaborates estimation whole field photoelastic parameters. The ten-step method is used for data acquisition, as the isoclinic data is not affected by small optical misalignment [12]. This acquired data gives wrapped isoclinic data with inconsistent zone and wrapped isochromatic data with an ambiguous zone. Further to get full field data, wrapped isoclinic data is unwrapped using adaptive quality guided algorithm (AQGPU) [1]. This algorithm unwraps data with minimal user input. The same procedure is followed to unwrap the isochromatic phasemap.The whole fringe order is evaluated for standard problems like disc under diametrical compression, ring under diametrical compression and plate with a hole under tensile load. The estimated parameter is compared with the analytical solution.

2.2 Whole field digital photoelastic parameter estimation

2.2.1 Data acquisition in digital photoelasticity

Phase shifting techniques (PST) are one of the widely used methodologies for quantitative extraction of an isochromatic and isoclinic parameter at every pixel in the domain. In phase shifting techniques, specific phase shifts are introduced for the recorded images by rotating optical elements [10]. Ten-step method [34] is one of such phase shifting techniques. Recently, Ramji and Prasath [12] recommended the use of ten-step phase shifting method for manual polariscope with digital photoelastic applications. It has been found that ten-step method gives both isoclinic and isochromatic parameter with greater accuracy as compared to other phase shifting methods even in the presence of the small optical misalignment. Figure 2.1 shows the optical element arrangement for photoelastic technique.

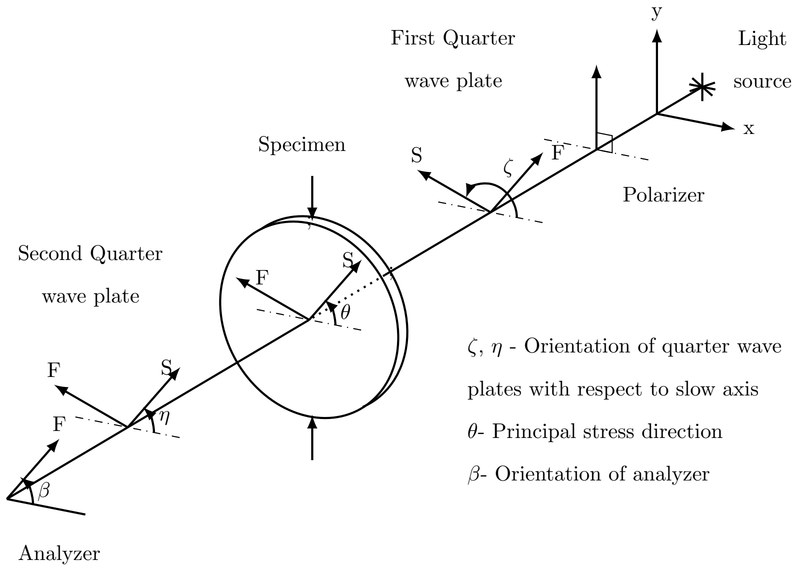

y
First Quarter
wave plate
Light
source
x
Specimen
S
ζ
F
Polarizer
Second Quarter
wave plate
F
S
θ
F
S
η
F
β
Analyzer
ζ, η - Orientation of quarter wave
plates with respect to slow axis
θ- Principal stress direction
β- Orientation of analyzer

Figure 2.1: Generic optical element arrangement for photoelasticity technique.

2.2.2 Ten-step phase shifting technique

The optical arrangements of the ten-step method are shown in Table (2.1). The first four steps correspond to the optical arrangements of the plane polariscope. The next six steps are that of a circular polariscope arrangement.

Table 2.1: Ten-step method: Polariscope arrangements and intensity equations for isoclinic and isochromatic evaluation

α	ζ	η	β	Intensity Equation
0	-	-	$\pi/2$	$I_1 = I_b + I_a \sin^2(\frac{\delta}{2}) \sin^2 2\theta$
$\pi/8$	-	-	$\pi/8$	$I_2 = I_b + \frac{I_a}{2} \sin^2(\frac{\delta}{2})(1 - \sin 4\theta)$
$\pi/4$	-	-	$\pi/4$	$I_3 = I_b + I_a \sin^2(\frac{\delta}{2}) \cos^2 2\theta$
$3\pi/8$	-	-	$3\pi/8$	$I_4 = I_b + \frac{I_a}{2} \sin^2(\frac{\delta}{2})(1 + \sin 4\theta)$
$\pi/2$	$3\pi/4$	$\pi/4$	$\pi/2$	$I_5 = I_b + \frac{I_a}{2}(1 + \cos\delta)$
$\pi/2$	$3\pi/4$	$\pi/4$	0	$I_6 = I_b + \frac{I_a}{2}(1 - \cos\delta)$
$\pi/2$	$3\pi/4$	0	0	$I_7 = I_b + \frac{I_a}{2}(1 - \sin 2\theta \sin\delta)$
$\pi/2$	$3\pi/4$	$\pi/4$	$\pi/4$	$I_8 = I_b + \frac{I_a}{2}(1 + \cos 2\theta \sin\delta)$
$\pi/2$	$\pi/4$	0	0	$I_9 = I_b + \frac{I_a}{2}(1 + \sin 2\theta \sin\delta)$
$\pi/2$	$\pi/4$	$3\pi/4$	$\pi/4$	$I_{10} = I_b + \frac{I_a}{2}(1 - \cos 2\theta \sin\delta)$

The wrapped isoclinic and isochromatic values are obtained by

$$\theta_c = \frac{1}{4} \tan^{-1}\left(\frac{I_4 - I_2}{I_3 - I_1}\right) \tag{2.1}$$

$$\delta_c = \tan^{-1}\left(\frac{(I_9 - I_7)\sin 2\theta + (I_8 - I_{10})\cos 2\theta}{I_5 - I_6}\right) \tag{2.2}$$

From Eqn. (2.1), one can get the wrapped isoclinic phase map in the range $-\pi/4$ to $+\pi/4$ with inconsistent zone and it needs to be unwrapped. Later, the unwrapped isoclinic data is used for the isochromatic phase map generation as given in Eqn. (2.2). Finally, the wrapped isochromatic phase map has to be unwrapped for getting the continuous fringe order. Here to evaluate $\tan^{-1}()$ in both θ_c and δ_c equation $atan2()$ function should be used.

2.3 Unwrapping of isoclinic phasemap

Figure 2.2a shows the wrapped isoclinic phasemap obtained the problem of a disk under diametrical compression. The wrapped isoclinic phasemap contains inconsistent zones. In these zones, the phasemap represents the principal stress direction of the other principal stress. The principal stress σ_1 direction over entire domain but in an inconsistent zone it corresponds to σ_2 direction and there is $\pi/2$ jump. The isochromatic phasemap estimated using wrapped isoclinic values in Eqn. 2.2 is shown in Fig. 2.3 and it has ambiguous zone. In non-ambiguous zone black to white fringe color transition is towards the loading point whereas in ambiguous zone it is reversed. Figure 2.2b shows the comparison of wrapped and analytical isoclinic values along a line ($y/r = 0.82$). The wrapped isoclinic value is not continuous as analytical value but there is $\pi/2$ jump and it will happen at a boundary of the inconsistent zone. To make the wrapped isoclinic value continuous, a constant value of $\pi/2$ should be added or subtracted in identified inconsistent zone. This process is called phase unwrapping. One can observe there is one to one correspondence between the inconsistent zones in isoclinic phasemap and the ambiguous zones in isochromatic phasemap. The ambiguous free zones in isochromatic phasemap can be obtained by use of unwrapped phasemap of isoclinic.

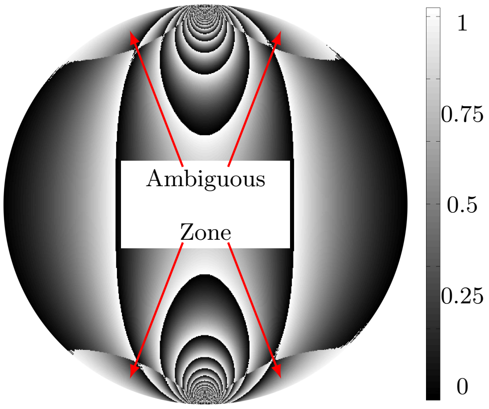
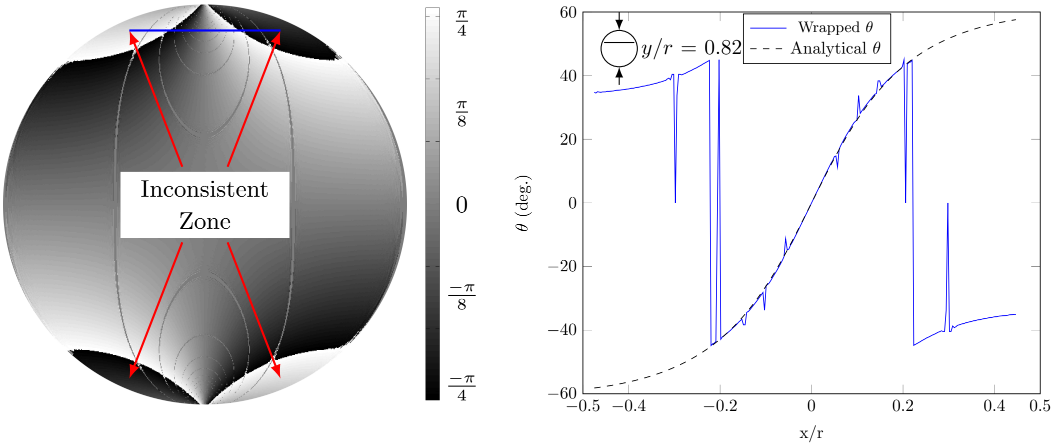


Figure 2.3: Wrapped isochromatic phasemap with ambiguous zone for theoretically generated disc under diametric compression.



(a) (b)

Figure 2.2: Phasemap for theoretically generated disc under diametric compression (a) Wrapped isoclinic phasemap with inconsistent zone (b) Comparison of wrapped isoclinic with analytical obtained values along a line ($y/R = 0.82$).

2.3.1 Adaptive quality guided isoclinic phase unwrapping algorithm

Ten step PST method yield the isoclinic values in the form of wrapped phasemap generated by employing Eqn. (2.1). In the isoclinic phasemap there is presence of abrupt change in gray values near loading point. In these zones, the phasemap represents the principal stress direction of the other principal stress and these zones are called inconsistent zones. It is because the arctangent function returns the principal isoclinic value in the range $\pi/4$ to $+\pi/4$ whereas, physically, the isoclinic value must be in the range $\pi/2$ to $+\pi/2$. Therefore, one needs to unwrap the isoclinic phase map to get them in the range $\pi/2$ to $+\pi/2$ for further utility and it is carried out by AQGPU algorithm [1].

To start AQGPU algorithm, first quality map need to be generated which will guide in deciding a path for unwrapping process autonomously. This quality map is defined as the measure of the statistical variance of the phase derivatives. The variance of the phase derivative is given as

$$\frac{\sqrt{\sum\left(\Delta^{x}_{i,j} - \bar{\Delta}^{x}_{m,n}\right)^2 + \sum\left(\Delta^{y}_{i,j} - \bar{\Delta}^{y}_{m,n}\right)^2}}{k^2} \tag{2.3}$$

where for each sum, the indices (i, j) range over a k x k neighborhood of each center pixel

(m,n) and $\Delta^x_{i,j}, \Delta^y_{i,j}$ are the partial derivatives of the phase value as given below

$$\Delta^x_{i,j} = W_{i+1,j} - W_{i,j} \tag{2.4}$$

$$\Delta^y_{i,j} = W_{i,j+1} - W_{i,j} \tag{2.5}$$

where W is the wrapped phase value. The window size k considered for this problem is chosen as 3 x 3. The phase derivative variance represents the badness of phase data. The quality values are negative of phase derivative variance value. The quality map is shown in Fig. (2.4) for disc and it is normalized in range $0-1$, black color corresponds to zero value shows bad quality.

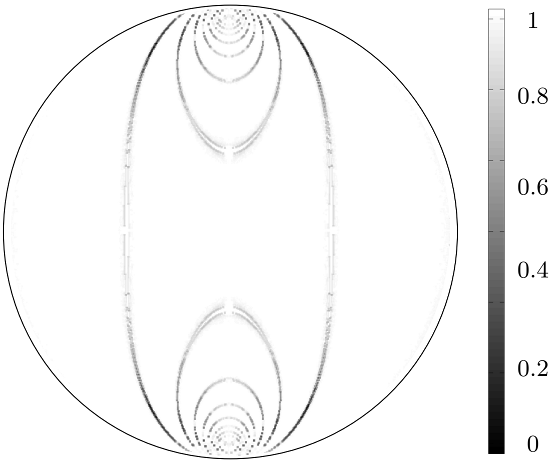


Figure 2.4: Quality map for disc under diametrical compression.

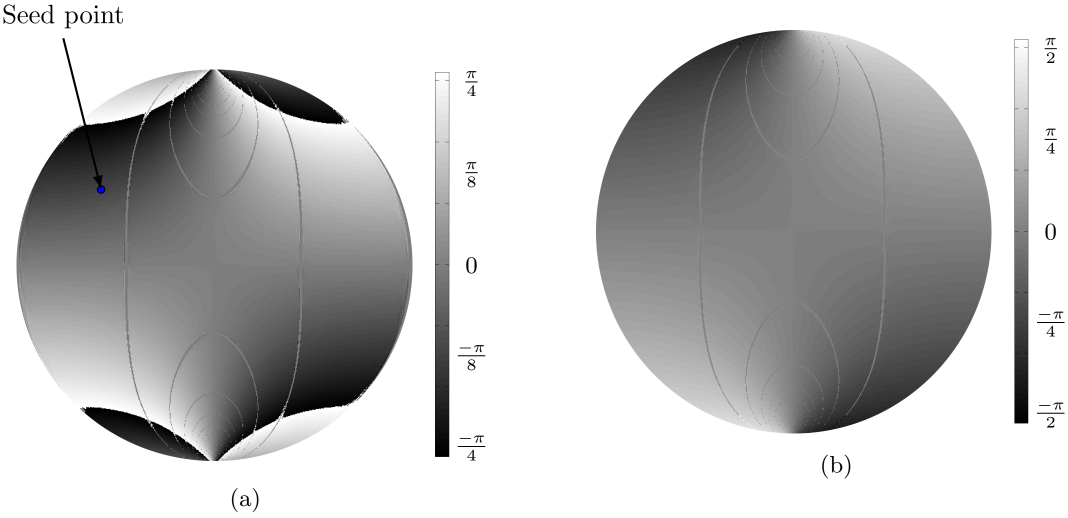
Table 2.2: Checking condition for isoclinic unwrapping based upon pixel position for AQGPU algorithm [1]

Sr. No.	Pixel position	Checking condition
1	Left Pixel (LP)	$abs(\theta_{i,j} - \theta_{i-1,j}) > \theta_{tol}), then$ $\theta_{i-1,j} = \begin{cases} \theta_{i-1,j} + \frac{\pi}{2} & \theta_{i,j} \leq \frac{-\pi}{2} \ \& \ \theta_{i-1,j} \leq 0 \ \& \ (\theta_{i-1,j} - \theta_{i,j}) > \theta_{tol} \\ \theta_{i-1,j} - \frac{\pi}{2} & \theta_{i,j} > \frac{-\pi}{2} \ \& \ (\theta_{i-1,j} - \theta_{i,j}) > \theta_{tol} \end{cases}$ $\theta_{i-1,j} = \begin{cases} \theta_{i-1,j} - \frac{\pi}{2} & \theta_{i,j} \geq \frac{\pi}{2} \ \& \ \theta_{i-1,j} \geq 0 \ \& \ (\theta_{i-1,j} - \theta_{i,j}) < -\theta_{tol} \\ \theta_{i-1,j} + \frac{\pi}{2} & \theta_{i,j} < \frac{\pi}{2} \ \& \ (\theta_{i-1,j} - \theta_{i,j}) < -\theta_{tol} \end{cases}$
2	Right Pixel (RP)	$abs(\theta_{i,j} - \theta_{i+1,j}) > \theta_{tol}, then$ $\theta_{i+1,j} = \begin{cases} \theta_{i+1,j} + \frac{\pi}{2} & \theta_{i,j} \leq \frac{-\pi}{2} \ \& \ \theta_{i+1,j} \leq 0 \ \& \ (\theta_{i+1,j} - \theta_{i,j}) > \theta_{tol} \\ \theta_{i+1,j} - \frac{\pi}{2} & \theta_{i,j} > \frac{-\pi}{2} \ \& \ (\theta_{i+1,j} - \theta_{i,j}) > \theta_{tol} \end{cases}$ $\theta_{i+1,j} = \begin{cases} \theta_{i+1,j} - \frac{\pi}{2} & \theta_{i,j} \geq \frac{\pi}{2} \ \& \ \theta_{i+1,j} \geq 0 \ \& \ (\theta_{i+1,j} - \theta_{i,j}) < -\theta_{tol} \\ \theta_{i+1,j} + \frac{\pi}{2} & \theta_{i,j} < \frac{\pi}{2} \ \& \ (\theta_{i+1,j} - \theta_{i,j}) < -\theta_{tol} \end{cases}$
3	Bottom Pixel (BP)	$abs(\theta_{i,j} - \theta_{i,j+1}) > \theta_{tol}, then$ $\theta_{i,j+1} = \begin{cases} \theta_{i,j+1} + \frac{\pi}{2} & \theta_{i,j} \leq \frac{-\pi}{2} \ \& \ \theta_{i,j+1} \leq 0 \ \& \ (\theta_{i,j+1} - \theta_{i,j}) > \theta_{tol} \\ \theta_{i,j+1} - \frac{\pi}{2} & \theta_{i,j} > \frac{-\pi}{2} \ \& \ (\theta_{i,j+1} - \theta_{i,j}) > \theta_{tol} \end{cases}$ $\theta_{i,j+1} = \begin{cases} \theta_{i,j+1} - \frac{\pi}{2} & \theta_{i,j} \geq \frac{\pi}{2} \ \& \ \theta_{i,j+1} \geq 0 \ \& \ (\theta_{i,j+1} - \theta_{i,j}) < -\theta_{tol} \\ \theta_{i,j+1} + \frac{\pi}{2} & \theta_{i,j} < \frac{\pi}{2} \ \& \ (\theta_{i,j+1} - \theta_{i,j}) < -\theta_{tol} \end{cases}$
4	Top Pixel (TP)	$abs(\theta_{i,j} - \theta_{i,j-1}) > \theta_{tol}, then$ $\theta_{i,j-1} = \begin{cases} \theta_{i,j-1} + \frac{\pi}{2} & \theta_{i,j} \leq \frac{-\pi}{2} \ \& \ \theta_{i,j-1} \leq 0 \ \& \ (\theta_{i,j-1} - \theta_{i,j}) > \theta_{tol} \\ \theta_{i,j-1} - \frac{\pi}{2} & \theta_{i,j} > \frac{-\pi}{2} \ \& \ (\theta_{i,j-1} - \theta_{i,j}) > \theta_{tol} \end{cases}$ $\theta_{i,j-1} = \begin{cases} \theta_{i,j-1} - \frac{\pi}{2} & \theta_{i,j} \geq \frac{\pi}{2} \ \& \ \theta_{i,j-1} \geq 0 \ \& \ (\theta_{i,j-1} - \theta_{i,j}) < -\theta_{tol} \\ \theta_{i,j-1} + \frac{\pi}{2} & \theta_{i,j} < \frac{\pi}{2} \ \& \ (\theta_{i,j-1} - \theta_{i,j}) < -\theta_{tol} \end{cases}$

Table 2.2 shows the checking condition for AQGPU algorithm and the quality information is used for the autonomous propagation of the algorithm while unwrapping. The inconsistent zone in isoclinic phasemap is located by checking the condition and a constant value of $\pi/2$ is added or subtracted for getting continuous phasemap. For unwrapping of a complex model, AQGPU algorithm is preferred as it unwraps autonomously with minimal user intervention. Using the AQGPU algorithm, unwrapping is done in a single step within the masked area. The unwrapping process starts with a seed point selected in a high-quality zone. Initially from

the given seed point, its four adjacent neighbors are unwrapped. These neighbors are sorted based upon their quality values (higher quality value to lower quality value) in a list called the adjoin list. A minimum quality threshold decides up to what value of quality the adjoin list would have and it changes as the unwrapping propagates. Four neighbors of the pixel with the highest-quality value from the adjoin list are then unwrapped. If any of the neighbor pixels has already been unwrapped, they are not considered for unwrapping. The highest quality pixel is then removed from the list. In this way, the path of unwrapping is decided by quality. It is essentially a region growing approach based upon flood filling algorithm. Then based on checking condition the isoclinic unwrapping is carried out. In checking condition the θ_{tol} value of $\pi/3$ is considered. The AQGPU algorithm is written in MATLAB software. Figure 2.5 shows the flowchart of the implemented AQGPU algorithm.

The wrapped and unwrapped isoclinic phasemap is shown in Fig. (2.6) and the seed point is marked. The seed point is a starting point for the AQGPU algorithm which is as user input. The close observation of unwrapped isoclinic phasemap shows there is a discontinuity in phasemap at isoclinic-isochromatic interaction region.



(a)

(b)

Figure 2.6: Phasemap for theoretically generated disc under diametric compression (a) wrapped isoclinic (b) unwrapped isoclinic.

2.4 Unwrapping of isochromatic phasemap

The unwrapped isoclinic phasemap values are used in Eqn. (2.2) and wrapped isochromatic phasemap without ambiguous zone is obtained. These isochromatic phasemap shows fractional

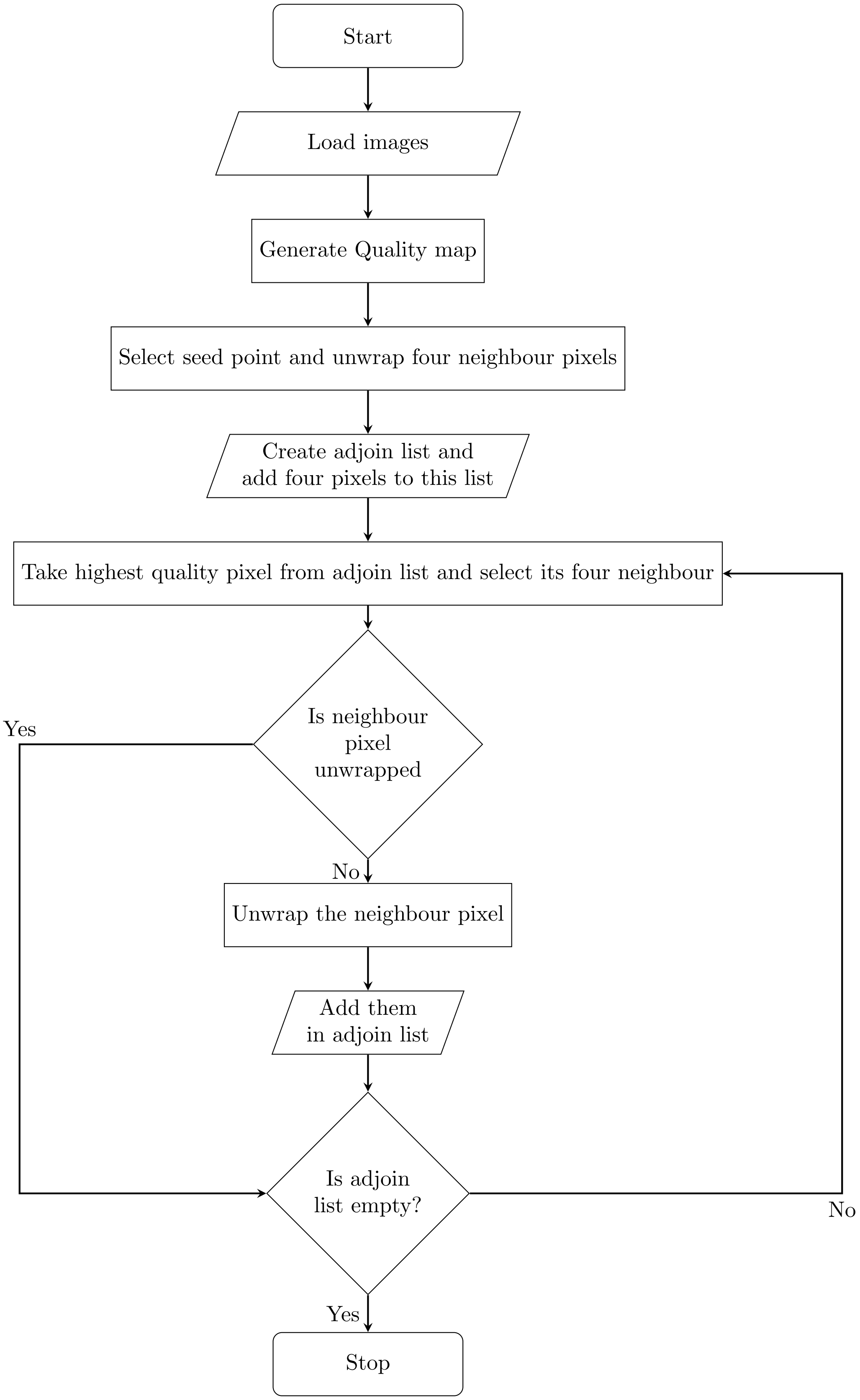

Start
Load images
Generate Quality map
Select seed point and unwrap four neighbour pixels
Create adjoin list and add four pixels to this list
Take highest quality pixel from adjoin list and select its four neighbour
Is neighbour pixel unwrapped
Yes
No
Unwrap the neighbour pixel
Add them in adjoin list
Is adjoin list empty?
No
Yes
Stop

Figure 2.5: Flowchart of AQGPU algorithm.

fringes and they are in order of $-\pi$ to $+\pi$. Before unwrapping of isochromatic, the obtained isochromatic phasemap need to convert in the range of 0 to 2π. It can be done as follows

$$\delta = \begin{cases} \delta_c & \text{for} \qquad \delta_c > 0 \\ 2\pi + \delta_c & \text{for} \qquad \delta_c \leq 0 \end{cases} \tag{2.6}$$

The wrapped isochromatic phasemap is then unwrapped in the same manner as that of isoclinic phasemap. The seed point needs to be selected and corresponding fringe order has to be given as input to start isochromatic unwrapping. Figure (2.7) shows the wrapped and unwrapped isochromatic phasemap.

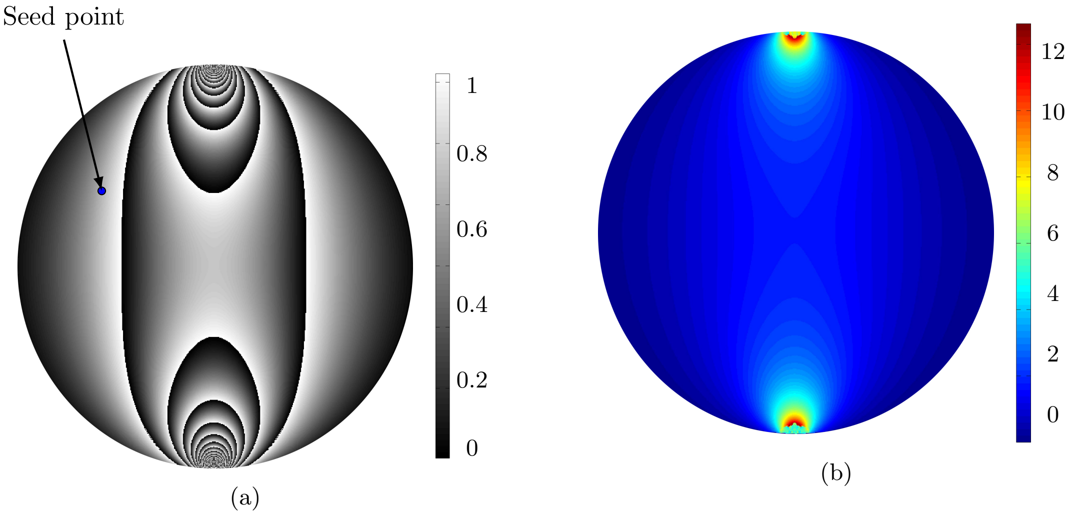


Figure 2.7: isochromatic phasemap for theoretically generated disc under diametric compression (a) Wrapped isochromatic (b) Unwrapped isochromatic.

2.5 Photoelastic parameter estimation in white light

In earlier sections, ten-step phase shifting method for determining isoclinic and isochromatic parameter has been discussed. It is clear from unwrapped isoclinic phasemap, there is discontinuity in isoclinic value at isoclinic-isochromatic interaction (refer Fig. (2.6b)). To overcome this problem in isoclinic phasemap, several researcher suggested the use of multiple wavelengths which helps to shifts the points with zero isochromatic intensity, thereby, achieving a continuous isoclinic [10]. The researcher advocated the use of color CCD cameras for image recording of color images with use of white light. The color image captured by CCD have three planes corresponding to the red, green and blue wavelength. In 1997 Petrucci [35] has proposed a method to get isoclinic phasemap without isoclinic-isochromatic interaction by using color images. The isoclinic phasemap is obtained from first four color images of ten-step method as

$$\theta_c = \frac{1}{4}\tan^{-1}\left(\frac{I_4 - I_2}{I_3 - I_1}\right) \tag{2.7}$$

where $I_i = I_{i,r} + I_{i,g} + I_{i,b}$, $i = 1, 2, 3, 4$ and subscripts $(i, r), (i, g), (i, b)$ represents the red, green and blue wavelength plane for image i. The isochromatic value is obtained by

$$\delta_c = \tan^{-1}\left(\frac{(I_9 - I_7)\sin 2\theta + (I_8 - I_{10})\cos 2\theta}{I_5 - I_6}\right) \tag{2.8}$$

Here $I_i = I_{i,g}$, $i = 5, 6, ..., 10$ and subscript (i, g) represents a green wavelength plane for image i. To get full field data same procedure as explained in previous sections is applicable.

2.6 Specimen preparation and Experimental procedure

Disc and ring specimens are prepared from cast epoxy sheet in a closed mold. The epoxy sheet is casted using resin CY-230 and hardener HY-951 in the ratio 10 : 1 by weight. The mixture is mixed at room temperature with due precaution taken to avoid the formation of any air bubbles. The cast epoxy sheets are then checked in polariscope for the presence of any residual stresses. Then the sheet is machined to get the required disc (60 mm Diameter) and ring (Inner Diameter-40 mm, Outer diameter-80 mm). The disc and ring specimens are studied under diametrical compression. Also, the plate with hole specimen of 200 mm length, 45 mm width is machined from cast epoxy sheet. A specimen have a hole of 10 mm at the center.

Figure (2.9) shows the typical experimental setup used in the photoelasticity study. Images are captured by CCD camera for different optical arrangements as shown in Table (2.1). Images for disc and ring are captured with white light and 3-CCD color camera. The images for the plate with hole sample are recorded using BASLER monochrome CCD camera that has a spatial resolution of 1392 1040 pixels. The monochromatic light source of a wavelength of 589 nm is used in the experiment for a plate with hole specimen. The specimen is loaded using a 10 kN Instron machine. The angles of a polarizer, quarter wave plate-I, quarter wave plate-II and analyzer are so arranged to get ten-step phase shifted images as per Table (2.1).

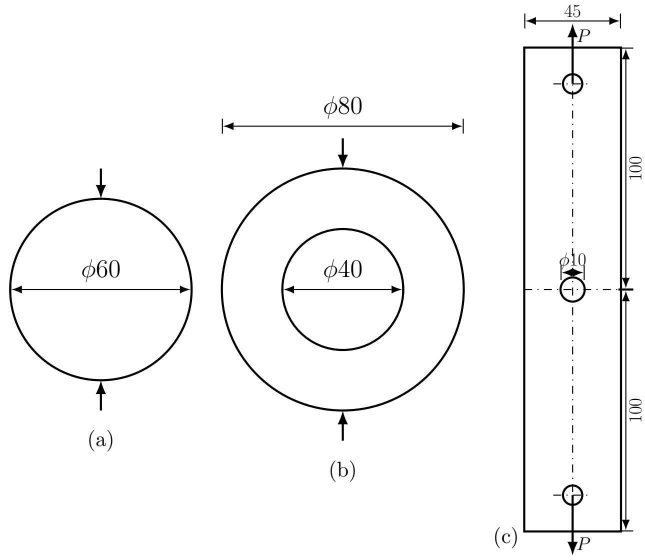


Figure 2.8: Specimens for digital photoelastic experiment (a) disc (b) ring (c) plate with hole.

2.7 Results

2.7.1 Disc under diametrical compression

The color images with ten-step method are captured for disc under diametrical compression. Figure (2.10a) shows wrapped isoclinic phasemap with inconsistent zones marked on the isoclinic phasemap. To start unwrapping by AQGPU algorithm, first quality map is generated and then seed point is selected in consistent zone as shown in Fig. (2.10a). Figure (2.10b) shows the unwrapped isoclinic phasemap obtained by AQGPU.

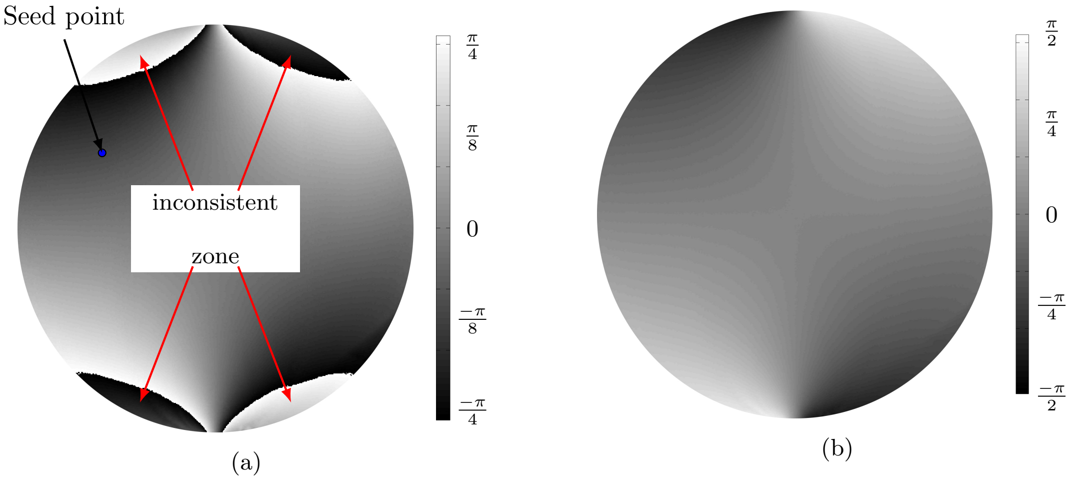


Figure 2.10: Isoclinic phasemap for disc under diametric compression (a) wrapped theta (b) unwrapped theta.

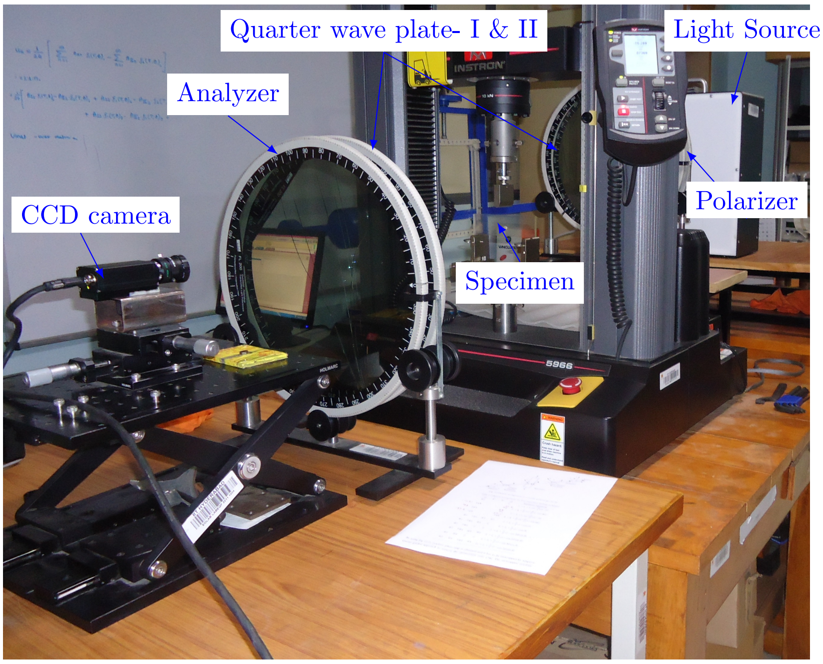


Figure 2.9: Experimental setup for digital photoelasticity.

The wrapped isochromatic phasemap without ambiguous zone is evaluated with use of unwrapped isoclinic value and is shown in Fig. (2.11a). To get continuous isochromatic value over the full domain, unwrapping of isochromatic phasemap is done using AQGPU. Here the seed point ($N = 1$) is selected as shown in Fig. (2.11a) and corresponding fringe order is given as input. The unwrapped isochromatic phasemap is shown in Fig. (2.11b).

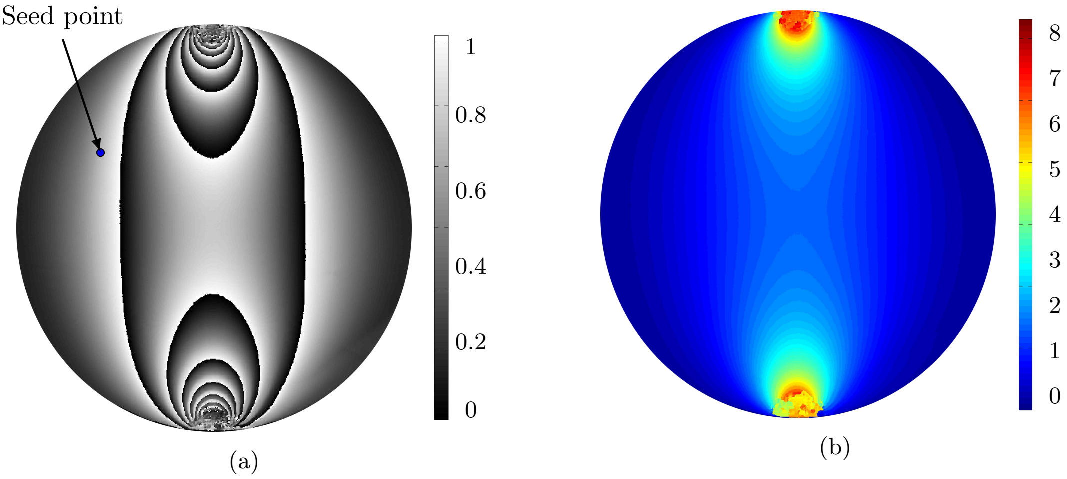


Figure 2.11: Isochromatic phasemap for disc under diametric compression at 500 N load (a) wrapped delta (b) unwrapped delta.

Figure (2.12) shows the quantitative comparison of experimentally obtained wrapped and unwrapped isoclinic values with theoretically calculated isoclinic values for a line ($y/2 = 0.82$). The experimental unwrapped isoclinic data closely matches with theoretical isoclinic value. Also, it is observed that isoclinic-isochromatic interaction is absent with use of color images.

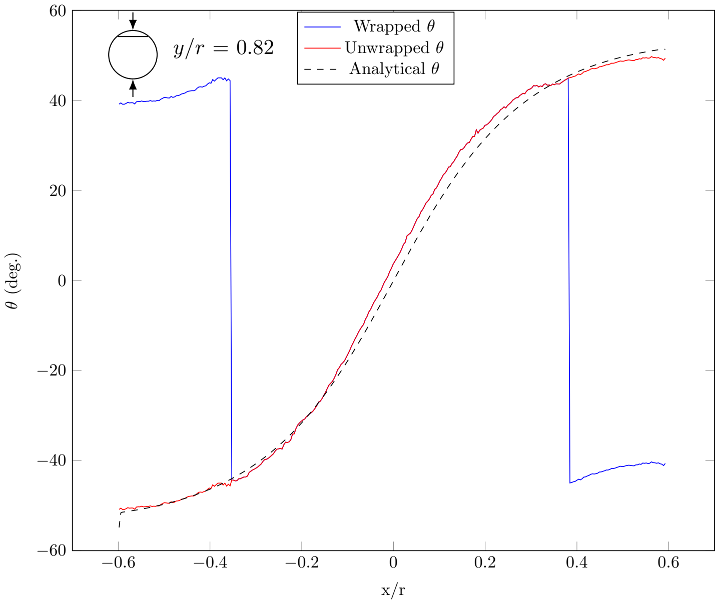


Figure 2.12: Comparison of wrapped and unwrapped isoclinic with analytical obtained values along a line ($y/r = 0.8$) for the problem of a disc under diametrical compression.

The experimentally obtained isochromatic values with analytical isochromatic values for a line at $y/r = 0.65$ in Fig. 2.13). There is a good coherence between analytical and experimental values.

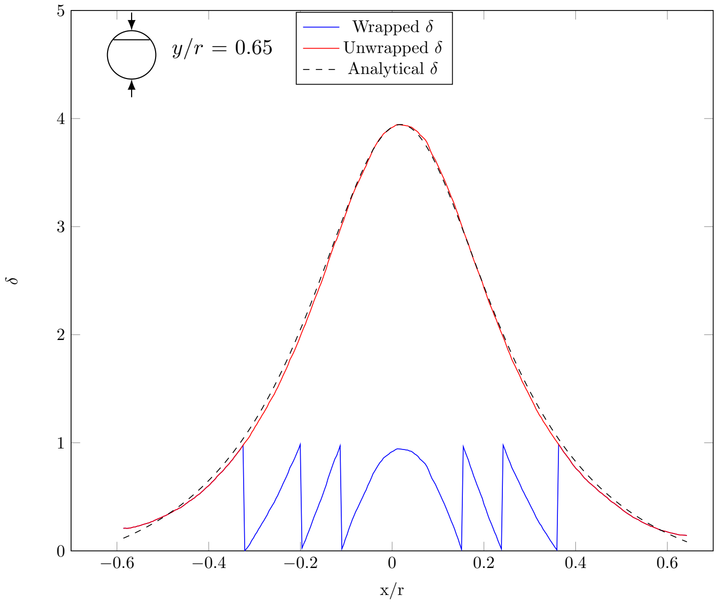


Figure 2.13: Comparison of wrapped and unwrapped isochromatic with analytical obtained values along a line ($y/r = 0.65$) for the problem of a disc under diametrical compression.

2.7.2 Ring under diametrical compression

The ten-step color images are captured for the ring under diametrical compression under a load of 250 N. The wrapped and unwrapped isoclinic phasemap are shown in Fig. (2.14). Both the wrapped and unwrapped isoclinic and isochromatic phasemap are shown in Fig. (2.15).

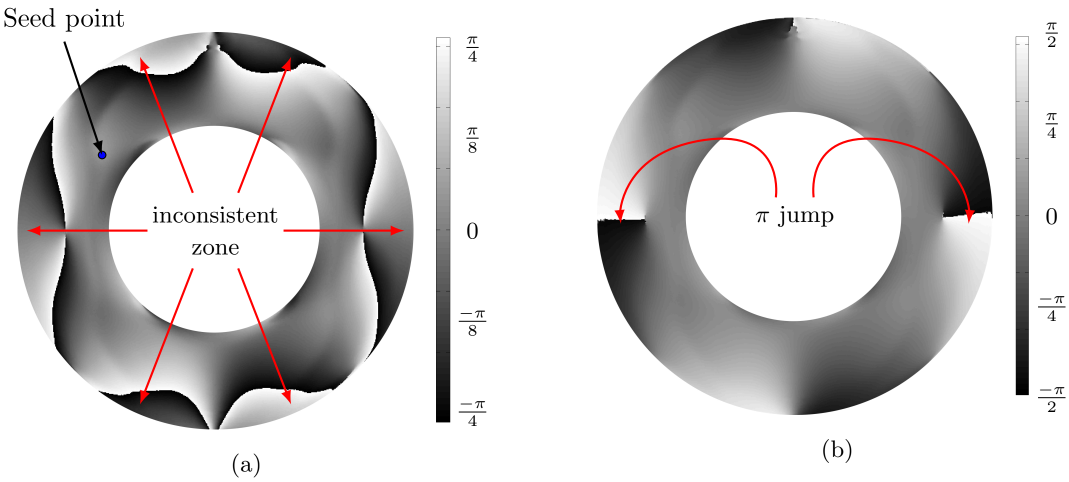


Figure 2.14: Isoclinic phasemap for ring under diametric compression for 250 N load (a) wrapped theta (b) unwrapped theta.

Experimentally obtained wrapped and unwrapped isoclinic values along a line $y/r = 0.8$ is

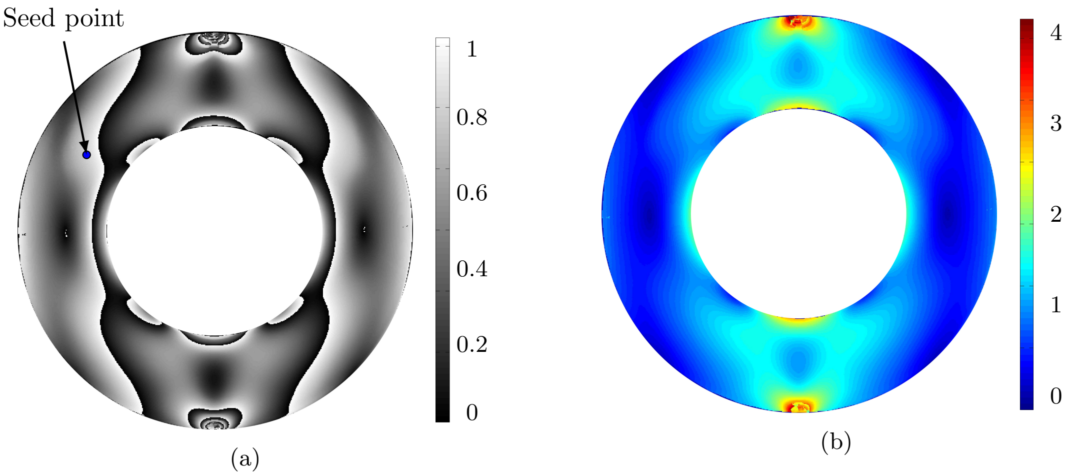


Figure 2.15: Isochromatic phasemap for ring under diametric compression at 250 N load (a) wrapped delta (b) unwrapped delta.

shown in Fig. (2.16) and similarly wrapped and unwrapped isochromatic values along line y/r = 0.65 is shown in Fig. (2.17).

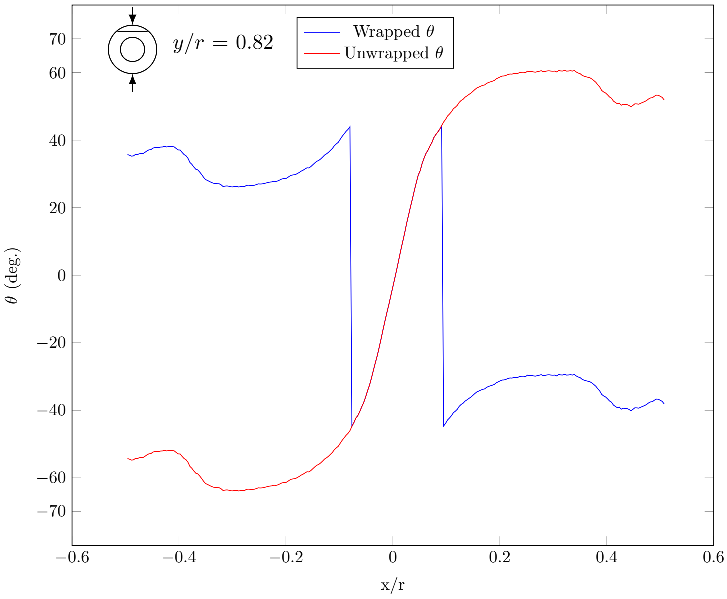


Figure 2.16: Comparison of wrapped and unwrapped isoclinic values along a line (y/r = 0.8) for the problem of a ring under diametrical compression.

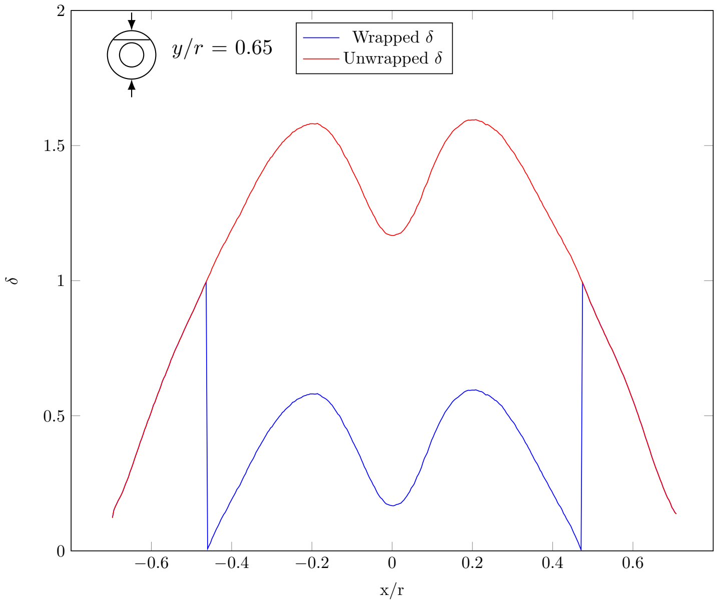


Figure 2.17: Comparison of wrapped and unwrapped isochromatic values along a line ($y/r = 0.65$) for the problem of a ring under diametrical compression.

2.7.3 Plate with hole specimen under tensile load

The ten-step images are captured for the plate with hole subjected to a tensile load of 750 N with monochrome camera. Figure (2.19) shows wrapped and unwrapped isoclinic and isochromatic phasemap for the same. Here, the entire zone in an isoclinic phasemap corresponds to σ_2 direction and is negated to get 1 direction as shown in Fig. (2.18b). Figure (2.19a) shows wrapped isochromatic phase map where black to white transition in a fringe is towards the high stress zone. The unwrapped isochromatic phasemap is shown in Fig. (2.19b). Comparison of wrapped and unwrapped isoclinic and isochromatic values along line $y/r = 0$ are shown in Figs. 2.20 and 2.21. It is clear from Fig. 2.21 that stress concentration is close to 3.

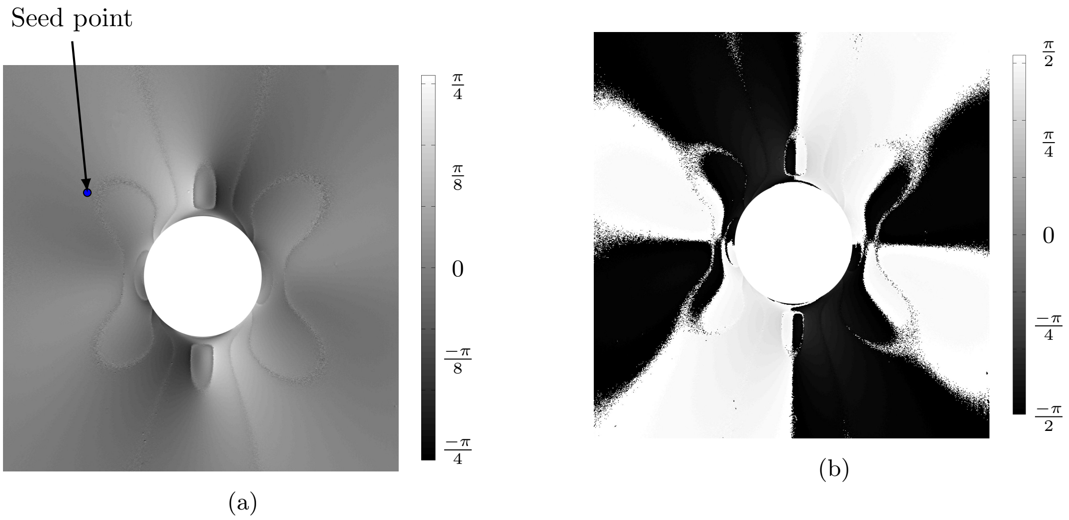


Figure 2.18: Isoclinic phasemap for a plate with hole subjected to a tensile load of 600 N (a) wrapped theta (b) unwrapped theta.

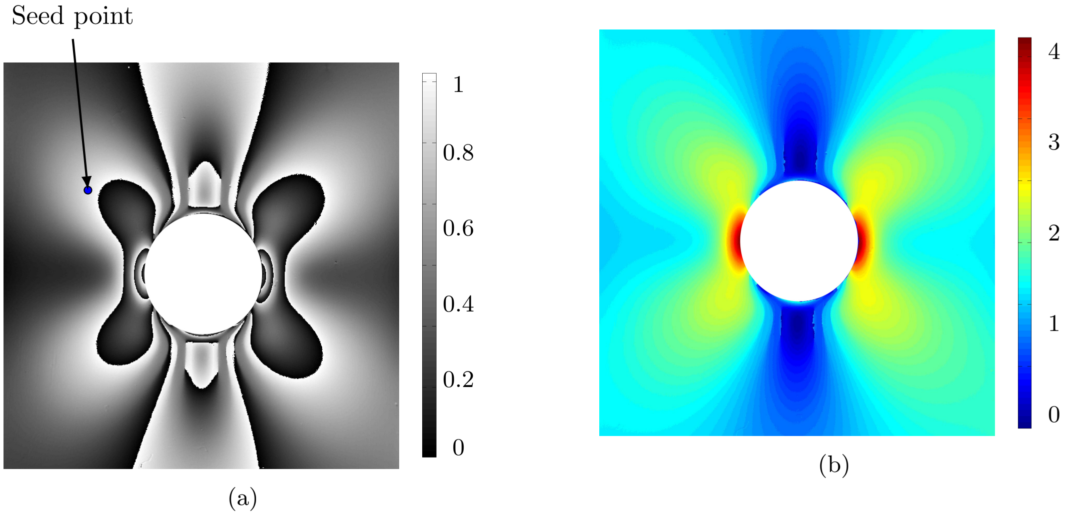


Figure 2.19: Isochromatic phasemap for a plate with hole subjected to a tensile load of 600 N (a) wrapped delta (b) unwrapped delta.

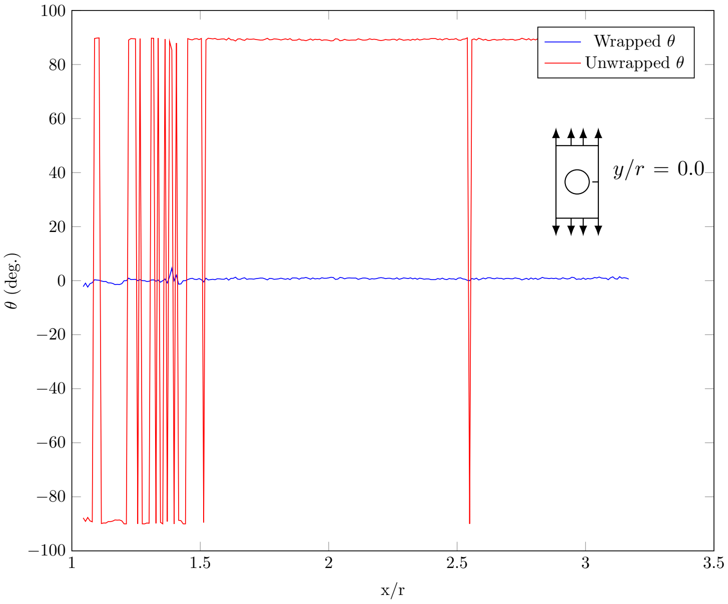


Figure 2.20: Comparison of wrapped and unwrapped isoclinic values along a line (y/r = 0.0) for the problem of a a plate with hole subjected to a tensile load.

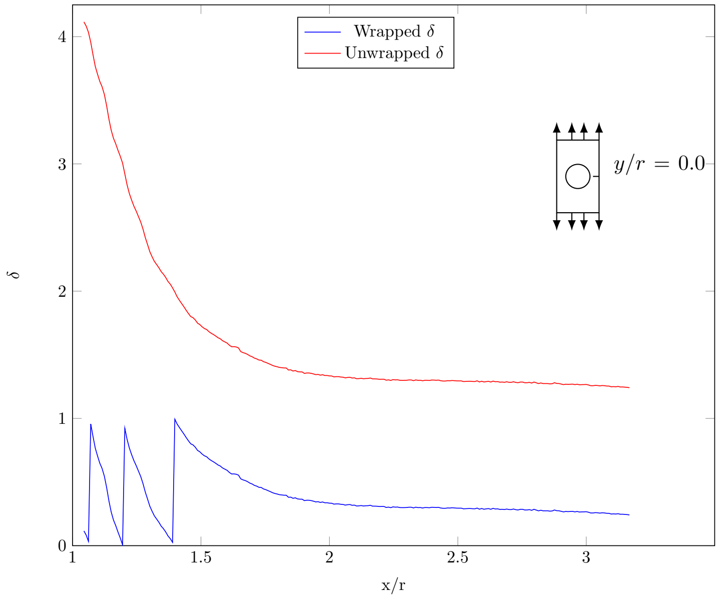


Figure 2.21: Comparison of wrapped and unwrapped isochromatic values along a line (y/r = 0.0) for the problem of a plate with hole subjected to a tensile load.

2.8 Closure

A GUI based code is written in MATLAB software to get the unwrapped isoclinic and isochromatic phasemap corresponding to the ten-step phase shifting algorithm. The isoclinic-isochromatic interaction is avoided using different wavelength from color images. Using AQGPU algorithm the quality map involving phase derivative variance is employed for driving the phase unwrapping algorithm autonomously. The AQGPU algorithm is firstly used for unwrapping the isoclinic phasemap and followed by isochromatic phasemap generation. The obtained results for disc are compared with the analytical and they agree very well thereby confirming the accuracy of the implementation. Later it is applied for different problems to test its applicability and has shown to perform accurately thereby assuming its versatility.

Chapter 3

A novel approach for evaluation of crack tip fracture parameters by linear least squares approach using digital photoelasticity technique

3.1 Introduction

Defects like crack, sharp notches and inclusions play a critical role in the failure of structural components. These defects reduce the strength and upon loading leads to the growth of crack followed by failure. The presence of crack alters the stress and strain field around the crack tip. In fracture mechanics, the stress intensity factor (SIF) expresses the strength of singular elastic stress field and they also characterize near-tip stress field. SIF is a function of applied load, crack length and geometry of the cracked body. The critical SIF value will decide the propagation of crack and the resulting failure under service load. SIF can be estimated using analytical, numerical, or experimental techniques. However, analytical methods are restricted to simple configurations and boundary conditions. For complex configurations, SIF needs to be extracted using either numerical or experimental method. In this work experimental evaluation of SIF using digital photoelasticity is carried out. In the past researchers have been estimating SIF with whole field non-contact optical methods such as holographic interferometry [2], Moiré interferometry [3,4], electronic-speckle-pattern interferometry [3], coherent gradient sensing [5], method of caustics [6], photoelasticity [7], digital image correlation (DIC) [8, 9]. Further, contact methods such as resistance strain gauge [?] were also used. Methods like holography and other interferometric techniques are very sensitive to vibration and require a coherent light source and complex experimental setup.

Among these experimental techniques, photoelasticity provides rich field data for complex geometry and loading with simple optical setup and specimen preparation. Photoelasticity is an optical, non-contact technique for whole field stress analysis which provides the information of principal stress difference (isochromatic) and principal stress direction (isoclinic) in the form of fringe contours. In the early days of its development, quantitative isoclinic (θ_c) and isochromatic (N) data were obtained only along the fringe contours. With the advent of personal computer-based digital image processing systems, automation of photoelastic parameter estimation has now become popular [10] and is often referred as digital photoelasticity.

The fracture parameters are evaluated by finding coefficients of a curve fitted multi-parameter stress field equation over the experimental isochromatic data surrounding crack tip. The objective function is defined as the square of error between experimental and reconstructed fringe order obtained from the multi-parameter equation. This objective function is minimised to obtain the coefficient values using non-linear over deterministic technique where always an initial guess of the coefficient is a must. Hence, a solution is not straight forward and sometimes doesn't converge easily especially in the case of mixed mode problems. The extracted coefficients of the multi-parameter equation are related to the crack tip SIF parameter. Further, finding the exact location of the crack tip is not possible in the above approach and it always results in an uncertainty of the extracted fracture parameters. The real advantage of digital photoelasticity technique is not exploited to the fullest extent as only the isochromatic data is utilised for SIF extraction. In this work, the availability of pixel-wise isochromatic and isoclinic data has enabled us to convert the non-linear regression problem into a linear regression problem with respect to unknown coefficients. The linear regression problem is solved over the chosen grid around crack tip by an over deterministic least square approach. This approach ensures fast and accurate determination of crack tip fracture parameters including the crack tip location.

A ten-step phase shifting technique (PST) [34] is employed to get the whole field isochromatic and isoclinic phasemap over the cracked specimens near the crack tip. The wrapped phase maps are then unwrapped using adaptive quality guided phase unwrapping (AQGPU) algorithm [1]. The data is collected from unwrapped isoclinic and isochromatic phasemap and it is used to estimate SIF by solving the multi-parameter stress field equation in an over deterministic linear least square sense. We have studied different specimen configurations like single edge notch specimen (SEN), centre slant crack (CSC) and crack-inclusion specimens made of epoxy material and they all are subjected to tensile load.

3.2 Specimen preparation and Experimentation

3.2.1 Specimen fabrication

In this study three kinds of cracked specimens are considered namely- SEN specimen, straight crack-inclusion and CSC under tensile load. The SEN and CSC specimens are cut from epoxy sheet casted by mixing commercially available Araldite CY 230 epoxy resin and HY 951 hardener in the proportion of 10:1 by weight. The resin-hardener mixture is mixed thoroughly with due precaution taken to avoid air bubble formation. The mixture is then poured into the mould and allowed to cure for 24 hours at room temperature. The moulded sheet of 6 mm thickness is taken out and then the specimens of the size of 200 mm x 40 mm are machined by milling. During the machining process, precaution is taken to avoid excessive heat generation and high cutting force which will lead to residual stress. Afterwards, the crack is introduced with the help of a grinded hacksaw blade of thickness 0.2 mm and further the crack tip is made sharp by using a toothed razor blade. The SEN specimen has a crack of 10 mm length. For creating CSC specimen, initially, 1 mm hole is drilled at a center of the specimen and then the blade is inserted to generate the inclined crack. The crack is at 45^0 inclination to the loading direction and it is of 20 mm length. To make crack tip sharp same procedure is followed as that of SEN specimen. The specimen geometry of SEN and CSC configuration are shown in Fig. 3.1a and Fig. 3.1b respectively.

For the crack-inclusion specimen, the sheet is casted in a closed mould using the same procedure as explained above. In this mould, a hole is provided to insert a glass rod of 5 mm diameter at

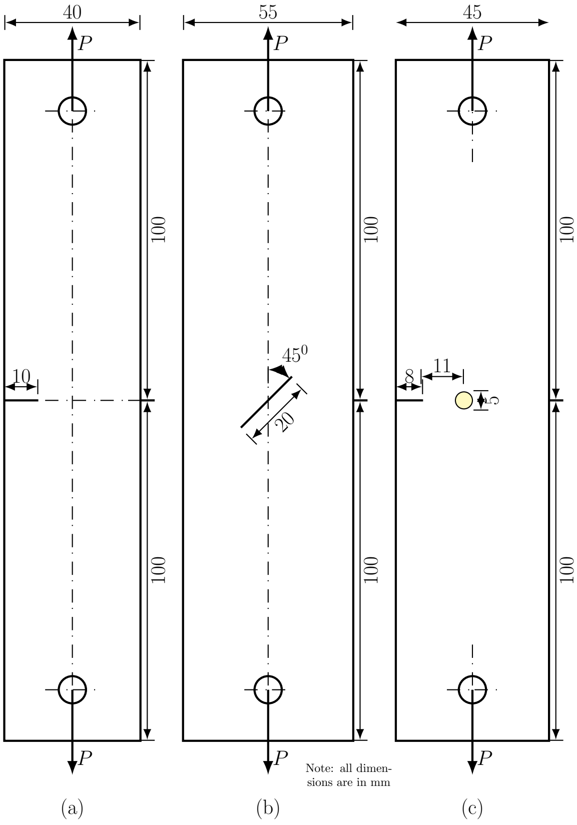


Figure 3.1: Cracked specimen configuration made of epoxy material (a) SEN (b) CSC (c) crack-inclusion.

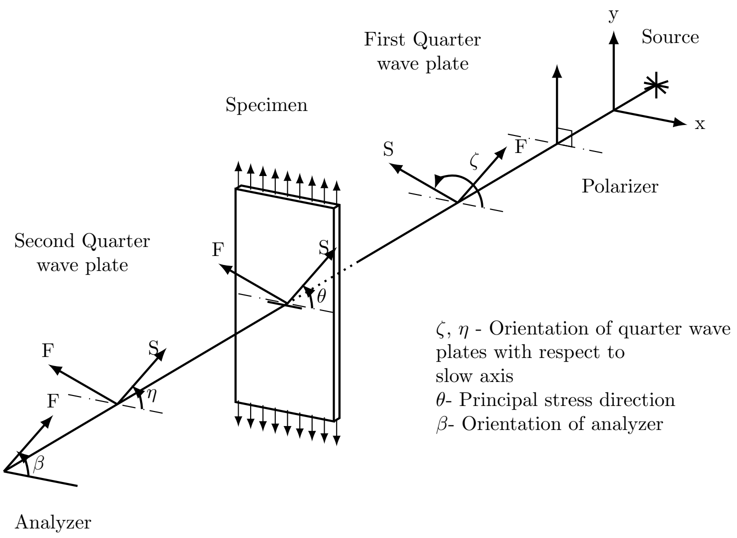


Figure 3.2: Generic photoelastic experimental arrangement.

an appropriate location. This glass rod will act as an inclusion embedded in an epoxy matrix (See Fig. 3.1c) making it predominantly a mode-I problem.

The polariscope arrangement consists of a light source, a polarizer, a first quarter-wave plate, the specimen, a second quarter-wave plate, and an analyzer. Figure (3.2) shows the schematic of photoelasticity setup being used for obtaining ten-step PST images.

3.2.2 Ten-step method

The whole field isoclinic and isochromatic data is estimated using the ten-step PST technique [34]. Recently, Ramji and Prasath [12] have done an error analysis to find the effectiveness of ten-step methodology for photoelastic parameter estimation and they found to be robust against various error sources. The optical arrangements of the ten-step method are shown in Table 3.1. The first four steps correspond to the optical arrangements of the plane polariscope which are used to generate the wrapped isoclinic phasemap. The next six arrangements correspond to the circular polariscope arrangement being used for isochromatic phasemap generation.

The isoclinic parameter (θ_c) is obtained as

$$\theta_c = \frac{1}{4}\tan^{-1}\left(\frac{I_4 - I_2}{I_3 - I_1}\right), \tag{3.1}$$

where θ_c lies in the range $-\pi/4$ to $+\pi/4$. The isoclinic phasemap values from equation 3.1 has

Table 3.1: Ten-step method: Polariscope arrangements and intensity equations for isoclinic and isochromatic evaluation

α	ζ	η	β	Intensity Equation
$\frac{\pi}{2}$	-	-	0	$I_1 = I_b + I_a \sin^2(\frac{\delta}{2}) \sin^2 2\theta$
$\frac{5\pi}{8}$	-	-	$\frac{\pi}{8}$	$I_2 = I_b + \frac{I_a}{2} \sin^2(\frac{\delta}{2})(1 - \sin 4\theta)$
$\frac{3\pi}{4}$	-	-	$\frac{\pi}{4}$	$I_3 = I_b + I_a \sin^2(\frac{\delta}{2}) \cos^2 2\theta$
$\frac{7\pi}{8}$	-	-	$\frac{3\pi}{8}$	$I_4 = I_b + \frac{I_a}{2} \sin^2(\frac{\delta}{2})(1 + \sin 4\theta)$
$\frac{\pi}{2}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{2}$	$I_5 = I_b + \frac{I_a}{2}(1 + \cos\delta)$
$\frac{\pi}{2}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	0	$I_6 = I_b + \frac{I_a}{2}(1 - \cos\delta)$
$\frac{\pi}{2}$	$\frac{3\pi}{4}$	0	0	$I_7 = I_b + \frac{I_a}{2}(1 - \sin 2\theta \sin\delta)$
$\frac{\pi}{2}$	$\frac{3\pi}{4}$	$\frac{\pi}{4}$	$\frac{\pi}{4}$	$I_8 = I_b + \frac{I_a}{2}(1 + \cos 2\theta \sin\delta)$
$\frac{\pi}{2}$	$\frac{3\pi}{4}$	0	0	$I_9 = I_b + \frac{I_a}{2}(1 + \sin 2\theta \sin\delta)$
$\frac{\pi}{2}$	$\frac{3\pi}{4}$	$\frac{3\pi}{4}$	$\frac{3\pi}{4}$	$I_{10} = I_b + \frac{I_a}{2}(1 - \cos 2\theta \sin\delta)$

to be unwrapped by AQGPU algorithm to remove any inconsistent zone thereby forcing theta in the range $-\pi/2$ to $+\pi/2$. The unwrapped values of isoclinic (θ_{uw}) are used in equation 3.2 for getting the isochromatic phasemap without any ambiguous zones [1]. The isochromatic value is obtained by

$$\delta_c = \tan^{-1}\left(\frac{(I_9 - I_7)\sin 2\theta_{uw} + (I_8 - I_{10})\cos 2\theta_{uw}}{I_5 - I_6}\right) \tag{3.2}$$

Finally, the isochromatic phasemap is unwrapped using the same AQGPU algorithm to get the continuous fringe order. The data of isoclinic and fringe order will be given as input for estimating the crack tip SIF parameter.

3.2.3 Experimental analysis

Figure 3.3 shows the experimental arrangement of transmission photoelastic setup used in this study. The ten-step phase shifted images are recorded using BASLER monochrome CCD (charged coupled device) camera for the optical arrangements shown in Table 2.1. The CCD camera has a resolution of 1392 x 1040. The specimens are loaded using the Instron 5600 machine of 10 kN capacity. Material stress fringe value of the epoxy specimen (F_σ) is 11 N/mm/fringe. The whole field fringe order and isoclinic values surrounding the crack tip are required to estimate the SIF. By using ten-step method, isoclinic data is firstly generated and it is further unwrapped to get the wrapped isochromatic phasemap without any ambiguity.

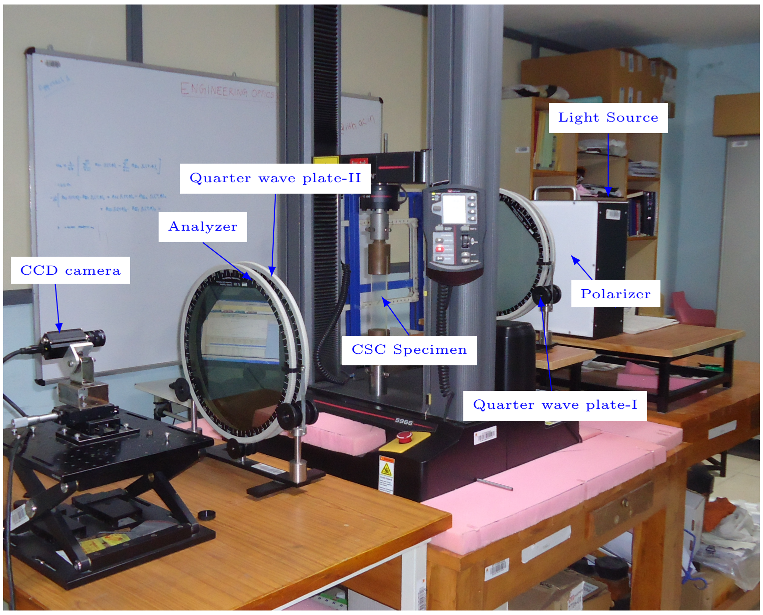


Figure 3.3: Experimental setup for SIF estimation involving digital photoelasticity.

The AQGPU algorithm [1] is used to unwrap both isoclinic and isochromatic parameter.

Figure 3.4a shows the wrapped isoclinic phasemap and it is unwrapped to remove the inconsistent zone as shown in Fig. 3.4b for SEN specimen for a tensile load of 705 N. The unwrapped isoclinic is used to get the isochromatic phasemap without any ambiguity and it is shown in Fig. 3.4c. To verify the accuracy, this phasemap is compared with dark field photoelastic fringe contours as shown in Fig. 3.4d. The wrapped isochromatic phasemap is then unwrapped to get the total fringe order over the model domain and it is shown in Fig. 3.4e. Similarly, isoclinic and isochromatic phasemaps are shown for a crack-inclusion specimen with a tensile load of 640 N in Fig. 3.6. Figure 3.5 shows the isoclinic and isochromatic phasemap for a CSC specimen obtained at a tensile load of 600 N.

3.3 Experimental evaluation of Mixed mode fracture parameters

In this section, a methodology to estimate the fracture parameters from digital photoelastic experiment is discussed. The photoelastic parameter around the tip of crack is collected automatically with the help of software interface developed using MATLAB which is linked to ten-step phase data. It extracts fringe order (N), principal stress direction (θ_{uw}) (unwrapped

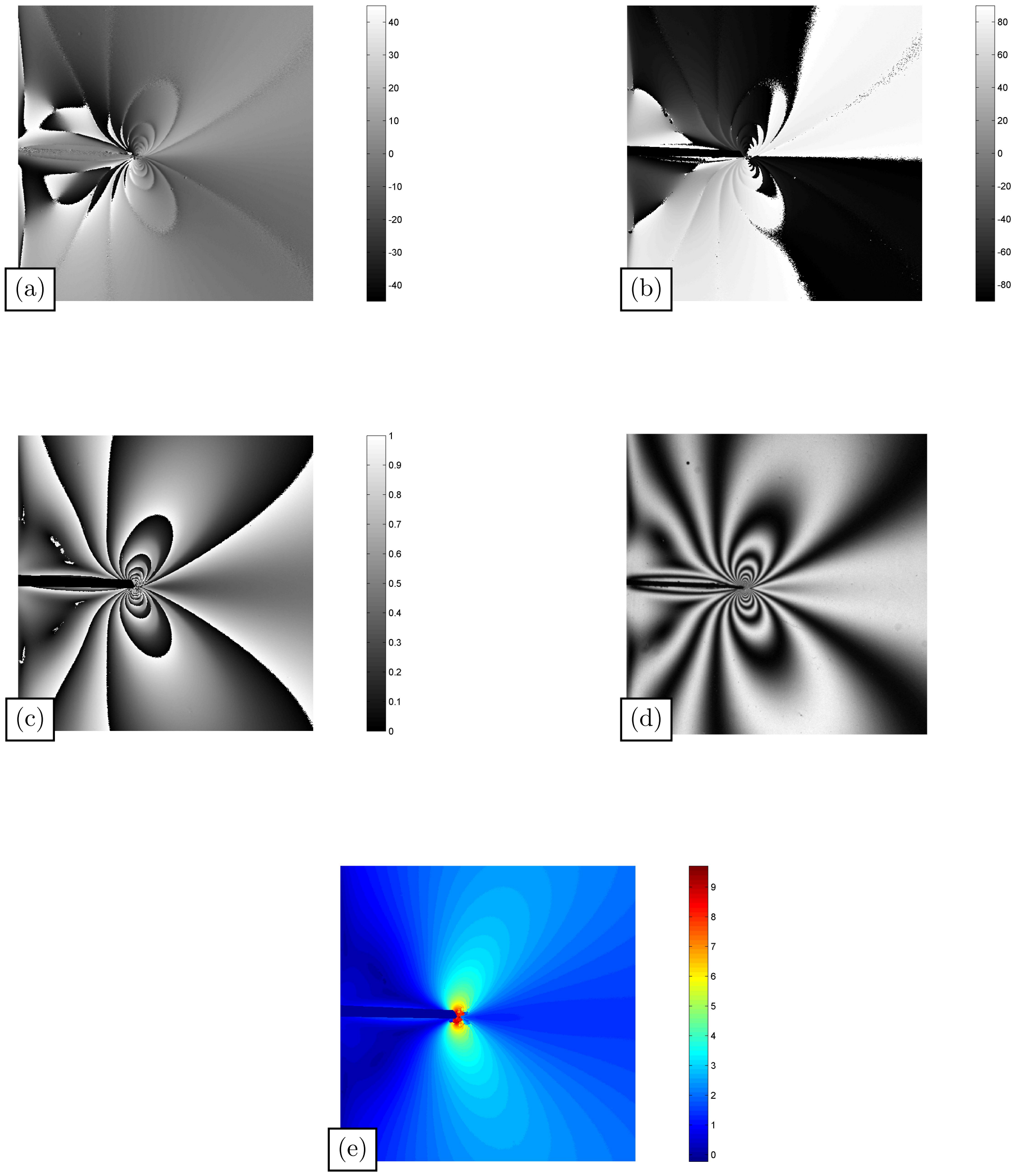


Figure 3.4: Isoclinic and isochromatic phasemap for SEN specimen obtained at a tesile load of 705 N (a) wrapped isoclinic (b) unwrapped isoclinic (c) wrapped isochromatic (d) dark field photoelastic image (e) unwrapped isochromatic.

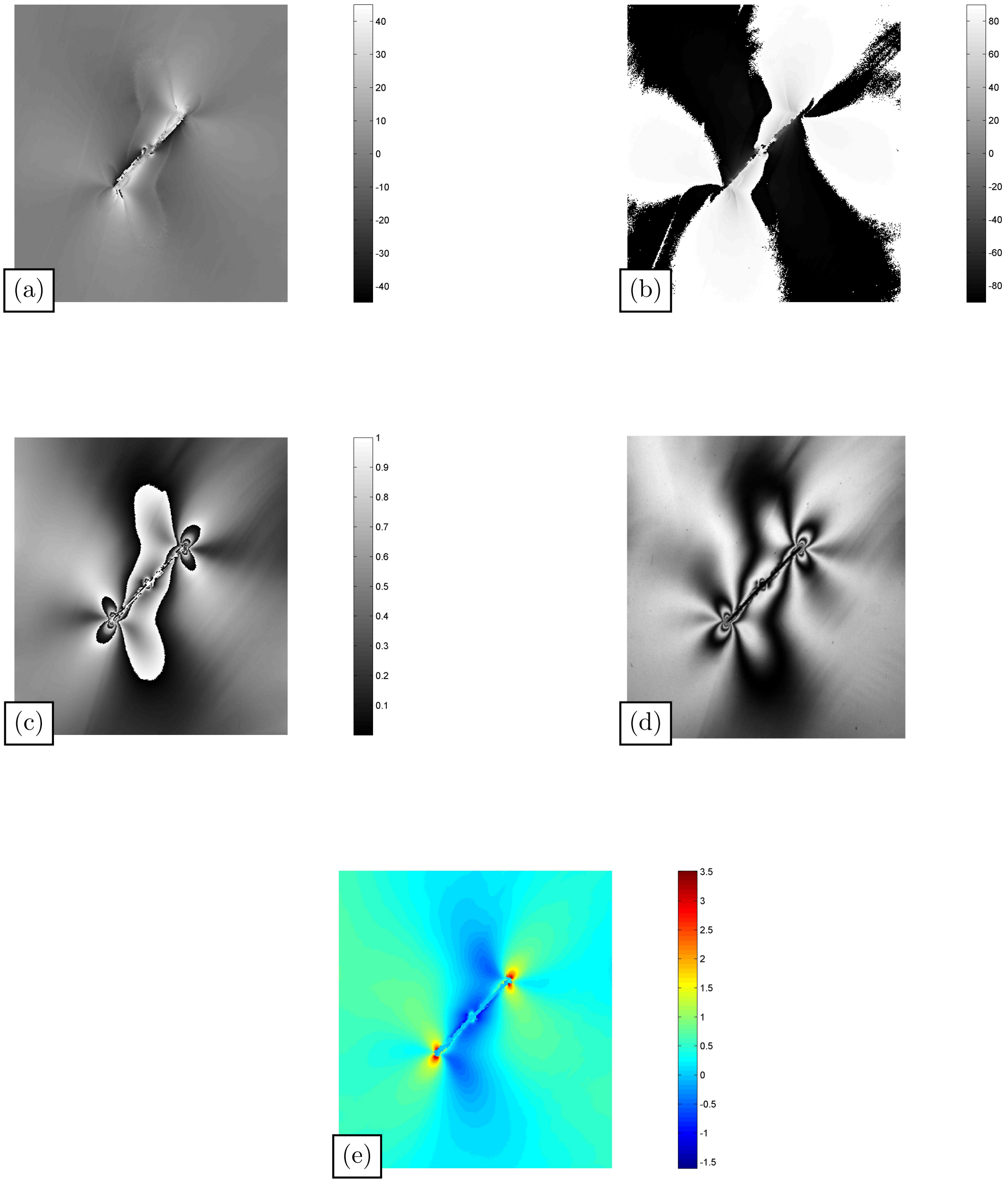


Figure 3.5: Isoclinic and isochromatic phasemap for CSC specimen obtained at a tensile load of 640 N (a) wrapped isoclinic (b) unwrapped isoclinic (c) wrapped isochromatic (d) dark field photoelastic image (e) unwrapped isochromatic.

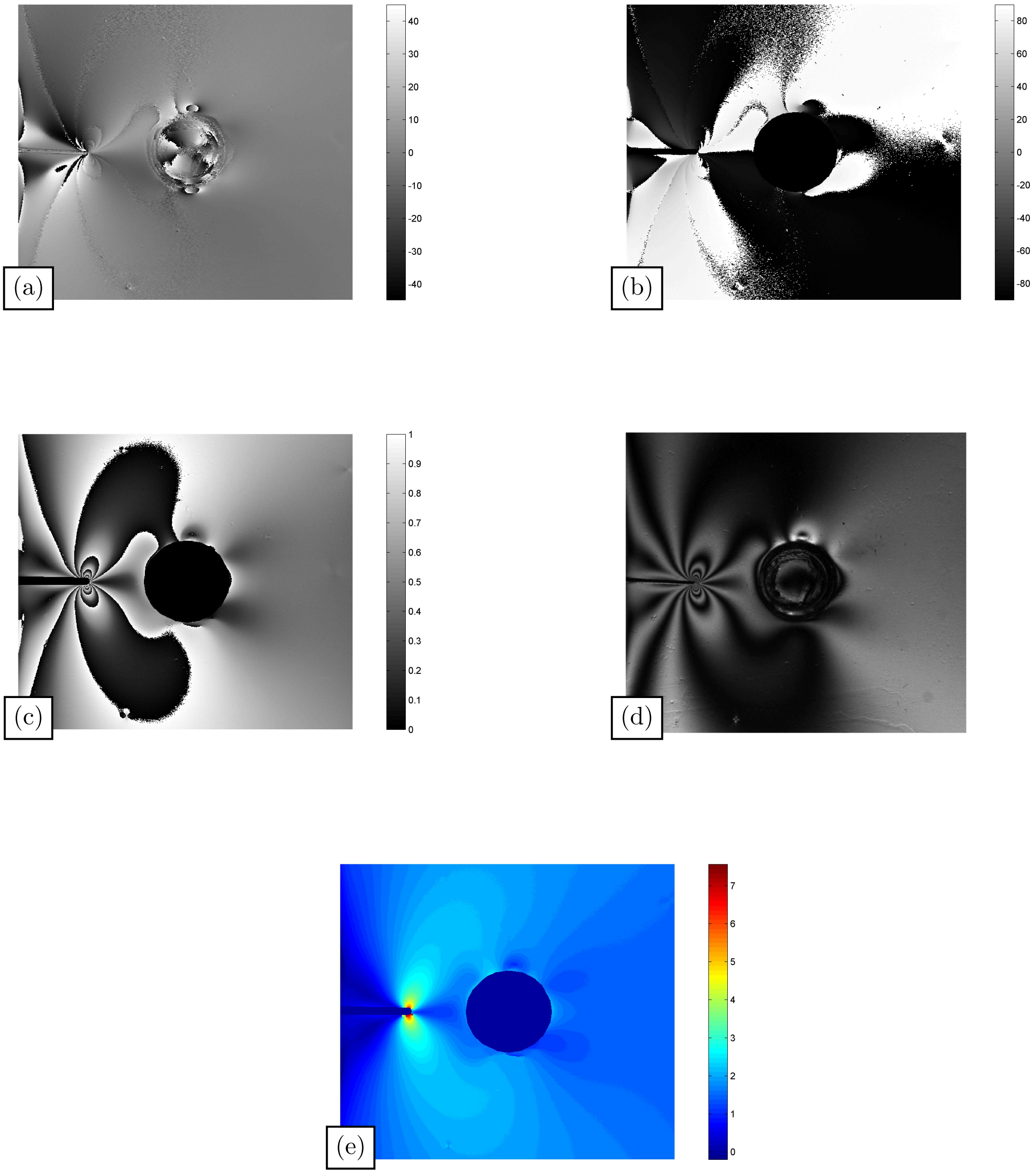


Figure 3.6: Isoclinic and isochromatic phasemap for crack-inclusion specimen obtained at a tensile load of 640 N (a) wrapped isoclinic (b) unwrapped isoclinic (c) wrapped isochromatic (d) dark field photoelastic image (e) unwrapped isochromatic (the inclusion part is carved out for clarity).

data), the coordinates of each collected point and crack tip location. The data collection zone is annular and its minimum radius is chosen so as to avoid the plastic zone [36] and three-dimensional stress effect nearer to the crack tip [37]. Figure 3.7 shows the annular zone considered for the data collection. The fringe order is related to principal stress difference by the stress optics law as given below:

$$\sigma_1 - \sigma_2 = \frac{NF_\sigma}{h}, \tag{3.3}$$

where N is the fringe order, F_σ is the material stress fringe value and h is the thickness of specimen. The normal stress component difference and shear stress are defined in terms of principal stress difference and principal stress direction as given below using Mohr's circle

$$\begin{Bmatrix} \sigma_x - \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{Bmatrix} (\sigma_1 - \sigma_2)\cos 2\theta_{uw} \\ \frac{\sigma_1 - \sigma_2}{2}\sin 2\theta_{uw} \end{Bmatrix}. \tag{3.4}$$

3.3.1 Multi-parameter stress field equations

The multi-parameter stress field equation for mixed mode crack are reported in Ramesh et al. [17] . These stress field equations in general form are shown below

$$\boldsymbol{\phi} = \sum_{n=1}^{\infty} \frac{nA_{In}}{2} r^{\frac{n-2}{2}} \boldsymbol{M} - \sum_{n=1}^{\infty} \frac{nA_{IIn}}{2} r^{\frac{n-2}{2}} \boldsymbol{N}, \tag{3.5}$$

where

$$\boldsymbol{\phi} = \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} \tag{3.6}$$

$$\boldsymbol{M} = \begin{Bmatrix} (2 + (-1)^n + \frac{n}{2})\cos s\theta - s\cos t\theta \\ (2 - (-1)^n - \frac{n}{2})\cos s\theta + s\cos t\theta \\ -((-1)^n + \frac{n}{2})\sin s\theta + s\sin t\theta \end{Bmatrix} \tag{3.7}$$

$$\boldsymbol{N} = \begin{Bmatrix} (2 - (-1)^n + \frac{n}{2})\sin s\theta - s\sin t\theta \\ (2 + (-1)^n - \frac{n}{2})\sin s\theta + s\sin t\theta \\ -((-1)^n - \frac{n}{2})\cos s\theta + s\cos t\theta, \end{Bmatrix} \tag{3.8}$$

where $s = (n/2) - 1$, $t = (n/2) - 3$, σ_x and σ_y are the normal stress component along x and y directions respectively, τ_{xy} is in plane shear stress, r and θ are the polar coordinates with origin at the crack tip and n is the number of parameters. The coefficients A_{In} and A_{IIn} define crack tip stress field and they are related to SIF as $A_{I1} = K_I/\sqrt{2\pi}$ and $A_{II1} = -K_{II}/\sqrt{2\pi}$. The T-stress term ($\sigma_{0x}$) also plays a very critical role for tilting the crack tip fringe contour ahead of the crack tip. The σ_{0x} is related to second term of mode-I series as $A_{I2} = -\sigma_{0x}/4$.

3.3.2 Formulation for linear approach

One can rewrite the equation (3.5) in general form as follows

$$\sigma_x - \sigma_y = \sum_{n=1}^{\infty} A_{In} f_{In}(r,\theta) - \sum_{n=1}^{\infty} A_{IIn} f_{IIn}(r,\theta) \tag{3.9}$$

$$\tau_{xy} = \sum_{n=1}^{\infty} A_{In} g_{In}(r,\theta) - \sum_{n=1}^{\infty} A_{IIn} g_{IIn}(r,\theta), \tag{3.10}$$

where

$$f_{In} = r^{\frac{n-2}{2}}\{2((-1)^n + \frac{n}{2})\cos s\theta - 2s\cos t\theta\} \tag{3.11}$$

$$f_{IIn} = r^{\frac{n-2}{2}}\{2(-(-1)^n + \frac{n}{2})\sin s\theta - 2s\sin t\theta\} \tag{3.12}$$

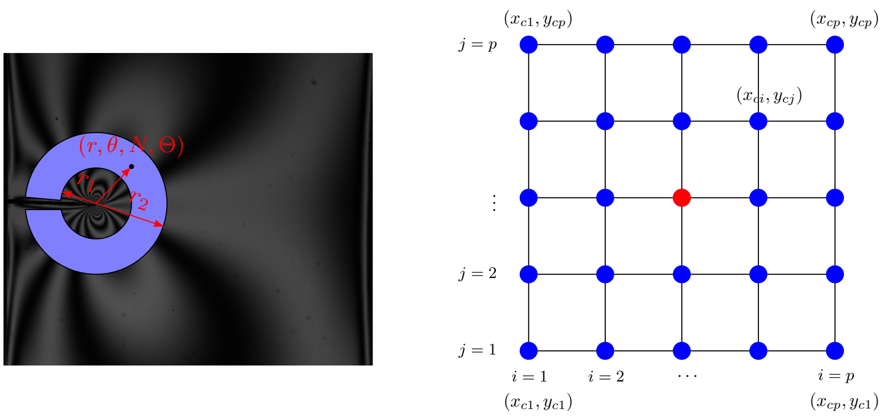
$$g_{In} = r^{\frac{n-2}{2}}\{-((-1)^n + \frac{n}{2})\sin s\theta + s\sin t\theta\} \tag{3.13}$$

$$g_{IIn} = r^{\frac{n-2}{2}}\{-((-1)^n - \frac{n}{2})\cos s\theta + s\cos t\theta\}. \tag{3.14}$$

The terms f_{In}, f_{IIn}, g_{In} and g_{IIn} are functions of number of parameters n, co-ordinates r and θ. In many cases, it is difficult to select the exact location of the crack-tip from the photoelsticity images being captured as the spatial resolution is very low. Therefore, always there is a certain amount of uncertainty getting associated with SIF estimation. To circumvent this problem the crack tip location (x_c, y_c) can also be treated as unknown parameters along with the coefficients A_{In} and A_{IIn} in equations (3.9) and (3.10). Consider the point (x, y) in cartesian coordinate having origin at arbitrary location. The crack tip location is related to r and θ as follows

$$r = \sqrt{(x - x_c)^2 + (y - y_c)^2} \tag{3.15}$$

$$\theta = \tan^{-1}\left(\frac{y - y_c}{x - x_c}\right), \tag{3.16}$$



(a) (b)

Figure 3.7: Zone of data collection and the corresponding square grid. (a) Schematic diagram of the annular region used in data collection for SIF estimation in case of SEN specimen (b) schematic representation of the square grid pattern used for obtaining the optimal crack tip location.

where, x_c and y_c are the locations of the crack tip relative to an arbitrary cartesian coordinate system. This equation allow us to translate coordinate system with respect to the crack tip. The number of parameters n in multi-parameter stress field equation 3.9 and 3.10 are limited to finite number for computational purpose. For a single point p, the truncated n parameter equations 3.9 and 3.10 can be written in a matrix form as given below

$$\underbrace{\begin{Bmatrix} \sigma_x - \sigma_y \\ \tau_{xy} \end{Bmatrix}}_{\boldsymbol{\sigma_p}} = \underbrace{\begin{bmatrix} f_{I1}(r_p, \theta_p) & g_{I1}(r_p, \theta_p) \\ f_{I2}(r_p, \theta_p) & g_{I2}(r_p, \theta_p) \\ \vdots & \vdots \\ f_{In}(r_p, \theta_p) & g_{In}(r_p, \theta_p) \\ -f_{II1}(r_p, \theta_p) & -g_{II1}(r_p, \theta_p) \\ -f_{II2}(r_p, \theta_p) & -g_{II2}(r_p, \theta_p) \\ \vdots & \vdots \\ -f_{IIn}(r_p, \theta_p) & -g_{IIn}(r_p, \theta_p) \end{bmatrix}^{\mathrm{T}}}_{\boldsymbol{Q_p}} \underbrace{\begin{Bmatrix} A_{I1} \\ A_{I2} \\ \vdots \\ A_{In} \\ A_{II1} \\ A_{II2} \\ \vdots \\ A_{IIn} \end{Bmatrix}}_{\boldsymbol{a}}. \tag{3.17}$$

The rewritten equation in matrix is shown below as

$$\boldsymbol{\sigma_p} = \ \boldsymbol{Q_p^{\mathrm{T}}} \ \boldsymbol{a}. \tag{3.18}$$

Extending the equation 3.17 for m collected data points surrounding the crack tip, the solution in terms of matrix with n parameter can be written as

$$\boldsymbol{\sigma} = \mathbf{C}\left(x_c, y_c\right)\boldsymbol{a}, \tag{3.19}$$

where $\boldsymbol{\sigma} = \left[\boldsymbol{\sigma_1^T}\ \boldsymbol{\sigma_2^T}\ \ldots\ \boldsymbol{\sigma_m^T}\right]^{\mathrm{T}}$ and $\mathbf{C} = \left[\boldsymbol{Q_1^T}\ \boldsymbol{Q_2^T}\ \ldots\ \boldsymbol{Q_m^T}\right]^{\mathrm{T}}$. Here, $\boldsymbol{\sigma}$ is the vector consisting of the experimental data as estimated from equation 3.9. The matrix $\mathbf{C}$ is dependent on x_c and y_c which is a rectangular matrix of the order $2m \times 2n$ and $\boldsymbol{a}$ is the vector consisting of unknown mode I and mode II parameters . The values of x_c, y_c and $\boldsymbol{a}$ will be estimated by minimizing the objective function:

$$J(x_c, y_c, \boldsymbol{a}) = \frac{1}{2}(\boldsymbol{\sigma} - \mathbf{C}\left(x_c, y_c\right)\boldsymbol{a})^{\mathrm{T}}(\boldsymbol{\sigma} - \boldsymbol{C}\left(x_c, y_c\right)\boldsymbol{a}). \tag{3.20}$$

The objective function J is of non quadratic form for stress in terms of unknown parameters. It depends on the unknown crack tip coordinates x_c and y_c. But when x_c and y_c are known, J becomes quadratic and a closed form solution do exist for it. Here, to estimate J both normal stress component difference and shear stress are considered. By considering only normal stress component difference or shear stress leads to SIF value colser to actual value but the reconstructed fringe pattern does not match with experimental fringe data. This would crate difficulty in choosing number of parameters for multi-parameter stress field equation. The closed form solution for the unknown vector of parameters ($\boldsymbol{a}$), where the objective function has a global minimum is as follow:

$$\boldsymbol{a} = (\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{C}^{\mathrm{T}}\boldsymbol{\sigma}, \tag{3.21}$$

where $(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{C}^{\mathrm{T}}$ is the pseudo inverse of $\mathbf{C}$. We select multiple $(x_{ci}, y_{cj}), i = 1, 2, .., p, j = 1, 2, ..., p$ locations around the crack tip as shown in Fig. 3.7 and for each of these location we obtain the unknown parameters $\mathbf{a}_{ij}$ using equation 3.21. For every (x_{ci}, y_{cj}), having known $\mathbf{a}_i$ we calculate J_{ij}. Out of all the grid points (see Fig. 3.7b) we select the crack tip location $(x_c^*, y_c^*) = (x_{ci}, y_{cj})$ and unknown parameters $\mathbf{a}^* = \mathbf{a}_{ij}$ which corresponds to the location (x_{ci}, y_{cj}) at which J_{ij} attains the lowest value. Mathematically our idea to find the optimal fracture parameters and crack tip location can be represented as follows:

$$[\boldsymbol{a}^{*\mathrm{T}} \quad x_c^* \quad y_c^*]^{\mathrm{T}} = \arg\min\left[\min\left(J_{ij}\right)\right], \tag{3.22}$$

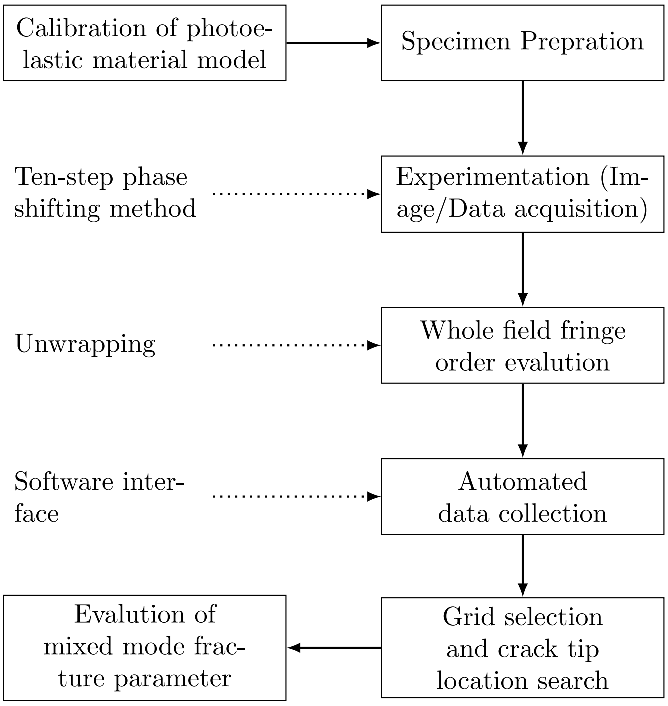

Calibration of photoelastic material model
Specimen Prepration
Ten-step phase shifting method
Experimentation (Image/Data acquisition)
Unwrapping
Whole field fringe order evalution
Software interface
Automated data collection
Evalution of mixed mode fracture parameter
Grid selection and crack tip location search

Figure 3.8: Flowchart showing steps to evaluate fracture parameters using digital photoelasticity.

where $i = 1, 2, ..., p$, $j = 1, 2, ..., p$, x_c and y_c are the crack tip coordinates, x_{c1} and x_{cp} are the minimum and maximum x coordinate values of the square grid, y_{c1} and y_{cp} are the minimum and maximum y coordinate values of the square grid, r_i and r_o are the inner and outer radii of data collection zone as shown in Fig. 3.7, n is the number of unknown parameters (see equation 3.17). The algorithm to get the fracture parameter vector $\boldsymbol{a}$ is summarised in Fig. 3.8.

3.4 Results and Discussions

The fringe order and isoclinic data are extracted using a in-house software developed in MATLAB for digital photoelasticity applications. It is based on the ten-step phase shifting method and AQGPU algorithm [1] is adapoted for unwrapping the phasemaps. The collected data surrounding the crack tip for SEN, CSC and crack-inclusion specimens are extracted and they are given as input for estimating mixed mode fracture parameters.

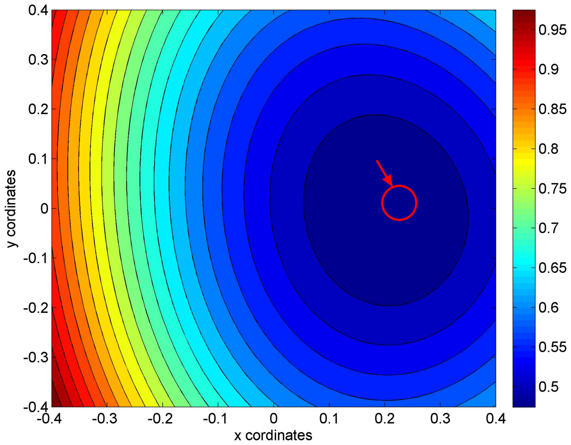


Figure 3.9: Normalized error plot for experimental estimation of SIFs for SEN specimen.

3.4.1 Experimental determination of SIF for SEN specimen

Using an in-house developed software interface, the fringe order and isoclinic data are collected in the annular zone around the crack tip for the SEN specimen subjected to a tensile load of 705 N. Also it contains pixel coordinates for each selected data point along with the crack tip location which is approximately selected by the user. The selected crack tip acts as origin of the coordinate system, a square of 0.4 mm length and a grid size of 0.01 mm is created. For each grid point, the value of **a** is computed. The value of **a** is chosen such way that J has a minimum value in the grid and the corresponding K_I is estimated. Figure 3.9 shows the normalized error plot with respect to selected crack tip where the zone of least error is marked corresponding to the exact crack tip location.

The experimentally evaluated value of K_I for SEN specimen is 23.42 MPa$\sqrt{\text{mm}}$ with 7-parameters. The corresponding value of J is 0.09 MPa^2. The new location for crack tip based on minimum error is $(0.20, 0.00)$ with respect to manually selected crack tip location. The SIF value for SEN specimen is calculated from analytical expression [38] as given below:

$$K_I = \sigma\sqrt{\pi a}F(\beta), \tag{3.23}$$

where σ is the far field stress, a is the crack length. $F(\beta)$ for SEN specimen with finite geometry is expressed as

$$F(\beta) = 1.12 - 0.231\beta + 10.550\beta^2 - 21.710\beta^3 + 30.382\beta^4, \tag{3.24}$$

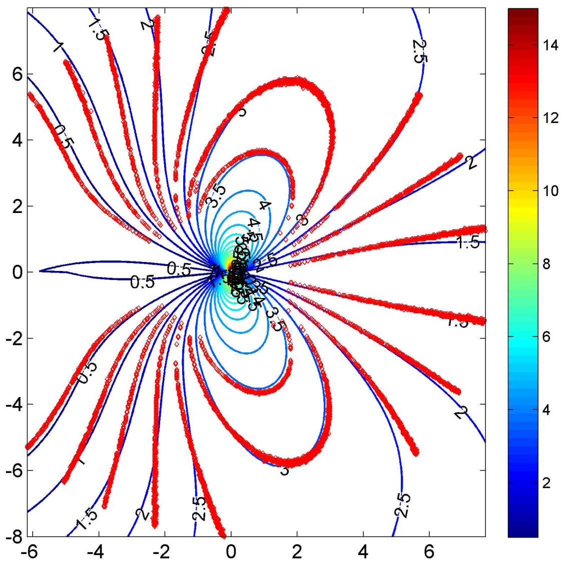


Figure 3.10: Experimental (red markers) and reconstructed principal stress difference field (fringe contour) obtained for 7 parameter solution superposed over each other in case of SEN specimen being subjected to a tensile load of 705 N.

where β is crack length to width ratio (a/w). In this case $\beta = 0.25$. The analytical SIF value calculated based on equation 3.24 is 24.70 MPa$\sqrt{\text{mm}}$. The experimental SIF value has got a deviation of 5.3% with respect to the analytical value. The values of different parameters for 2, 4 and 7 parameter solution for SEN specimen are summarised in Table 3.2. For confirming the accuracy of results, the reconstructed and experimental (red markers) fringes for 7-parameters are superimposed over each other as shown in Fig. 3.10. Here, it is confirmed that the 7-parameter solution has got a good match with the experimental field.

3.4.2 Experimental determination of SIF for CSC specimen

The data collection for CSC specimen is done in the same manner as explained in section 3.4.1 but the annular region is confined to $0.2 < r/a < 0.8$, where a is half slant crack length. Figure 3.11 shows the normalized error plot obtained with respect to the selected crack tip location and zone of least error is marked which corresponds to the exact crack tip location. The CSC specimen is subjected to a tensile load of 600 N.

The mixed mode values K_I and K_{II} for the CSC specimen are calculated from the analytical

Table 3.2: Crack tip fracture parameters for SEN specimen

	2-parameter	4-parameter	7-parameter
K_I (MPa$\sqrt{\text{mm}}$)	23.26	23.40	23.37
K_{II} (MPa$\sqrt{\text{mm}}$)	0.19	0.12	0.10
A_{I1} (MPa(mm)$^{1/2}$)	9.3361	9.3903	9.3433
A_{I2} (MPa)	-0.4904	-0.4960	-0.4753
A_{I3} (MPa(mm)$^{-1/2}$)		0.0287	0.1117
A_{I4} (MPa(mm)$^{-1}$)		-0.0084	-0.0179
A_{I5} (MPa(mm)$^{-3/2}$)			0.0016
A_{I6} (MPa(mm)$^{-2}$)			0.0006
A_{I7} (MPa(mm)$^{-5/2}$)			-0.0002

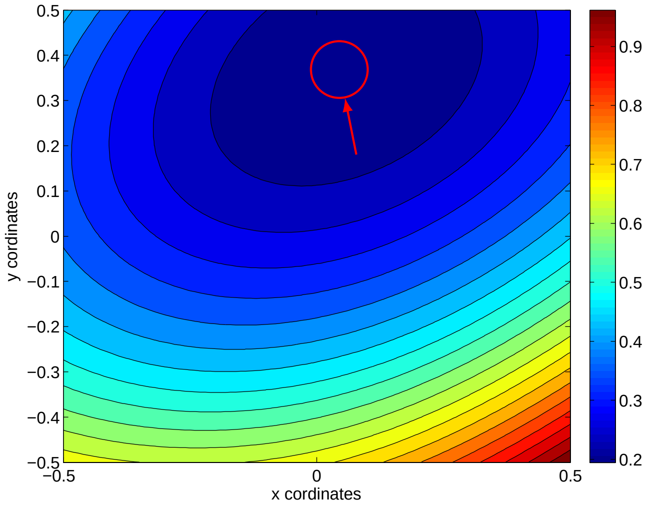


Figure 3.11: Normalized error plot for experimental estimation of SIFs for CSC specimen.

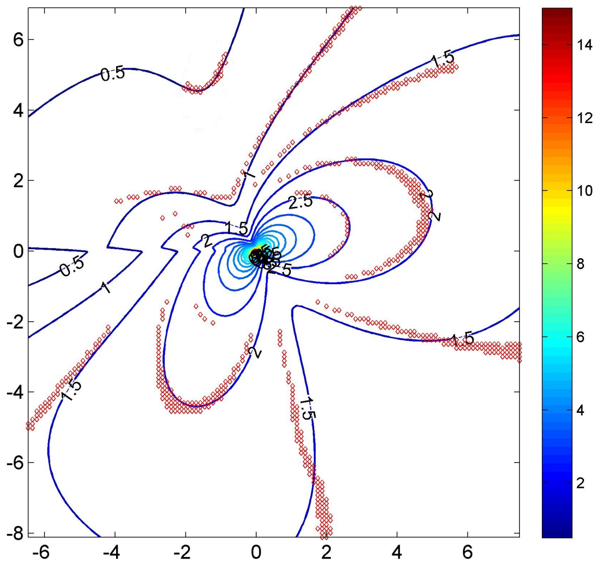


Figure 3.12: Experimental (red markers) and reconstructed principal stress difference fields (fringe contour) obtained for 10 parameter solution superposed over each other in case of CSC specimen for a tensile load of 600 N.

expressions [39] as

$$K_I = \sigma\sqrt{\pi a}F_I(\beta), \tag{3.25}$$

$$K_{II} = \sigma\sqrt{\pi a}F_{II}(\beta). \tag{3.26}$$

From Ref. [39], the F_I and F_{II} values are 0.5594 and 0.5239 for 45° center inclined crack for $a/w = 0.36$. The analytical values of K_I and K_{II} obtained using equation 3.26 are 5.97 MPa$\sqrt{\text{mm}}$ and 5.60 MPa$\sqrt{\text{mm}}$ respectively. The experimentally estimated values of K_I and K_{II} are 5.45 MPa$\sqrt{\text{mm}}$ and 5.96 MPa$\sqrt{\text{mm}}$ respectively corresponding to 10-parameters. The 10-parameters solution is decided based on the overlapping of reconstructed and experimental fringe pattern as shown in Fig. 3.12. The experimental K_I and K_{II} are found be close to the analytical estimate with an error of 8.4% and 9.4% respectively. This deviation could arise due to various error sources like less accurate photoelastic parameter determination, not so perfect straight crack front along with slight deviation in inclination, interaction of hole stresses with crack tip etc.,The coordinates of the new location of crack tip is $(0.06, 0.36)$ from the manually selected crack tip location.

Table 3.3: Crack tip fracture parameters obtained for the CSC specimen

	2-parameter	4-parameter	7-parameter
K_I (MPa$\sqrt{\text{mm}}$)	4.53	5.44	5.47
K_{II} (MPa$\sqrt{\text{mm}}$)	4.52	5.87	5.97
A_{I1} (MPa(mm)$^{1/2}$)	2.0089	2.4105	2.4265
A_{I2} (MPa)	-0.0027	-0.0202	-0.0422
A_{I3} (MPa(mm)$^{-1/2}$)		0.0375	0.0224
A_{I4} (MPa(mm)$^{-1}$)		0.0005	0.0031
A_{I5} (MPa(mm)$^{-3/2}$)			-0.0008
A_{I6} (MPa(mm)$^{-2}$)			-0.0001
A_{I7} (MPa(mm)$^{-5/2}$)			0.0000
A_{II1} (MPa(mm)$^{1/2}$)	-2.0035	-2.6016	-2.6453
A_{II2} (MPa)	0.0000	0.0000	0.0000
A_{II3} (MPa(mm)$^{-1/2}$)		-0.0717	-0.0855
A_{II4} (MPa(mm)$^{-1}$)		0.0052	0.0037
A_{II5} (MPa(mm)$^{-3/2}$)			0.0020
A_{II6} (MPa(mm)$^{-2}$)			-0.0012
A_{II7} (MPa(mm)$^{-5/2}$)			0.0001

3.4.3 SIF estimation for crack-inclusion specimen

The proposed SIF estimation methodology is also extended for the crack-inclusion interaction problem. Figure 3.13 shows the normalized error plot with respect to selected crack tip location and zone of least error is marked which corresponds to exact crack tip location. Experimentally estimated value of K_I is 18.5 MPa$\sqrt{\text{mm}}$ corresponding to a tensile load of 640 N. Here, 7-parameter solution is found to be sufficient as the reconstructed fringe pattern coincides with the experimental fringe pattern (see Fig. 3.14).

For quantitative comparison, the SIF value for the same problem is also calculated using the FEA. In this work, 2D analysis of crack-inclusion specimen is carried out using ABAQUS. The analysis is done with 8-noded plane strain elements. The model is of dimension 200 mm x 45 mm with 5 mm glass inclusion having an edge crack has 8 mm length (see Fig. 3.1c). The mesh pattern around the crack tip is kept very fine in order to capture high-stress gradient. The quarter point element is used to capture square root singularity at the crack tip. The mesh convergence is achieved with 80 elements in circumferential and 60 along the radial direction. Figure 3.15 shows the zoomed view of the mesh surrounding the crack tip as well as glass inclusion. Away from the crack-inclusion interaction zone, a coarser mesh is being used in order to reduce the degrees of freedom. In total, there are 9294 elements corresponding to

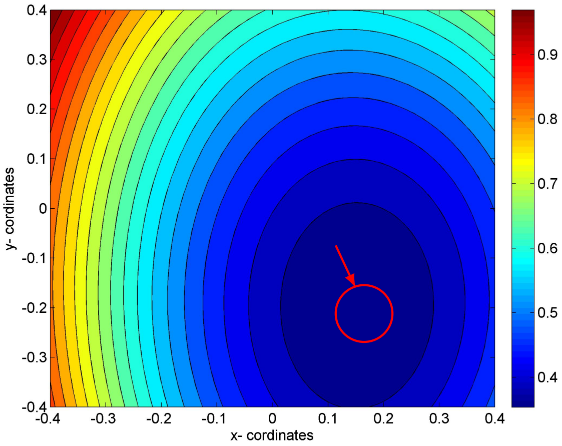


Figure 3.13: Normalized error plot obtained for the experimental estimation of SIFs for crack-inclusion specimen.

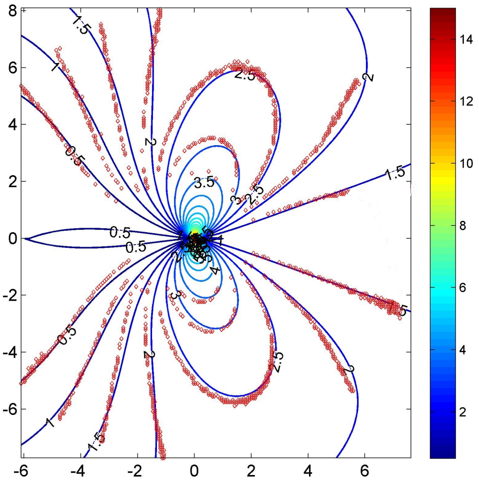


Figure 3.14: Experimental (red markers) and reconstructed principal stress difference fields (fringe contour) obtained for 7 parameter solution superposed over each other in case of crack-inclusion specimen being subjected to a tensile load of 640 N.

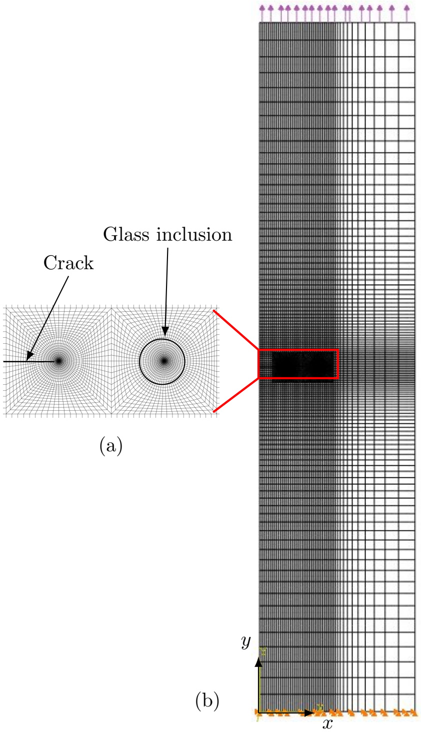


Figure 3.15: Finite element model for the crack-inclusion specimen (a) Zoomed view of the crack tip and glass inclusion (b) complete model.

56718 degrees of freedom. The interface between glass inclusion and the matrix is modeled with fine mesh and a perfect matrix inclusion bonding is considered. The degrees of freedom along the bottom edge $y = 0$ is arrested and a tensile load of 640 N magnitude is applied on the top edge. The SIF value for crack-inclusion specimen obtained from FEA is 19.67 MPa$\sqrt{\text{mm}}$. The experimentally evaluated SIF value has percentage deviation of 5.9 % with respect FEA evaluated value. The fracture parameters obtained for 2, 4 and 7 parameter solution for the crack-inclusion specimen are summarised in Table 3.4 .

The SIF values for SEN, crack-inclusion and CSC specimens determined above are solved for 10 parameters for the comparative study. The number of parameters needed for the accurate estimate of SIF is decided by a convergence study. It has been found that J (see equation (3.20)) attains a constant value after iteratively increasing the number of parameters beyond 7 for SEN, crack-inclusion and CSC specimens as shown in Fig. 3.16.

But the number of parameters in a multi-parameter equation are decided based on a matching of the reconstructed contours with the experimental contour along with the convergence of error. With the increase in a number of parameters, the SIF values also converge and further increase in parameter values beyond 7, there is no significant change in SIF value. This is true

Table 3.4: Crack tip fracture parameters obtained for the crack-inclusion specimen

	2-parameter	4-parameter	7-parameter
K_I (MPa$\sqrt{\text{mm}}$)	18.18	18.30	18.50
K_{II} (MPa$\sqrt{\text{mm}}$)	0.09	0.07	0.05
A_{I1} (MPa(mm)$^{1/2}$)	7.2517	7.3013	7.3801
A_{I2} (MPa)	-0.4596	-0.4525	-0.4321
A_{I3} (MPa(mm)$^{-1/2}$)		0.0072	0.0607
A_{I4} (MPa(mm)$^{-1}$)		-0.0058	-0.0136
A_{I5} (MPa(mm)$^{-3/2}$)			0.0014
A_{I6} (MPa(mm)$^{-2}$)			0.0004
A_{I7} (MPa(mm)$^{-5/2}$)			-0.0002

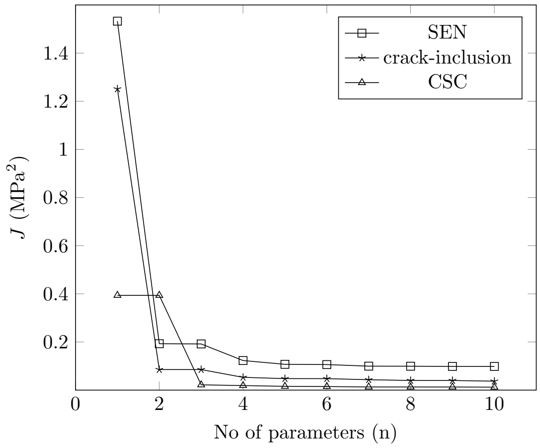
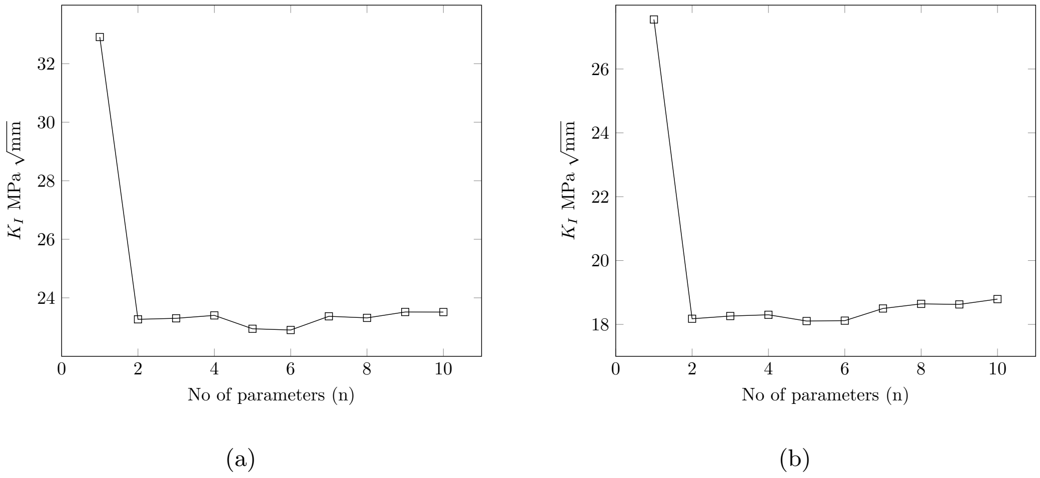


Figure 3.16: Variation of J with increasing number of parameters obtained for SEN, CSC and ,crack-inclusion specimen.



(a)

(b)

Figure 3.17: Variation of mode-I SIF with increasing number of parameters for (a) SEN specimen (b) crack-inclusion specimen.

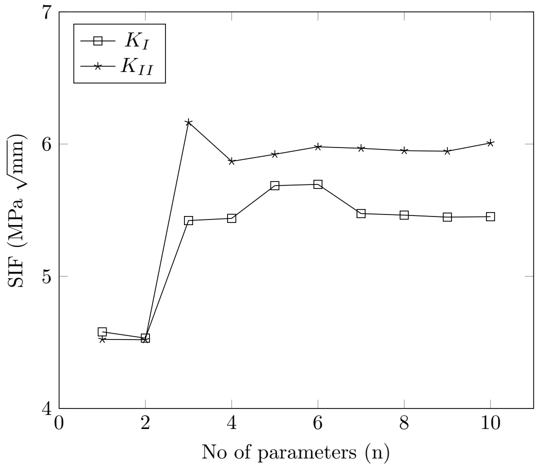


Figure 3.18: Variation of mode-I and mode-II SIF with increasing number of parameters for CSC specimen.

for all the three configurations as shown in Figures 3.17a, 3.17b and 3.18. Therefore, fringe reconstruction and matching is the primary requirement followed up by the error convergence to decide upon the number of parameters suffice for SIF estimation.

3.5 Conclusions

In this work, fracture parameters are estimated for SEN, CSC and crack-inclusion specimen configurations using the digital photoelastic technique. Full field isochromatic and isoclinic data over the cracked specimen are estimated using the ten-step phase shifting technique in tandem with AQGPU algorithm for phase unwrapping. An over deterministic linear least square approach is proposed and implemented for the first time towards SIF determination involving multi-parameter stress field equation in the digital photoelasticity domain. Linearisation is possible because of the availability of accurate isoclinic and isochromatic data over the model domain utilising the advantage of the digital photoelasticity to the fullest extent. This linear approach has got a better convergence and attains global minimum as compared to the conventional approach of non-linear over deterministic least square minimization as mentioned in the literature. The SIF values for all the three specimen configurations are found to be closely matching with the analytical or FEA estimate, thereby confirming the accuracy of proposed methodology. Still, the accuracy of SIF estimate could be improved by increasing the accuracy of the isoclinic parameter data using the white light photoelasticity thereby eliminating the isochromatic-isoclinic interaction noise.

Chapter 4

Numerical and experimental estimation of strain intensity factor for rigid line inclusion

4.1 Introduction

In recent years, composite materials find wide applications in marine, aerospace and automotive sectors. Composite materials are combinations of two or more phases, e.g., fiber and matrix. They are used for structural applications in the form of both continuous and short fiber composite structures. The continuous fiber composite are replacing metallic structural parts especially in aerospace industries, while the short fiber composites are used instead of plane polymeric material for electrical, packaging and automobile applications [22]. In short fiber composites, both fiber and matrix share the applied load, and the load transfer between the matrix and fiber happens via the fiber/matrix interface. As a consequence, the short fiber composites have superior strength and elastic stiffness over the parent polymeric material [23]. However, the fibers could also lead to singular stress field in the matrix near the tip of fiber. If micro voids are present near the inclusion tip, the singular stress field will cause void growth, coalesce and micro-cracking. Moreover, the fiber-matrix interface is the weakest link in fiber reinforced composite laminates. Hence, it is important to understand the interaction between fiber and matrix in fiber composites from a damage mechanism perspective. As a first step towards understanding the mechanics of short fiber composites, the problem of a rigid line inclusion embedded in an elastic matrix is usually studied. The rigid line inclusion is assumed to play the role of a short fiber. This assumption is valid since (a) the thickness of the fiber is negligible in comparison to other dimensions of the composite and (b) the elastic modulus of the fiber is much larger than that of the matrix material. Analysis of the stress field and fracture parameters of a rigid line inclusion in an elastic matrix could provide interesting insights on the fiber-matrix interaction in short fiber composites.

This chapter is organized as follows. In section 4.2, the mechanics of a rigid line inclusion embedded in an infinite matrix is reviewed. The duality principle along with Stroh formulation is used to obtain the singular stress field at the inclusion tip. We point out that characterizing the singularity in the elastic fields in terms of remote strain leads to strain intensity factors, which are independent of material properties of the matrix. The multi-parameter stress field equations are derived for rigid line inclusion problem in section 4.3. In section 4.4, we report on the numerical method (based on the reciprocal theorem) used to calculate the strain intensity factor for a rigid inclusion embedded in a finite matrix, via asymptotic and actual elastic fields near an inclusion tip. The actual stress field is obtained using a finite element analysis. In order to ascertain the accuracy of the finite element solution, photoelasticity experiments have been performed, and the obtained fringe patterns are qualitatively compared. The procedure for fabricating a rigid line inclusion embedded in an epoxy matrix is discussed in section 4.5. In section 4.6, the fringe contours obtained from photoelasticity experiments are compared with FEA results. For qualitative comparison we have adopted an existing algorithm to reconstruct the photoelastic fringe patterns. The strain intensity factors estimated, using the proposed numerical procedure, are then discussed. Further, the shear transfer length along the inclusion is also estimated. Finally, experimental strain intensity factor is also estimated and compared with numerical estimates. The conclusions of the paper are summarized in section 4.7.

4.2 Elastic field due to a rigid line inclusion embedded in an infinite elastic matrix

In this section we review the elasticity solution for the following problem using the Stroh formulation: a rigid line inclusion embedded in an infinite isotropic elastic matrix, which is subjected to a remote uniform strain. The Stroh formulation is adopted with the anticipation that the present framework can be extended to the case of an anisotropic matrix material with little effort. In the following we first outline the general theoretical framework used in the Stroh formulation. We then proceed to calculate the elastic fields for a crack in an infinite matrix that is subjected to a remote loading. Finally, using duality principles, introduced by Ni and Nasser [32], we estimate the stress, strain and displacement field around an embedded rigid line inclusion.

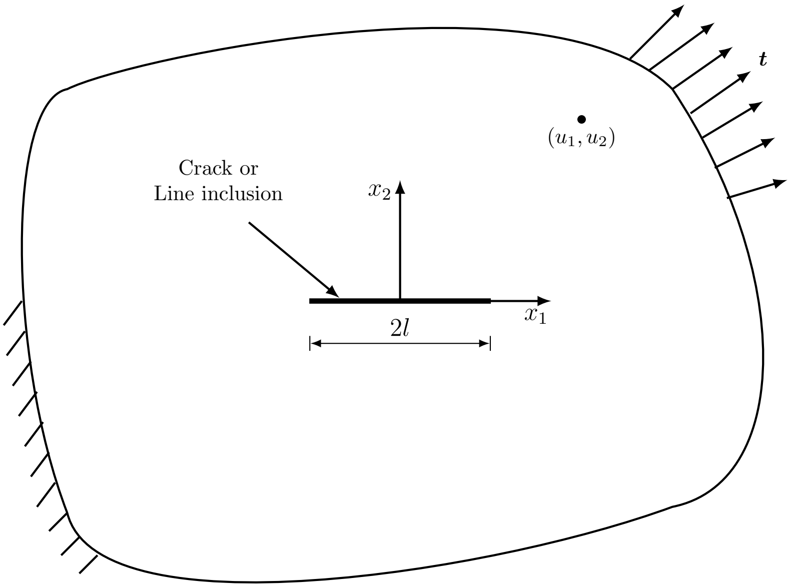


Figure 4.1: A schematic of the problem investigated. A rigid line inclusion embedded in an infinite elastic matrix subjected to a remote loading.

4.2.1 General theory

Let the displacement of a point in the (x_1, x_2) directions be denoted by (u_1, u_2), respectively (see Fig. 4.1). The strain components are given by

$$\varepsilon_{ij} = \frac{u_{i,j} + u_{j,i}}{2}, \tag{4.1}$$

where the subscript j denotes differentiation with respect to x_j. The constitutive behavior of the matrix material is assumed to be linear elastic, according to

$$\sigma_{ij} = C_{ijkl}\varepsilon_{kl}, \tag{4.2}$$

where C_{ijkl} represents the components of the elasticity tensor. In the absence of body and inertial forces, the force equilibrium is written as

$$\sigma_{ij,j} = 0. \tag{4.3}$$

On substituting Eqn. (4.1) and Eqn. (4.2) in Eqn. (4.3), we get

$$C_{ijkl} u_{k,lj} = 0. \tag{4.4}$$

Note that the Eqn. (4.4) represents force equilibrium in x_i direction. The three equations of (4.4) can be solved to obtain the three components of the displacements. Let us now assume the following general solution for the displacement field u_k

$$u_k = a_k f(z), \tag{4.5}$$

where the function $f(z)$ (with $z \equiv x_1 + p x_2$) is determined from the given boundary conditions. Here, a_k and p are constants to be determined by solving an eigenvalue problem, discussed next. By substituting Eqn. (4.5) in the equilibrium Eqn. (4.3), we obtain

$$\{Q_{ik} + p(R_{ik} + R_{ik}^T) + p^2 T_{ik}\} a_k = 0, \tag{4.6}$$

where $Q_{ik} \equiv C_{i1k1}$, $R_{ik} \equiv C_{i1k2}$, $R_{ik}^T \equiv C_{i2k1}$ and $T_{ik} \equiv C_{i2k2}$. We can now clearly see that p is an eigenvalue and the corresponding eigenvector is a_k. Nontrivial solutions for a_k exist only if the determinant of the matrix $(Q_{ik} + p(R_{ik} + R_{ik}^T) + p^2 T_{ik})$ vanishes, i.e.,

$$\left| Q_{ik} + p(R_{ik} + R_{ik}^T) + p^2 T_{ik} \right| = 0. \tag{4.7}$$

Since the coefficients of p in the above Eqn. (4.7) are real, the solution for p will be a set of three complex conjugate pairs [40]. Thus the above characteristic equation is of order six, having six roots for p. Likewise, we will have a set of three complex conjugate pairs of eigenvectors. Let us denote the three sets of eigenvalues as $(p_1, \bar{p}_1)$, $(p_2, \bar{p}_2)$ and $(p_3, \bar{p}_3)$. The corresponding eigenvectors are denoted as $(\boldsymbol{a}_1, \bar{\boldsymbol{a}}_1)$, $(\boldsymbol{a}_2, \bar{\boldsymbol{a}}_2)$, $(\boldsymbol{a}_3, \bar{\boldsymbol{a}}_3)$.

Substituting Eqn. (4.5) and Eqn. (4.1) in Eqn. (4.2), the stress components can be written as

$$\sigma_{ij} = C_{ijkl}\left(\delta_{l1} + p\delta_{l2}\right)a_k f'(z) = \left(C_{ijk1} + pC_{ijk2}\right)a_k f'(z). \tag{4.8}$$

To proceed further, we note the following identity from Eqn. (4.7)

$$\left(R^T_{ik} + pT_{ik}\right) = \frac{-1}{p}\left(Q_{ik} + pR_{ik}\right). \tag{4.9}$$

The stress expression shown in Eqn. (4.8) can be written for $j = 1$ as follows

$$\begin{aligned}\sigma_{i1} &= \left(C_{i1k1} + pC_{i1k2}\right)a_k f'(z) = \left(Q_{ik} + pR_{ik}\right)a_k f'(z),\\ &= -p\left(R^T_{ik} + pT_{ik}\right)a_k f'(z) = -pb_i f'(z) = -\frac{\partial}{\partial x_2}(b_i f(z)),\\ &= -\frac{\partial \phi_i}{\partial x_2},\end{aligned} \tag{4.10}$$

where ϕ_i is the stress function defined by

$$\phi_i \equiv b_i f(z),$$

with

$$b_i \equiv \left(R^T_{ik} + pT_{ik}\right)a_k.$$

Similarly for $j = 2$

$$\begin{aligned}\sigma_{i2} &= \left(C_{i2k1} + pC_{i2k2}\right)a_k f'(z) = \left(R^T_{ik} + pT_{ik}\right)a_k f'(z),\\ &= b_i f'z = \frac{\partial}{\partial x_1}(b_i f(z)),\\ &= \frac{\partial \phi_i}{\partial x_1}.\end{aligned} \tag{4.11}$$

Now, the general solution for the displacement field and stress function can be obtained using superposition as

$$\boldsymbol{u} = \sum_{I=1}^{3}\{\boldsymbol{a}_I f(z_I) + \bar{\boldsymbol{a}}_I f(\bar{z}_I)\}, \tag{4.12}$$

$$\boldsymbol{\phi} = \sum_{I=1}^{3}\{\boldsymbol{b}_I f(z_I) + \bar{\boldsymbol{b}}_I f(\bar{z}_I)\}, \tag{4.13}$$

where $z_k \equiv x_1 + p_k x_2$ and $\bar{z}_k \equiv x_1 + \bar{p}_k x_2$. Further, Eqns. 4.12 and 4.13 can be simplified by noting that the displacement and stress functions are real. We thus have

$$\begin{aligned} \boldsymbol{u} &= 2\Re\left(\sum_{I=1}^{3}\{\boldsymbol{a}_I f(z_I)\}\right) = 2\Re\{\boldsymbol{A}\boldsymbol{f}(z)\}, && (4.14)\\ \boldsymbol{\phi} &= 2\Re\left(\sum_{I=1}^{3}\{\boldsymbol{b}_I f(z_I)\}\right) = 2\Re\{\boldsymbol{B}\boldsymbol{f}(z)\}, && (4.15)\end{aligned}$$

where $\Re(\bullet)$ represents the real component of $(\bullet)$ and

$$\begin{aligned} \boldsymbol{A} &= [\boldsymbol{a}_1 \;\; \boldsymbol{a}_2 \;\; \boldsymbol{a}_3], \quad B = [\boldsymbol{b}_1 \;\; \boldsymbol{b}_2 \;\; \boldsymbol{b}_3], && (4.16)\\ \boldsymbol{f}(z) &= [f(z_1) \;\; f(z_2) \;\; f(z_3)]^T. && (4.17)\end{aligned}$$

To obtain the complete solution, $\boldsymbol{f}(z)$ has to be determined by evaluating the displacement/stress field at the boundaries. Equations 4.14 and 4.15 have been derived for a general anisotropic material that has distinct eigenvalues. However, in the case of isotropic materials, the three eigenvalues are identical and are equal to $i \equiv \sqrt{-1}$. In addition, the eigenvectors are $(1, i, 0)^T$, $(-i\kappa/2, -\kappa/2, 0)^T$ and $(0, 0, 1)^T$, where $\kappa = 3 - 4\nu$ for plane strain and $\kappa = (3-\nu)/(1+\nu)$ for plane stress, with ν being the Poisson's ratio of matrix material. Note that the first two eigenvectors are linearly dependent, and therefore solution has to be modified to accommodate the repeating eigenvector as [see Eqn. 13.1-1 in Ref. [41]]

$$\boldsymbol{u} = 2\Re\left\{\boldsymbol{A}\boldsymbol{F}(z)\boldsymbol{q}\right\} = 2\Re\left\{f(z)\boldsymbol{A}\boldsymbol{q} + x_2 f'(z)\boldsymbol{A}\boldsymbol{J}_{12}\boldsymbol{q}\right\}, \qquad (4.18)$$

$$\boldsymbol{\phi} = 2\Re\left\{\boldsymbol{B}\boldsymbol{F}(z)\boldsymbol{q}\right\}, = 2\Re\left\{f(z)\boldsymbol{B}\boldsymbol{q} + x_2 f'(z)\boldsymbol{B}\boldsymbol{J}_{12}\boldsymbol{q}\right\}, \qquad (4.19)$$

where

$$\boldsymbol{A} = \begin{bmatrix} 1 & -i\frac{\kappa}{2} & 0 \\ i & -\frac{\kappa}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{B} = -2\mu i \begin{bmatrix} 1 & -\frac{i}{2} & 0 \\ i & -\frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}, \quad \boldsymbol{F}(z) = \begin{bmatrix} f(z) & x_2 f'(z) & 0 \\ 0 & f(z) & 0 \\ 0 & 0 & f(z) \end{bmatrix},$$
$$\boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}, \quad \boldsymbol{J}_{12} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad (4.20)$$

Recall that $\boldsymbol{A}$ and $\boldsymbol{B}$ are determined from the material parameters, whereas $\boldsymbol{F}(z)$ and $\boldsymbol{q}$ have to be chosen appropriately so as to satisfy the boundary conditions.

4.2.2 Stress field due to a inclusion

We use the general framework discussed in previous section to solve for the stress field around a crack subjected to remote stress, and then apply duality principle to deduce the stress field around an rigid line inclusion tip subjected to remote strain field. Consider an infinite elastic plate with an elliptical hole with major axis a and minor axis b, parametrized by β. i.e.,

$$x_1 = a\cos(\beta), \qquad x_2 = b\sin(\beta), \qquad \text{with} \;\; 0 \leq \beta \leq 2\pi. \tag{4.21}$$

Since we are interested in a planar elastic solution, the applied remote stress field components are taken to be σ_{11}^{∞}, σ_{22}^{∞} and σ_{12}^{∞}. The resulting strain field components are ϵ_{11}^{∞}, ϵ_{22}^{∞} and ϵ_{12}^{∞}. To proceed further, the ellipse in the z plane is mapped to a unit circle in ζ plane using the following transformation [24]

$$\zeta = \frac{z + \sqrt{z^2 - a^2 + b^2}}{a+b}, \tag{4.22}$$

where $z = x_1 + ix_2$. It can be shown that, for the points belonging to the surface of hole, $\zeta = e^{i\beta}$ with $|\zeta| = 1$. This transformation helps us to apply the traction boundary conditions on the hole periphery.

In the absence of a discontinuity, the displacement and stress function vectors can be denoted as $\boldsymbol{u}^{\infty}$ and $\boldsymbol{\phi}^{\infty}$, which are given by

$$\boldsymbol{u}^{\infty} = x_1\boldsymbol{\epsilon}_1^{\infty} + x_2\boldsymbol{\epsilon}_2^{\infty}, \qquad \phi^{\infty} = x_1\boldsymbol{t}_2^{\infty} - x_2\boldsymbol{t}_1^{\infty}, \tag{4.23}$$

where

$$\boldsymbol{t}_1^{\infty} \equiv \left\{ \begin{array}{c} \sigma_{11}^{\infty} \\ \sigma_{12}^{\infty} \end{array} \right\}, \quad \boldsymbol{t}_2^{\infty} \equiv \left\{ \begin{array}{c} \sigma_{21}^{\infty} \\ \sigma_{22}^{\infty} \end{array} \right\}, \quad \boldsymbol{\epsilon}_1^{\infty} \equiv \left\{ \begin{array}{c} \epsilon_{11}^{\infty} \\ \epsilon_{12}^{\infty} \end{array} \right\}, \quad \boldsymbol{\epsilon}_2^{\infty} \equiv \left\{ \begin{array}{c} \epsilon_{21}^{\infty} \\ \epsilon_{22}^{\infty} \end{array} \right\}. \tag{4.24}$$

Let us now find the perturbation of the elasticity solution due to the traction boundary condition on the periphery of the hole. Since the stress state is uniform at infinity and the traction vanishes at the hole periphery, we can impose the following conditions on the stress function

$$\begin{aligned} \boldsymbol{\phi} &= 0 \qquad \text{for} \ \ |\zeta| = 1, &(4.25)\\ \boldsymbol{\phi} &= \boldsymbol{\phi}^{\infty} \qquad \text{for} \ \ |\zeta| \to \infty. &(4.26) \end{aligned}$$

In order to satisfy the far field traction boundary conditions, the following solution can be admitted

$$\boldsymbol{u} = \boldsymbol{u}^{\infty} + 2\Re\left\{\boldsymbol{A}\boldsymbol{F}(z)\boldsymbol{q}\right\}, \tag{4.27}$$

$$\boldsymbol{\phi} = \boldsymbol{\phi}^{\infty} + 2\Re\left\{\boldsymbol{B}\boldsymbol{F}(z)\boldsymbol{q}\right\}. \tag{4.28}$$

The far field boundary condition can be satisfied by choosing $f(z) = \zeta^{-1}$ in Eqn. (4.20.3). The vector of coefficients $\boldsymbol{q}$ is determined through the satisfaction of traction free boundary condition on the elliptical hole, see Eqn. (4.25); i.e., $\zeta = e^{i\beta}$. We thus have

$$\boldsymbol{q} = -\frac{1}{2}\boldsymbol{B}^{-1}(a\boldsymbol{t}_2^{\infty} - ib\boldsymbol{t}_1^{\infty}). \tag{4.29}$$

By substituting the solution for $\boldsymbol{q}$ in Eqn. (4.27) and by letting $b \to 0$ for a sharp crack we get the elastic fields due to remote traction. Then the solution reads as

$$\begin{aligned} \boldsymbol{u}(x_1, x_2) &= x_1\boldsymbol{\epsilon}_1^{\infty} + x_2\boldsymbol{\epsilon}_2^{\infty} - \Re\left\{\omega\boldsymbol{A}\boldsymbol{B}^{-1} + \omega'\boldsymbol{A}\boldsymbol{J}_{12}\boldsymbol{B}^{-1}\right\}\boldsymbol{t}_2^{\infty}, &(4.30)\\ \boldsymbol{\phi}(x_1, x_2) &= x_1\boldsymbol{t}_2^{\infty} - x_2\boldsymbol{t}_1^{\infty} - \Re\left\{\omega\boldsymbol{I} + \omega'\boldsymbol{B}\boldsymbol{J}_{12}\boldsymbol{B}^{-1}\right\}\boldsymbol{t}_2^{\infty}, &(4.31) \end{aligned}$$

where $\omega = \left(\sqrt{z^2 - l^2} - z\right)$, $\omega' = x_2\left(z/\sqrt{z^2 - l^2} - 1\right)$ and a is replaced by l for half length of the crack and $z = x_1 + ix_2$.

In order to obtain the elastic solution for the inclusion, we use the general duality principle introduced by Ni and Nasser [32]. The solution for the rigid line inclusion is then obtained as [42]

$$\begin{aligned} \boldsymbol{u}(x_1, x_2) &= x_1\boldsymbol{\epsilon}_1^{\infty} + x_2\boldsymbol{\epsilon}_2^{\infty} - \Re\left\{\omega\boldsymbol{I} + \omega'\boldsymbol{A}\boldsymbol{J}_{12}\boldsymbol{A}^{-1}\right\}\boldsymbol{d}^{\infty}, &(4.32)\\ \boldsymbol{\phi}(x_1, x_2) &= x_1\boldsymbol{t}_2^{\infty} - x_2\boldsymbol{t}_1^{\infty} - \Re\left\{\omega\boldsymbol{B}\boldsymbol{A}^{-1} + \omega'\boldsymbol{B}\boldsymbol{J}_{12}\boldsymbol{A}^{-1}\right\}\boldsymbol{d}^{\infty}, &(4.33) \end{aligned}$$

where $\boldsymbol{d}^{\infty} = (-\epsilon_{11}^{\infty}, 0)^T$. The full field solution comprises of two parts. The first two terms contain the homogeneous solution (in absence of the inclusion) and the third term represents the inhomogeneous solution (due to the presence of inclusion). Note that the inhomogeneous part of the solution is only due to the far-field strain component ϵ_{11}, which acts parallel to the inclusion. This is in contrast to the crack problem, where the inhomogeneous part of the solution has contributions from both remote normal σ_{22}^{∞} and shear stress σ_{12}^{∞}. Now using Eqns. (4.1), (4.10), (4.11), (4.32) and (4.33), we can obtain the full field stress and strain expressions due to the remotely applied strain ϵ_{11}^{∞} in case of a rigid line inclusion problem. These results, after simplification, are summarized below

$$\begin{aligned}
\sigma_{11} &= \frac{\mu\epsilon_{11}^{\infty}}{\kappa}\Re\left\{\Gamma(3+\kappa)+2\Gamma' i\right\}+\frac{\mu(\kappa+1)}{\kappa-1}\epsilon_{11}^{\infty}, && (4.34)\\
\sigma_{22} &= \frac{\mu\epsilon_{11}^{\infty}}{\kappa}\Re\left\{-\Gamma(1-\kappa)+2\Gamma' i\right\}+\frac{\mu(3-\kappa)}{\kappa-1}\epsilon_{11}^{\infty}, && (4.35)\\
\sigma_{12} &= \frac{\mu\epsilon_{11}^{\infty}}{\kappa}\Re\left\{-\Gamma i(1+\kappa)-2\Gamma'\right\}, && (4.36)
\end{aligned}$$

$$\begin{aligned}
\epsilon_{11} &= \frac{\epsilon_{11}^{\infty}}{\kappa}\Re\left\{\Gamma\kappa+\Gamma' i\right\}+\epsilon_{11}^{\infty}, && (4.37)\\
\epsilon_{22} &= -\frac{\epsilon_{11}^{\infty}}{\kappa}\Re\left\{\Gamma+\Gamma' i\right\}, && (4.38)\\
\epsilon_{12} &= \frac{\epsilon_{11}^{\infty}}{\kappa}\Re\left\{\Gamma i\frac{1+\kappa}{2}-\Gamma'\right\}, && (4.39)
\end{aligned}$$

where $\Gamma = \left(z/\sqrt{z^2-l^2}-1\right)$ and $\Gamma' = x_2\left(1/\sqrt{z^2-l^2}-z^2/\left(z^2-l^2\right)^{3/2}\right)$ with $z = x_1+ix_2$.

We now deduce the order of singularity and the corresponding strain intensity factor from the above full-field elastic solution. For this purpose we transform the Eqns. (4.34-4.39) into polar coordinate system (r,θ) with an origin located at the right tip of the inclusion (See Fig. 4.2). We then replace the variable z with $\psi+l$, where $\psi = re^{i\theta}$ and let $r/l \to 0$ to obtain the asymptotic inclusion tip stress and strain field solution as shown below

$$\begin{aligned}
\sigma_{11} &= \frac{2\mu\epsilon_{11}^{\infty}}{\kappa}\left(\frac{l}{2r}\right)^{1/2}\cos\left(\frac{\theta}{2}\right)\left[1+\frac{1+\kappa}{2}-\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{3\theta}{2}\right)\right], && (4.40)\\
\sigma_{22} &= \frac{2\mu\epsilon_{11}^{\infty}}{\kappa}\left(\frac{l}{2r}\right)^{1/2}\cos\left(\frac{\theta}{2}\right)\left[1-\frac{1+\kappa}{2}+\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{3\theta}{2}\right)\right], && (4.41)\\
\sigma_{12} &= \frac{2\mu\epsilon_{11}^{\infty}}{\kappa}\left(\frac{l}{2r}\right)^{1/2}\sin\left(\frac{\theta}{2}\right)\left[\frac{1+\kappa}{2}+\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{3\theta}{2}\right)\right], && (4.42)
\end{aligned}$$

$$\epsilon_{11} = \frac{\epsilon_{11}^{\infty}}{\kappa}\left(\frac{l}{2r}\right)^{1/2}\cos\left(\frac{\theta}{2}\right)\left[\kappa - \sin\left(\frac{\theta}{2}\right)\sin\left(\frac{3\theta}{2}\right)\right], \tag{4.43}$$

$$\epsilon_{22} = -\frac{\epsilon_{11}^{\infty}}{\kappa}\left(\frac{l}{2r}\right)^{1/2}\cos\left(\frac{\theta}{2}\right)\left[1 - \sin\left(\frac{\theta}{2}\right)\sin\left(\frac{3\theta}{2}\right)\right], \tag{4.44}$$

$$\epsilon_{12} = \frac{\epsilon_{11}^{\infty}}{\kappa}\left(\frac{l}{2r}\right)^{1/2}\sin\left(\frac{\theta}{2}\right)\left[\frac{1+\kappa}{2} + \cos\left(\frac{\theta}{2}\right)\cos\left(\frac{3\theta}{2}\right)\right]. \tag{4.45}$$

Similar results have also been reported by Noselli et al. [33] and Wang et al. [26]. On inspecting the above solution for the strain field, it is natural for us to define a strain intensity factor (rather than a stress intensity factor) as

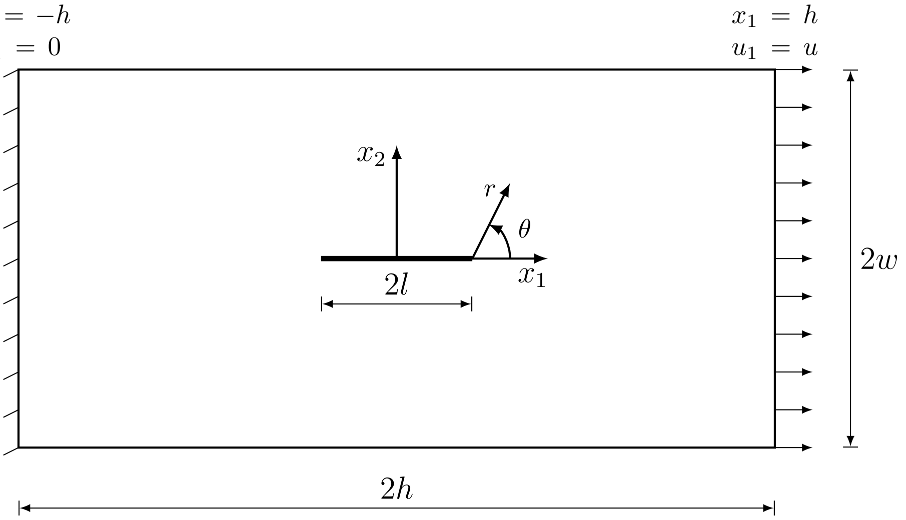


Figure 4.2: The rigid line inclusion in a matrix subjected to uniform displacement.

$$K_I^{\epsilon} = \lim_{r\to 0}\epsilon_{11}(\theta = 0^{\circ})\sqrt{r}. \tag{4.46}$$

In the present problem, i.e., for infinite geometry we have $K_I^{\epsilon} = \epsilon_{11}^{\infty}\sqrt{l/2}$. The following salient features can be noted from the above mentioned solution. (1) The order of stress singularity for the inclusion tip is the same as that of a crack tip. (2) The asymptotic stress field is always symmetric and depends only on the applied normal strain in the direction of the inclusion. This is in contrast to the asymptotic field near a crack tip, where the stress field can also be antisymmetric due to a mode II loading. Consequently, for planar loading case only a single strain intensity factor definition is applicable for the inclusion problem. (3) The strain intensity factor is independent of the material properties of the matrix material.

Attempts to characterize the magnitude of singularity have been made in the past [26, 33]. In these articles the magnitude of singularity was defined based on the stress components.

Consequently, they listed mode I and II stress intensity factors for the line inclusion problem. They also noted that the stress intensity factors also depend on the elastic properties of the matrix material. Recall that from a fracture mechanics perspective, the stress intensity factor is usually interpreted as a loading parameter. Hence it should not be a function of the material properties. This feature is lost for the stress intensity factor in the present problem. Therefore, it is appropriate to characterize the magnitude of singularity in terms of a strain intensity factor as discussed above. We re-emphasize that the strain intensity factor for a rigid line inclusion is independent of the properties of the matrix material. For completeness, we elucidate the consequences of quantifying the singularity in-terms of a stress intensity factor in the appendix A.

4.3 Multi-parameter equations for rigid line inclusion embedded in an elastic matrix

In this section, multi-parameter stress field equations are derived based on the elastic asymptotic stress field around tip of inclusion for rigid line inclusion problem. Consider the inclusion as shown in Fig. 4.1. Let (r, θ) be a cylindrical polar system and origin at tip of inclusion. Now considering the Airy's stress function assumed to be of the form

$$\Phi = r^{\lambda+1}\left[A\cos((\lambda+1)\theta) + B\sin((\lambda-1)\theta) + C\cos((\lambda+1)\theta + D\sin((\lambda-1)\theta))\right] \tag{4.47}$$

where A, B, C and D are constants and they are deduced based on the boundary condition of defined problem. The boundary conditions for rigid line inclusion problem are as given below

$$u_r = 0; \qquad u_\theta = 0 \qquad \text{on} \quad \theta = \pm\pi \tag{4.48}$$

The stress function is determined by imposing the above boundary conditions.The equations are solved by using the Michell solution, where we get solution for symmetric part as

$$\begin{aligned} \lambda &= \pm n, \qquad\qquad n \text{ is an integer}, &(4.49)\\ C &= \frac{\kappa-\lambda}{1+\lambda}A, &(4.50)\end{aligned}$$

and

$$
\begin{aligned}
\lambda &= \pm\frac{2n+1}{2}, \qquad\qquad n \text{ is an integer,} && (4.51)\\
C &= -\frac{\kappa+\lambda}{1+\lambda}A. && (4.52)
\end{aligned}
$$

Similarly for anti-symmetric part

$$
\begin{aligned}
\lambda &= \pm n, \qquad\qquad n \text{ is an integer,} && (4.53)\\
D &= -\frac{\kappa+\lambda}{1+\lambda}B, && (4.54)
\end{aligned}
$$

and

$$
\begin{aligned}
\lambda &= \pm\frac{2n+1}{2}, \qquad\qquad n \text{ is an integer,} && (4.55)\\
D &= \frac{\kappa-\lambda}{1+\lambda}B. && (4.56)
\end{aligned}
$$

The displacement to be finite at the inclusion tip so that the feasible values of λ are only positive numbers including zero. The stress field thus obtained can be written as

$$
\begin{aligned}
\sigma_{rr} &= -2\mu\lambda r^{\lambda-1}\left[A(\lambda-3)\cos((\lambda-1)\theta)+C(\lambda+1)\cos((\lambda+1)\theta)\right]\\
&\quad -2\mu\lambda r^{\lambda-1}\left[B(\lambda-3)\sin((\lambda-1)\theta)+D(\lambda+1)\sin((\lambda+1)\theta)\right], && (4.57)\\
\sigma_{\theta\theta} &= 2\mu\lambda r^{\lambda-1}\left[A(\lambda+1)\cos((\lambda-1)\theta)+C(\lambda+1)\cos((\lambda+1)\theta)\right]\\
&\quad +2\mu\lambda r^{\lambda-1}\left[B(\lambda+1)\sin((\lambda-1)\theta)+D(\lambda+1)\sin((\lambda+1)\theta)\right], && (4.58)\\
\sigma_{r\theta} &= 2\mu\lambda r^{\lambda-1}\left[A(\lambda-1)\sin((\lambda-1)\theta)+C(\lambda+1)\sin((\lambda+1)\theta)\right]\\
&\quad -2\mu\lambda r^{\lambda-1}\left[B(\lambda-1)\cos((\lambda-1)\theta)+D(\lambda)\cos((\lambda+1)\theta)\right]. && (4.59)
\end{aligned}
$$

In Eqns. (4.57 -4.59), replacing $\lambda = n/2$, where n is number of terms for elastic asymptotic stress field function expansion. The Eqns. (4.57 -4.59) can be rewritten elegantly considering Eqns. (4.49-4.56) as

$$\begin{aligned}
\sigma_{rr} &= -2\mu\frac{A_{In}n}{2}r^{\frac{n}{2}-1}\left[\left(\frac{n}{2}-3\right)\cos\left(\frac{n}{2}-1\right)\theta+\left((-1)^{n}\kappa-\frac{n}{2}\right)\cos\left(\frac{n}{2}+1\right)\theta\right]\\
&- 2\mu\frac{A_{IIn}n}{2}r^{\frac{n}{2}-1}\left[\left(\frac{n}{2}-3\right)\sin\left(\frac{n}{2}-1\right)\theta-\left((-1)^{n}\kappa+\frac{n}{2}\right)\sin\left(\frac{n}{2}+1\right)\theta\right], \quad (4.60)\\
\sigma_{\theta\theta} &= 2\mu\frac{A_{In}n}{2}r^{\frac{n}{2}-1}\left[\left(\frac{n}{2}+1\right)\cos\left(\frac{n}{2}-1\right)\theta+\left((-1)^{n}\kappa-\frac{n}{2}\right)\cos\left(\frac{n}{2}+1\right)\theta\right]\\
&+ 2\mu\frac{A_{IIn}n}{2}r^{\frac{n}{2}-1}\left[\left(\frac{n}{2}+1\right)\sin\left(\frac{n}{2}-1\right)\theta-\left((-1)^{n}\kappa+\frac{n}{2}\right)\sin\left(\frac{n}{2}+1\right)\theta\right], \quad (4.61)\\
\sigma_{r\theta} &= 2\mu\frac{A_{In}n}{2}r^{\frac{n}{2}-1}\left[\left(\frac{n}{2}-1\right)\sin\left(\frac{n}{2}-1\right)\theta+\left((-1)^{n}\kappa-\frac{n}{2}\right)\sin\left(\frac{n}{2}+1\right)\theta\right]\\
&- 2\mu\frac{A_{IIn}n}{2}r^{\frac{n}{2}-1}\left[\left(\frac{n}{2}-1\right)\cos\left(\frac{n}{2}-1\right)\theta-\left((-1)^{n}\kappa+\frac{n}{2}\right)\cos\left(\frac{n}{2}+1\right)\theta\right], \quad (4.62)
\end{aligned}$$

where A_{In} and A_{IIn} are constants with number of parameters n. These equations are transformed to cartesian coordinate from polar coordinates as

$$\begin{aligned}
\sigma_{xx} &= 2\mu\frac{A_{In}n}{2}r^{\frac{n}{2}-1}\left[\left(2-(-1)^{n}\kappa+\frac{n}{2}\right)\cos\left(\frac{n}{2}-1\right)\theta-\left(\frac{n}{2}-1\right)\cos\left(\frac{n}{2}-3\right)\theta\right]\\
&+ 2\mu\frac{A_{IIn}n}{2}r^{\frac{n}{2}-1}\left[\left(2+(-1)^{n}\kappa+\frac{n}{2}\right)\sin\left(\frac{n}{2}-1\right)\theta-\left(\frac{n}{2}-1\right)\sin\left(\frac{n}{2}-3\right)\theta\right], \quad (4.63)\\
\sigma_{yy} &= 2\mu\frac{A_{In}n}{2}r^{\frac{n}{2}-1}\left[\left(2+(-1)^{n}\kappa-\frac{n}{2}\right)\cos\left(\frac{n}{2}-1\right)\theta+\left(\frac{n}{2}-1\right)\cos\left(\frac{n}{2}-3\right)\theta\right]\\
&+ 2\mu\frac{A_{IIn}n}{2}r^{\frac{n}{2}-1}\left[\left(2-(-1)^{n}\kappa-\frac{n}{2}\right)\sin\left(\frac{n}{2}-1\right)\theta+\left(\frac{n}{2}-1\right)\sin\left(\frac{n}{2}-3\right)\theta\right], \quad (4.64)\\
\sigma_{xy} &= 2\mu\frac{A_{In}n}{2}r^{\frac{n}{2}-1}\left[\left((-1)^{n}\kappa-\frac{n}{2}\right)\sin\left(\frac{n}{2}-1\right)\theta+\left(\frac{n}{2}-1\right)\sin\left(\frac{n}{2}+1\right)\theta\right]\\
&- 2\mu\frac{A_{IIn}n}{2}r^{\frac{n}{2}-1}\left[\left((-1)^{n}\kappa+\frac{n}{2}\right)\cos\left(\frac{n}{2}-1\right)\theta-\left(\frac{n}{2}-1\right)\cos\left(\frac{n}{2}+1\right)\theta\right], \quad (4.65)
\end{aligned}$$

where $s = (n/2) - 1$, $t = (n/2) - 3$, σ_{xx} and σ_{yy} are the normal stress in x and y directions respectively, τ_{xy} is in plane shear stress, r and θ are the polar coordinates with origin at the inclusion tip and n is the number of parameters. The coefficient A_{I1} of multi-parameter stress field equations is related to mode-I strain intensity factor K_I^{ϵ} as $A_{I1} = K_I^{\epsilon}/\kappa$ and another coefficient A_{II1} represent mode-II fracture parameter. The above multi-parameter stress field equations are verified by considering equivalence between rigid line inclusion and crack tip problem. From Ref. [43], replacing $\kappa = -1$ in Eqns. (4.63- 4.65) leads to the multi-parameter stress field equations for a crack problem. These multi-parameter equations further are used for estimating the strain intensity factor experimentally.

4.4 Numerical method and estimation of strain intensity factor for finite domain

The analysis in the section (4.2) is valid for an inclusion embedded in an infinite elastic medium. However, for finite geometries the analytical framework may not be straight forward. In order to deduce the strain singularities for arbitrary geometry, we have developed a numerical framework, based on the reciprocal theorem, following the procedure described in Refs. [44–46]. To proceed further we need the general solution for asymptotic stress field near the inclusion tip. A standard exercise in elasticity [47] results in the following asymptotic field as shown below

$$\begin{aligned}
\sigma_{rr} &= -2\mu H\lambda r^{\lambda-1}\left[(\lambda-3)\cos((\lambda-1)\theta)-(\kappa+\lambda)\cos((\lambda+1)\theta)\right], && (4.66)\\
\sigma_{\theta\theta} &= 2\mu H\lambda r^{\lambda-1}\left[(\lambda+1)\cos((\lambda-1)\theta)-(\kappa+\lambda)\cos((\lambda+1)\theta)\right], && (4.67)\\
\sigma_{r\theta} &= 2\mu H\lambda r^{\lambda-1}\left[(\lambda-1)\sin((\lambda-1)\theta)-(\kappa+\lambda)\sin((\lambda+1)\theta)\right], && (4.68)
\end{aligned}$$

$$\begin{aligned}
\epsilon_{rr} &= H\lambda r^{\lambda-1}\left[(\kappa-\lambda)\cos((\lambda-1)\theta)+(\kappa+\lambda)\cos((\lambda+1)\theta)\right], && (4.69)\\
\epsilon_{\theta\theta} &= H\lambda r^{\lambda-1}\left[(\kappa+\lambda-2)\cos((\lambda-1)\theta)-(\kappa+\lambda)\cos((\lambda+1)\theta)\right], && (4.70)\\
\epsilon_{r\theta} &= H\lambda r^{\lambda-1}\left[(\lambda-1)\sin((\lambda-1)\theta)-(\kappa+\lambda)\sin((\lambda+1)\theta)\right], && (4.71)
\end{aligned}$$

$$\begin{aligned}
u_r &= Hr^{\lambda}\left[(\kappa-\lambda)\cos((\lambda-1)\theta)+(\kappa+\lambda)\cos((\lambda+1)\theta)\right], && (4.72)\\
u_\theta &= Hr^{\lambda}\left[(\kappa+\lambda)\sin((\lambda-1)\theta)-(\kappa+\lambda)\sin((\lambda+1)\theta)\right], && (4.73)
\end{aligned}$$

where H is a coefficient depending on the boundary conditions,

$$\lambda = \pm\frac{n}{2}, \qquad n \text{ is an integer,} \tag{4.74}$$

and (r,θ) are defined in Fig. 4.2. The singular fields near the inclusion tip can be written as

$$\begin{aligned}
u_i &= Hr^{\lambda}g_i\left(\lambda,\theta\right), && (4.75)\\
\sigma_{ij} &= 2\mu Hr^{\lambda-1}f_{ij}\left(\lambda,\theta\right). && (4.76)
\end{aligned}$$

where $[i,j] \in [r,\theta]$, f_{ij} and g_i are known functions of λ and θ deduced from Eqns. (4.66-4.68) and (4.72-4.73) respectively. By comparing the equations (4.43) and (4.69) we can see that H is the strain intensity factor. The procedure to calculate H is adopted from Akisanya and Fleck [44] and Carpenter and Byers [45] as described below. The reciprocal theorem can be stated as

$$\oint_C \left(\sigma_{ij} u_i^* - \sigma_{ij}^* u_i\right) n_j dS = 0. \tag{4.77}$$

where n_j is the unit normal to the contour C, whose outer radius is r_2 and the inner radius is r_1 (See Fig. 4.3). In the above equation (σ_{ij}, u_j) are the actual stress and displacement fields and (σ_{ij}^*, u_j^*) are suitably chosen auxiliary fields that satisfy the boundary conditions. The auxiliary fields are chosen to be the asymptotic fields with $\lambda^* = -\lambda$ (the rationale for this choice will be apparent later)

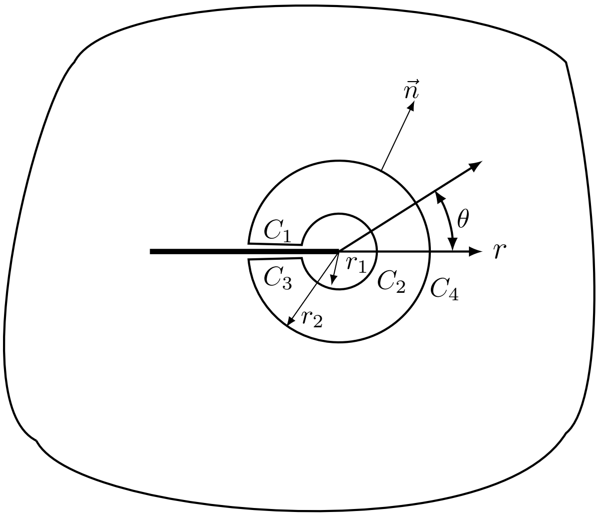


Figure 4.3: Contours around the inclusion tip for evaluating contour integral.

$$\begin{aligned} u_j^* &= H^* r^{\lambda^*} g_j\left(\lambda^*, \theta\right), & (4.78)\\ \sigma_{ij}^* &= 2\mu H^* r^{\lambda^*-1} f_{ij}\left(\lambda^*, \theta\right). & (4.79)\end{aligned}$$

One can subdivide the contour C into four parts as shown in Fig. 4.3 . The integral in Eqn. (4.77) vanishes over C_1 and C_3 because of the displacement boundary conditions. The

Eqn. (4.77) can be then rewritten as

$$\int_{C_2} \left(\sigma_{ij} u_i^* - \sigma_{ij}^* u_i\right) n_j dS = -\int_{C_4} \left(\sigma_{ij} u_i^* - \sigma_{ij}^* u_i\right) n_j dS. \tag{4.80}$$

For the line integral along the inner contour C_2, (σ_{ij}, u_j) are taken to be the asymptotic elastic fields (in which H is unknown) and for the line integral along the outer contour C_4, (σ_{ij}, u_j) are taken from the actual elasticity solution. First, we consider the integral along the contour C_4. Instead of performing a contour integral we perform an area integral. To wards this we define a scalar $m \equiv (r_2 - r)/(r_2 - r_1)$ which is unity on C_4 and vanishes on C_2. Now we can write,

$$\begin{aligned}
-\int_{C_4} \left(\sigma_{ij} u_i^* - \sigma_{ij}^* u_i\right) n_j dS, &= -\int_{C} m \left(\sigma_{ij} u_i^* - \sigma_{ij}^* u_i\right) n_j dS, && (4.81)\\
&= -\int_{A} \left(\sigma_{ij} u_i^* - \sigma_{ij}^* u_i\right) \frac{\partial m}{\partial x_j} dA && (4.82)\\
&= H^* \int_{A} \left(\sigma_{ij} r^{\lambda^*} g_i\left(\lambda^*, \theta\right) - u_i r^{\lambda^* - 1} 2\mu f_{ij}\left(\lambda^*, \theta\right)\right) \frac{\partial m}{\partial x_j} dA && (4.83)
\end{aligned}$$

where the last equality is obtained using the divergence theorem. We now consider the line integral along C_2. Substituting asymptotic elastic field and auxiliary field on the left hand side of equation Eqn. (4.80) the line integral becomes

$$\begin{aligned}
\int_{C_2} \left(\sigma_{ij} u_i^* - \sigma_{ij}^* u_i\right) n_j dS &= 2\mu H H^* \int_{-\pi}^{\pi} \left(f_{ij}(\lambda) g_i(\lambda^*) - g_i(\lambda) f_{ij}(\lambda^*)\right) n_j d\theta \\
&= c_1 H H^*, && (4.84)
\end{aligned}$$

where

$$c_1 = 2\mu \int_{-\pi}^{\pi} \left(f_{ij}(\lambda) g_i(\lambda^*) - g_i(\lambda) f_{ij}(\lambda^*)\right) n_j d\theta. \tag{4.85}$$

The value of c_1 can be calculated by performing the numerical integration. Note that c_1 is independent of r; this is due to the choice $\lambda^* = -\lambda$. Finally, Eqn. (4.80) can be rewritten using Eqns. (4.83) and (4.84) as given below

$$H = -\frac{1}{c_1} \int_{A} \left(\sigma_{ij} r^{\lambda^*} g_i\left(\lambda^*, \theta\right) - u_i r^{\lambda^* - 1} 2\mu f_{ij}\left(\lambda^*, \theta\right)\right) \frac{\partial m}{\partial x_j} dA. \tag{4.86}$$

In the above equation, the actual stress fields (σ_{ij}, u_i) are computed using finite elements, whereas the auxiliary fields are given by Eqn. (4.79). The FEA calculations are carried out using the commercial finite element package ABAQUS-6.9 [48]. The stress and displacement field from FEA is given as input to a script written in MATLAB [49] to estimate the strain intensity factor via Eqn. 4.86. The details of the finite element model is explained in section 4.6. Validation of the finite element analysis model is carried out by comparing the fringe pattern, surrounding the inclusion tip, obtained from FEA with that of photoelasticity experiments. The specimen fabrication and photoelastic experiments are discussed next.

4.5 Specimen Fabrication and Photoelastic Experimentation

The specimens are fabricated in a closed mould and it is made up of detachable perspex sheet. It has a cavity with 230 mm length, 120 mm width and 6 mm thickness. On the bottom plate of mould, a steel sheet of dimensions 20 mm x 6 mm, is placed orthogonal with the help of adhesive drops bonded at center (See Fig. 4.4). To realize the rigid line inclusion, we have used a 0.1 mm thick steel sheet. To enhance the bonding, steel sheet is polished using a fine (P 400) sandpaper. A matrix material is made of two-part epoxy resin Epofine-221 and hardener Finehard-1842, supplied by Fine Finish Organics Pvt. Ltd., India. The matrix is obtained by mixing the resin and hardener in proportion of 100:40 by weight and then poured into a mould. This mixture is allowed to cure for 48 hour at an ambient temperature of 28° C. From the molded sheet, specimens having a dimension of 200 mm x 100 mm x 6 mm containing a single inclusion of 20 mm length placed at the center is realized (See Fig. 4.5a). Similarly, specimen having an inclusion at 45° is also fabricated using the same technique as shown in Fig. (4.5b). Conventional photoelasticity experiment has been performed and the

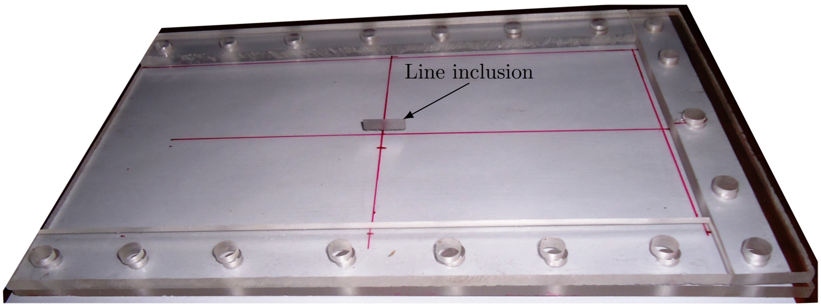


Figure 4.4: Mould for casting specimen along with line inclusion made of steel kept at center.

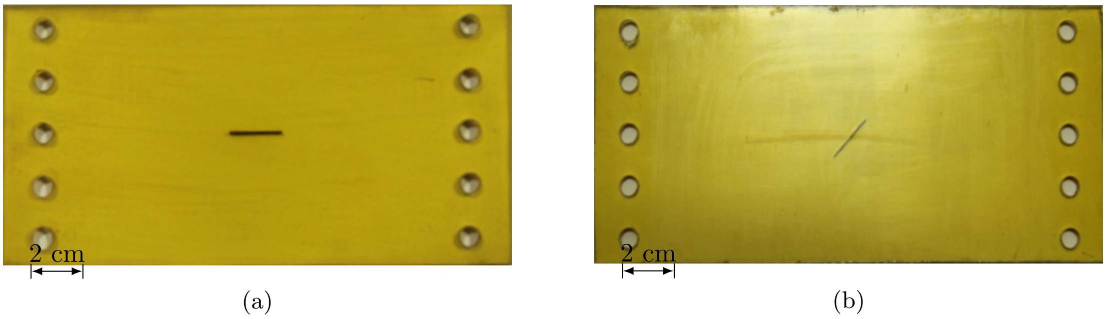


Figure 4.5: Rigid line inclusion with epoxy matrix sample (a) straight inclusion (b)inclined inclusion at 45°.

obtained dark field fringe pattern is captured with a CCD (charged couple device) camera. The polariscope arrangement consists of a light source, a polarizer, a first quarter-wave plate, the specimen, a second quarter-wave plate, and an analyzer. To get dark field fringe pattern, the analyzer is crossed with respect to the polarizer (90°) phase angle). The first quarter wave plate is oriented at 45° with respect to the polarizer and both quarter- wave plates are in crossed position. The optical elements are arranged so as to allow the light to propagate normal to the plane of the specimen. The loaded specimen is placed in between quarter-wave plates. Monochromatic light is used for obtaining gray scale images. Figure (4.6) shows the schematic of photoelasticity setup being used for obtaining the dark field fringe pattern.

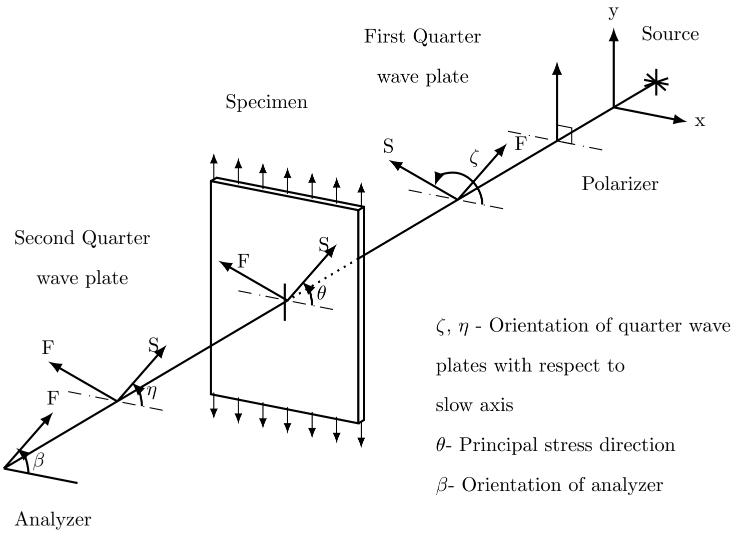


Figure 4.6: Schematic of photoelastic experimental arrangement.

4.6 Results

4.6.1 Comparison of analytical solution with experimental fringe pattern

The displacement and stress field for a rigid line inclusion embedded in an epoxy matrix can be obtained from the Eqn. (4.34). From the photoelastic experiments, one could obtain in-plane principal stress difference in the form of fringe pattern. Hence, for qualitative comparison, results from analytical solution and FEA are converted to photoelastic fringe pattern using a simple algorithm as explained in Appendix B. At every point in the specimen, the in-plane principal stress difference value is related to a fringe order as given below [10]

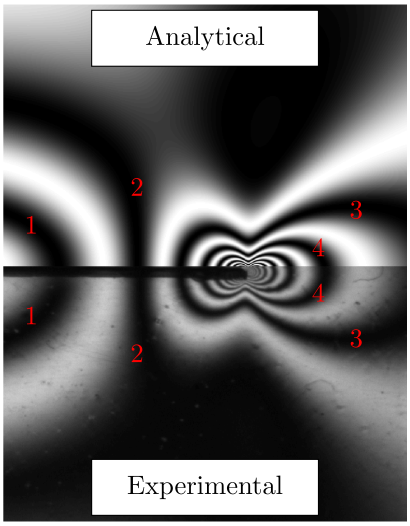


Figure 4.7: Top half represents fringe contour plotted from analytical solution and bottom half represents experimental dark field photoelastic fringe contour obtained for a tensile load of 60 N.

$$\sigma_1 - \sigma_2 = \frac{N f_\sigma}{h}, \tag{4.87}$$

where $\sigma_1 - \sigma_2$ is principal stress difference, h is thickness of specimen, f_σ is material stress fringe value and N is the fringe order. The material stress fringe value for epoxy matrix is 0.25 N/mm/fringe and it is obtained by a standard calibration procedure. The fringe order data is converted to the dark field intensity value by equation $I = I_a(1 - \cos(2\pi N))$, where I_a is amplitude of incident light and the obtained intensity information is used for reconstructing fringe contours pixel wise [10]. The full field analytical and experimental fringe contour obtained for a line inclusion specimen is shown in Fig. (4.7). Here, a tensile load of 60 N is applied parallel to the line inclusion. The fringe order is minimum at the center of inclusion and gradually increases towards the inclusion tip. At the tip of the inclusion singular stress field observed is same as that of a crack tip field. A very good coherence exists between the analytical and experimental fringe pattern thereby confirming the accuracy of the derived full field stress solution.

4.6.2 Comparison of FEA solution with experiment

We now validate the finite element model used to obtain the actual elasticity solution that is input to the strain intensity factor calculation. The validation is important because we model the rigid line inclusion using the constraint that the nodes lying on the inclusion line deform in a rigid manner.

A 2D plate is modeled with dimensions with following dimensions: $2l = 20$ mm, $2h = 200$ mm and $2w = 100$ mm. The plate is discretized using 8-nodded plane strain elements. The quarter point element is used to capture square root singularity at the tip of inclusion. A mesh convergence study has been performed to arrive at the number of circumferential element at the tip of inclusion: it is found to be 80 elements in the circumferential and 60 along the radial direction. Figure (4.8) shows a magnified view of the mesh surrounding the inclusion tip. Away from the inclusion, a coarse mesh is being used in order to reduce total number of degrees of freedom. In total, there are 17920 elements corresponding to 161877 degrees of freedom. The rigid inclusion is modeled as a rigid line defined by constraint equations in ABAQUS.

The degrees of freedom of the nodes on the edge at $x = -h$ are arrested. A tensile load of 60 N magnitude is applied on the edge at $x = +h$ and these results are compared with photoelastic fringes. For strain intensity factor calculation, an unit displacement (u) is applied on edge on the edge at $x = +h$ and applied strain value ϵ_{11}^{∞} is calculated as $u/(2h)$.

The dark field inclusion tip fringe contours are reconstructed from the nodal stress values obtained using FEA [50]. A brief description of the fringe plotting algorithm is given in Appendix B. The fringe plots obtained from FEA and photoelastic experiment are compared in Fig. (4.9). There is a very good coherence between the experimental and numerical results and thereby validating the FEA model used to perform simulations. Similarly results obtained for an inclined rigid line inclusion problem is shown in Fig. (4.10). It is taken at a load of 45 N. The inclined inclusion specimen has dimensions of 200 mm x 100 mm with 6 mm

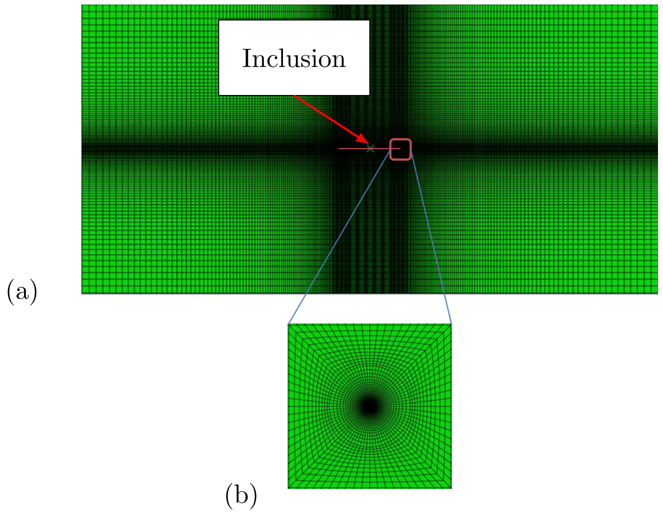


Figure 4.8: Finite element model (a) Complete specimen (b) Zoomed view of the line inclusion tip.

thickness and steel inclusion of 20 mm length with 45° orientation with respect to loading direction. The analytical stress fields are estimated for an inclined rigid line inclusion using Eqns. (4.34-4.36), and they are transformed to a coordinate system with axis aligned parallel to the loading direction. The reconstructed fringe plot from both analytical and FEA along with the experimental photoelastic fringe contours are shown for a qualitative comparison. Once again there is a good match thereby validating the FEA model.

4.6.3 Strain Intensity Factor for rigid line inclusion using numerical method

We now report on the calculation of the strain intensity factors for a rigid line inclusion aligned with the x_1 axis and subjected to a remote strain ϵ_{11}^{∞} for various combinations of l/w and l/h ratios (see Fig. 4.2). These strain intensity factors are determined using the methodology explained in section (4.4). The strain intensity factor for a finite geometry can be written as

$$K_I^{\epsilon} = \epsilon_{11}\sqrt{\frac{l}{2}}\, F\left(\frac{l}{w}, \frac{l}{h}\right) \tag{4.88}$$

where F is the geometrical correction factor. Recall that the strain intensity factor for an infinite geometry is $K_I^{\epsilon} = \epsilon_{11}^{\infty}\sqrt{l/2}$, implying that the correction factor is unity for infinite matrix. The values of F are estimated for different values of l/h and l/w ratio as shown in Fig. (4.11). The following conclusions can be drawn. (1) The value of the correction factor

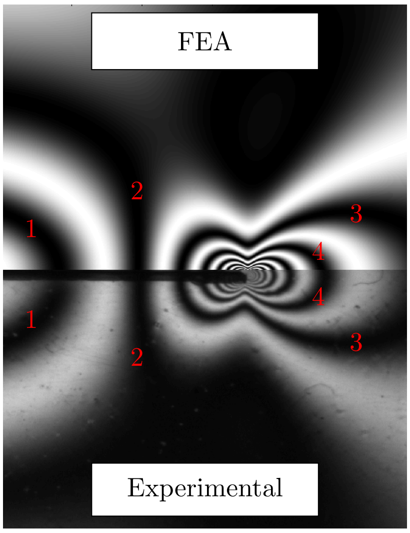


Figure 4.9: Top half represents fringe contour plotted from FEA whereas bottom half represents experimental dark field photoelastic fringe contour obtained for a tensile load of 60 N.

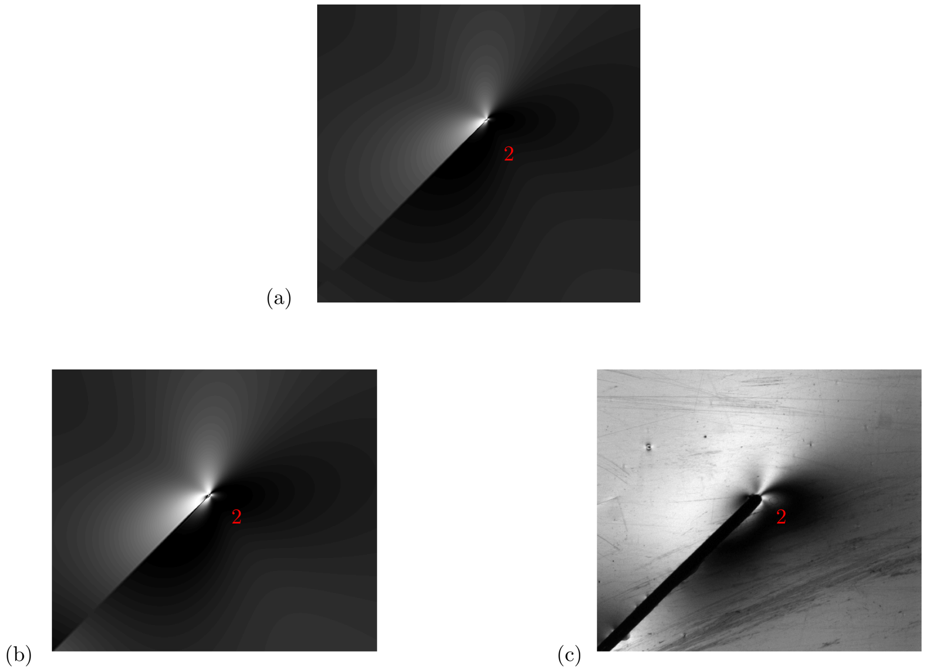


Figure 4.10: Comparison of dark field photoelastic fringe contour for a 45° inclined rigid line inclusion obtained for a tensile load of 45 N (a) reconstructed fringe contour from analytical solution (b) reconstructed fringe contour from FEA (c) fringe contour from photoelastic experiment.

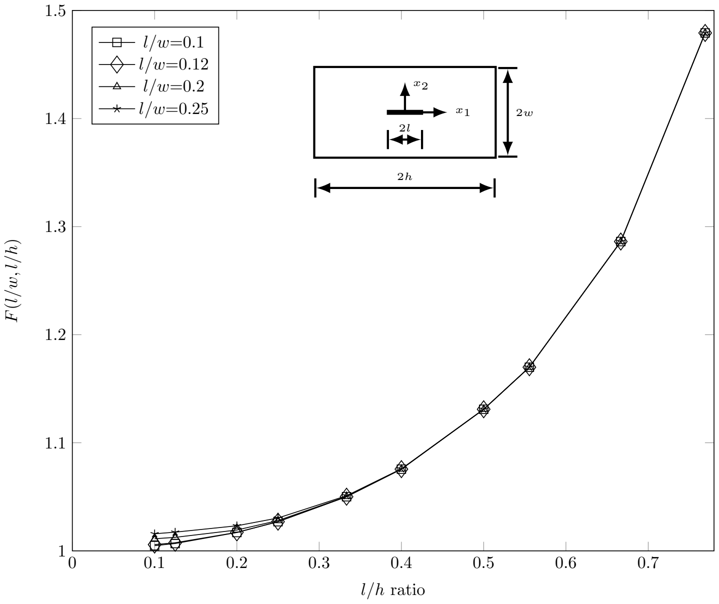


Figure 4.11: The geometrical correction factor for different l/h and l/w ratio obtained for the problem of straight line inclusion using FEA.

F approaches to unity for vanishing l/w and l/h. This indicates that the solution procedure reported in section 4.4 is correct. (2) The strain intensity factor increases with an increase in l/h. (3) The calculated correction factors are nearly insensitive to change in l/w ratio within the range of simulations performed. Recall, from the analytical solution (Eqn. 4.46) that the strain intensity factor is sensitive to the remote strain ϵ_{11}^{∞} only. Hence, we anticipate that the reported correction factors will be insensitive to any additional forms of loading.

4.6.4 Shear transfer length

Figure (4.12) show the shear stress variation along rigid line inclusion. The shear transfer length (l_s) is 3.64 mm from each side of inclusion tip towards center of inclusion. As one normalizes it with respect to length it turns out to be 0.36 l, where l is half rigid line inclusion length. It is arrived by considering a location where the normal stress (σ_{11}) matches with far field applied stress. This normal stress component is equivalent to that of peel stress in case of bonded joint. This section of inclusion is taking more shear stress as compared to central portion, where it is closer to zero.

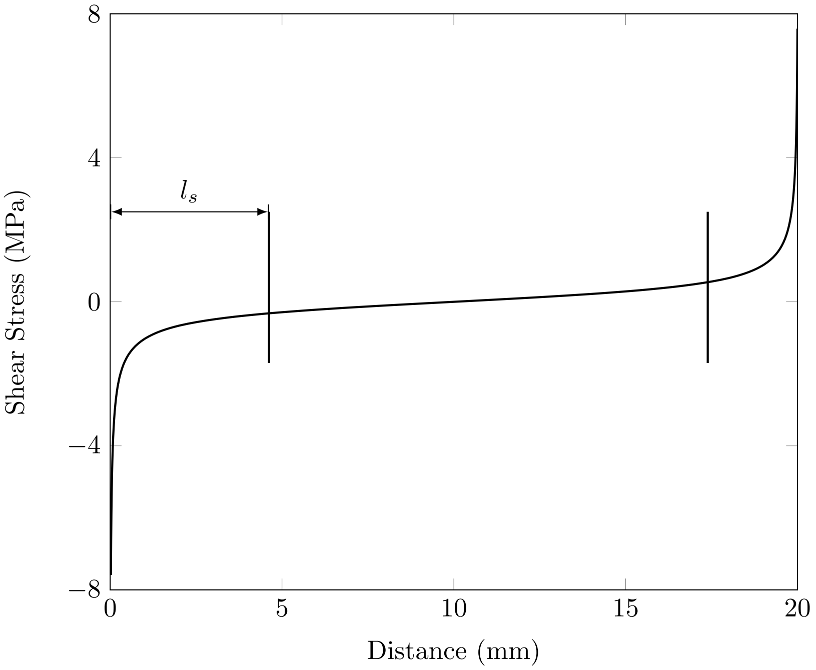


Figure 4.12: Shear stress distribution along inclusion length from one tip to another obtained from FEA for a straight inclusion of length $2l = 20\ mm$ being subjected to tensile load of 60 N.

Similarly, within the zone of shear transfer length the shear stress (σ_{12}) value is obtained much higher as that of central part of inclusion. It will cause interface debonding of line inclusion from the matrix as it exceeds the interface strength. This shear transfer length helps in deciding the minimum length of the inclusion to be maintained for an effective load transfer to take place.

4.6.5 Experimental evaluation of strain intensity factor

The whole field fringe order and isoclinic values are required to estimate the strain intensity factor using digital photoelasticity technique. By using the ten-step phase shifting method [34], isoclinic data is firstly generated and it is further unwrapped to get the wrapped isochromatic phasemap devoid of ambiguous zone. The AQGPU [1] algorithm is used to unwrap both isoclinic and isochromatic parameter. The same procedure as explained in Chapter 3, Sec 3.3 has been followed here for full field photoelastic data estimation.

Figure 4.13a shows the wrapped isoclinic phasemap and it is unwrapped to remove inconsistent zone as shown in Fig. 4.13b for the rigid line inclusion problem obtained at a tensile load of 45 N. The unwrapped isoclinic is used to get the isochromatic phasemap without any ambiguity and it is shown in Figure 4.13c. To verify the accuracy, this phasemap is compared with dark field photoelastic fringe contours as shown in Figure 4.13d. The wrapped isochromatic phasemap is then unwrapped to get the total fringe order over the model domain and it is shown in Figure 4.13e.

With automatic software interface, the fringe order and principal stress direction data are collected in annular zone around the inclusion tip. Also it contains pixel coordinates for each data and crack tip location which is approximately selected by using software interface. Material stress fringe value of the epoxy specimen (F_σ) is 0.27 N/mm/fringe. Here, proposed linear least square algorithm as explained in chapter 3 is followed along with the multi-parameter stress field equation derived recently for the rigid line inclusion problem. The selected inclusion tip act as origin for the coordinate system, where a square of 0.4 mm length and a grid size of 0.01 mm is created. For each grid point, the value of fracture parameters are computed. The value of fracture parameter is chosen such a way that the objective function has a minimum value in grid and corresponding K_I^ϵ is estimated. Figure 4.14 shows the normalized error plot with respect to selected the inclusion tip and zone of least error is marked which corresponds to exact inclusion tip location. The error J is defined as square of difference between experimental and reconstructed stress field data.

The experimentally evaluated value of strain intensity factor for a rigid line inclusion specimen is 0.0797 $\sqrt{\text{mm}}$ for 7-parameter solution. The corresponding error value J is 0.001 MPa^2. The new location for inclusion tip based on minimum error is $(0.07, -0.01)$ with respect to manually selected line inclusion tip location. The strain intensity factor value for a rigid line inclusion specimen is also estimated from the numerical method (see section 4.4) and it is found to be 0.0745 $\sqrt{\text{mm}}$. The experimental strain intensity value has got a deviation of 8% with respect to the numerical value. The strain intensity factor values obtained for different parameters such as 2, 4 and 7 are summarised in Table 4.1. For confirming accuracy of results, the reconstructed and experimental (red markers) fringes for a 7-parameter solution are superimposed over each other as shown in Fig. 4.15. Here, it is confirmed that the 7-parameter solution has good match with experiments. The convergence plot of SIF with respect to number of parameters are shown in Fig. 4.16). After seven parameters we can see that he value of K_I^ϵ has got stabilized hence double confirming the reconstructed field. It is the same in case of error convergence too (See Figure 4.17).

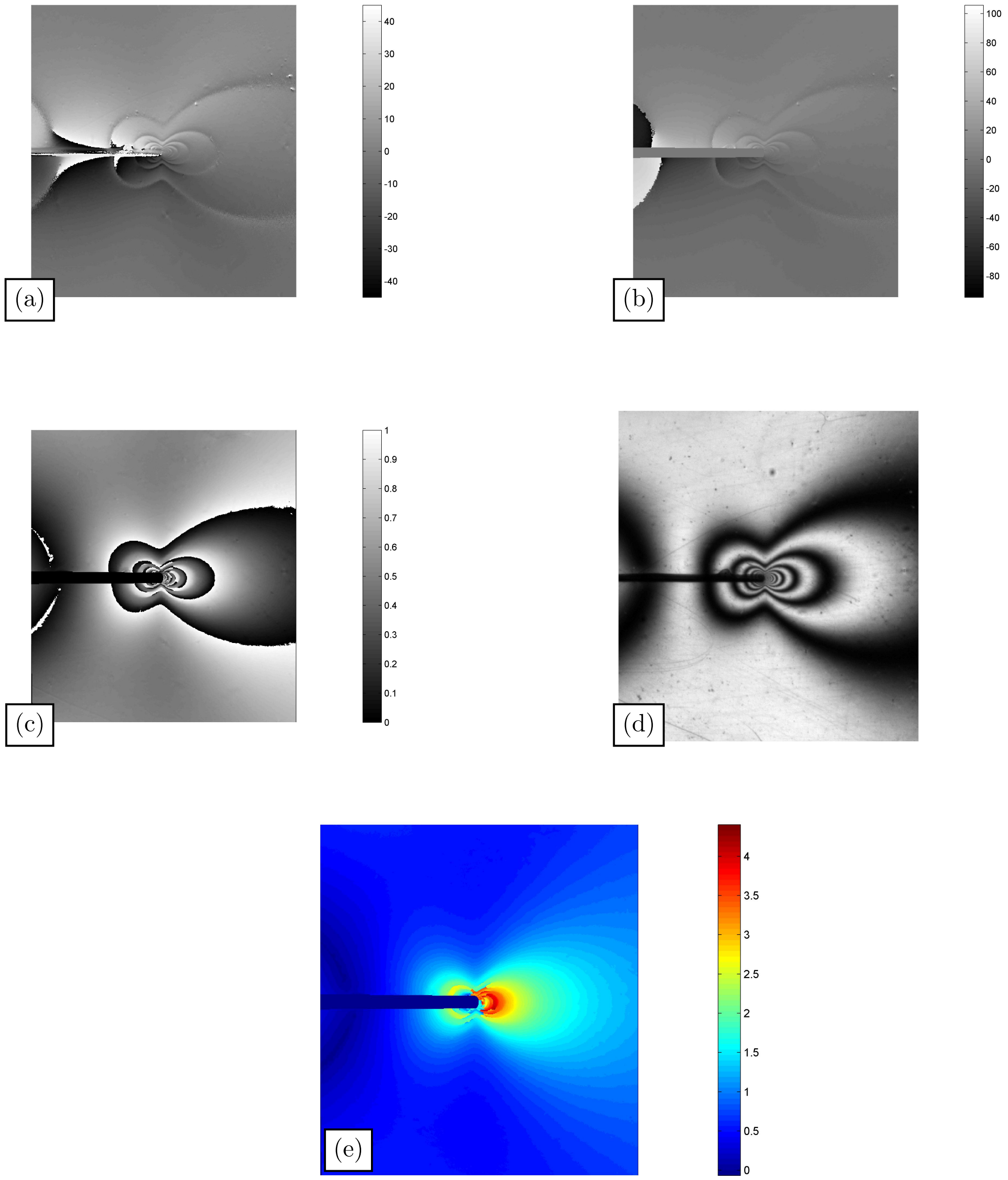


Figure 4.13: Near rigid inclusion tip isoclinic and isochromatic phasemap for rigid line inclusion specimen for a tensile load of 45 N (a) wrapped isoclinic (b) unwrapped isoclinic (c) wrapped isochromatic (d) dark field photoelastic image (e) unwrapped isochromatic.

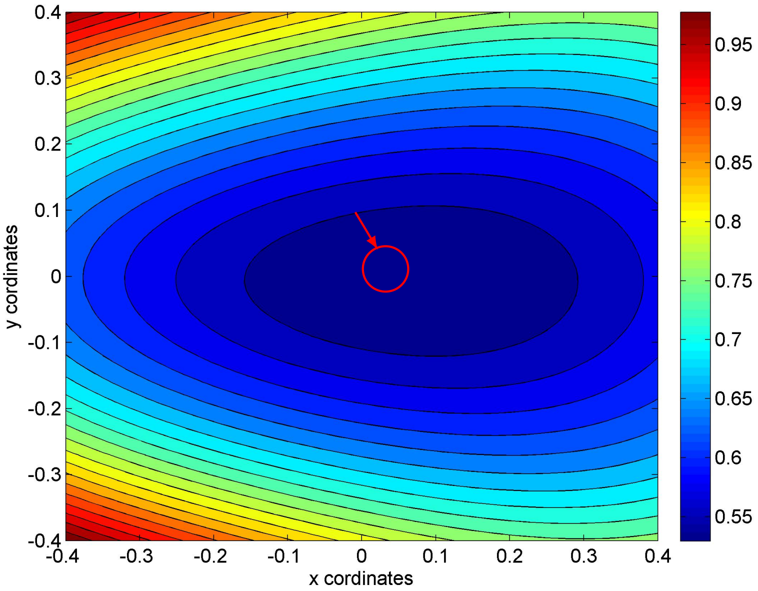


Figure 4.14: Normalized error plot over the square grid obtained for the experimental estimation of strain intensity factor for rigid line inclusion specimen

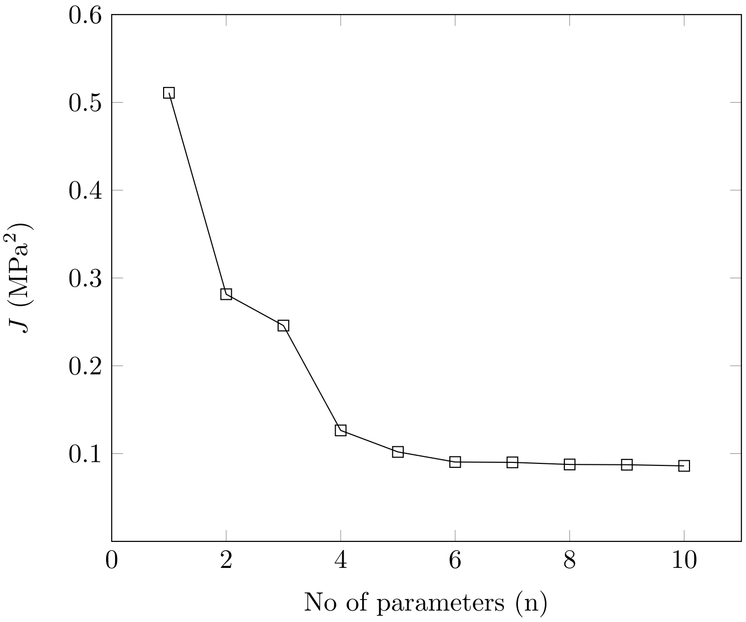


Figure 4.17: Variation of error with increasing number of parameters for the case of rigid line inclusion problem.

Table 4.1: inclusion tip fracture parameters for a rigid line inclusion specimen

	2-parameter	4-parameter	7-parameter
K_I^ϵ ($\sqrt{\text{mm}}$)	0.0482	0.0676	0.0797
A_{I1} ((mm)$^{1/2}$)	0.0402	0.0563	0.0664
A_{I2}	-0.0531	-0.0093	0.0610
A_{I3} ((mm)$^{-1/2}$)		0.0004	0.0026
A_{I4} ((mm)$^{-1}$)		0.0008	0.0006
A_{I5} ((mm)$^{-3/2}$)			-0.0001
A_{I6} ((mm)$^{-2}$)			0.0000
A_{I7} ((mm)$^{-5/2}$)			0.0000

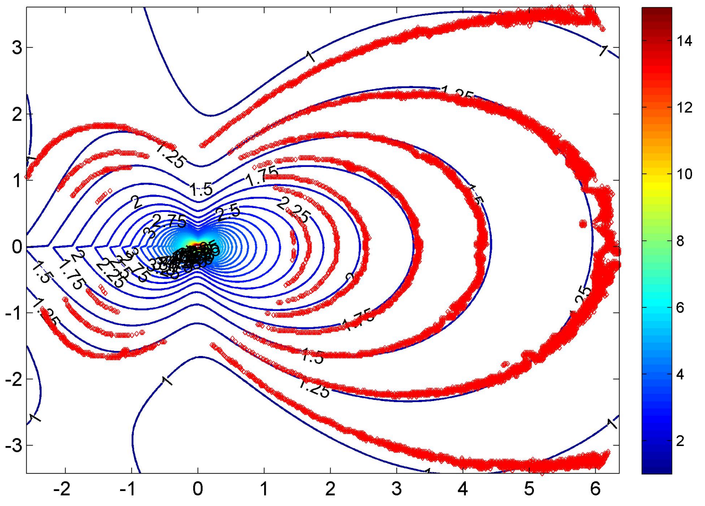


Figure 4.15: Experimental (red markers) and reconstructed principal stress difference fields (fringe contour) obtained for 7 parameter solution superposed over each other in case of rigid line inclusion specimen for tensile load of 45 N

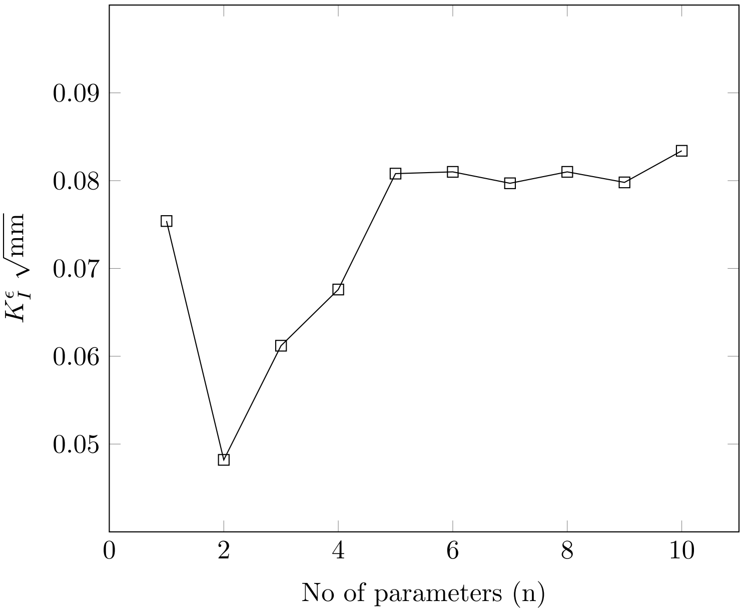


Figure 4.16: Variation of mode-I strain intensity factor with increasing number of parameters for rigid line inclusion specimen.

4.7 Closure

Explicit analytical expressions for the elastic stress, strain and displacement fields are obtained for the problem of a rigid line inclusion embedded in an isotropic matrix over complete domain. The solutions are obtained using Stroh's formulation in tandem with dual reciprocal theorem. Both straight and inclined rigid line inclusion problem under tensile load has been studied. It is found that the square root singularity is present at the tip of rigid line inclusion. The strain intensity factor is defined and it is found to be independent of matrix material properties. From the analytical solution, one can confirm that characterizing the singularity in-terms of remote strain rather than stress would lead to a simpler expression. Further, it is dependent on the longitudinal strain component. Using reciprocal theorem and FEA, strain intensity factor has been determined. The strain intensity factor obtained from the numerical technique compares closely with the analytical estimate thereby confirming the accuracy of the FE prediction. The geometry correction factor as well as the shear transfer length is also predicted using FEA in order to make the study complete. Finally, the strain intensity factor is estimated by using the digital photoelastic technique involving linear least square algorithm and it compares good against the numerical estimate.

Chapter 5

Conclusions and Recommendations

A GUI based in-house software is developed to get the unwrapped isoclinic and isochromatic phasemap based on the ten-step phase shifting algorithm. The isoclinic and isochromatic phasemap obtained for the disc is compared with the analytical and they agree very well thereby confirming the accuracy of the implementation. In addition, isoclinic and isochromatic phasemap are obtained for the problem of both ring under diametrical compression and plate with circular hole under tensile load. The study is carried using both monochrome and white light source. In order to avoid isoclinic-isochromatic interaction white light source is preferred where one gets colored images. In future, for industrial application isochromatic phasemap would be developed using single image.

In the present work, a new algorithm is proposed for the estimation of mixed-mode fracture parameters by solving multi-parameter stress field equations using the isochromatic and isoclinic data from digital photoelasticity technique. Along with the fracture parameters, the algorithm facilitates the extraction of crack tip location. A SIF estimation module based on the new algorithm for SIF estimation has been developed for the estimation of mixed mode SIFs for cracked specimens. The three cracked specimen configurations namely single edge notch, center slant crack and crack-inclusion specimen made from epoxy material are studied. An improved over deterministic linear least square algorithm is successfully implemented for SIF measurement involving multi-parameter stress field equation. This approach is much faster and more accurate than the conventional non-linear over deterministic least square scheme existing in the literature. Additionally, an optimization based technique is also integrated to determine the exact crack tip coordinate location. The estimated SIF values for the three configurations are found to be in close match with either analytical or FEA estimates, thereby confirming the accuracy of the developed methodology.

In addition, the problem of rigid line inclusion embedded in an elastic matrix is studied. Explicit analytical expressions for the elastic stress, strain and displacement fields are obtained for the problem of rigid line inclusion embedded in an isotropic matrix over the complete domain. It is found that the square root singularity is present at the tip of rigid line inclusion. The strain intensity factor is found to be appropriate parameter to quantify singularity as it is independent of matrix material property. Using reciprocal theorem and FEA, strain intensity factor has been determined. Further, multi-parameter stress field equation for rigid line inclusion problem is derived. An over deterministic linear least square technique is used for strain intensity factor estimation involving multi-parameter stress field equation using digital photoelasticity technique. The strain intensity factor obtained from both numerical and experimental approach compares closely with the numerical value thereby confirming the accuracy of the implemented algorithm.

In actual practice, the developed algorithm for SIF estimation for multiple defects would involve many challenges such as grid selection, accurate estimation of crack tip coordinate etc. Also the study of single rigid line inclusion is an ideal case, this can be extended to multiple rigid line inclusions with different orientations to understand fiber-matrix interaction in short fiber composites.

References

[1] M. Ramji and K. Ramesh. Adaptive Quality Guided Phase Unwrapping Algorithm for Whole-Field Digital Photoelastic Parameter Estimation of Complex Models. *Strain* 46, (2010) 184–194.

[2] T. Dudderar and H. Gorman. The determination of mode I stress-intensity factors by holographic interferometry. *Experimental Mechanics* 13, (1973) 145–149.

[3] A. Moore and J. Tyrer. Phase-stepped ESPI and moiré interferometry for measuring stress-intensity factor andJ integral. *Experimental mechanics* 35, (1995) 306–314.

[4] H. Rossmanith. Analysis of crack-tip Moire fringe patterns. *International Journal of Fracture* 21, (1983) 83–106.

[5] S. Ramaswamy, H. Tippur, and L. Xu. Mixed-mode crack-tip deformations studied using a modified flexural specimen and coherent gradient sensing. *Experimental mechanics* 33, (1993) 218–227.

[6] Y. Kawagishi, M. Shozu, and Y. Hirose. Experimental evaluation of stress field around crack tip by caustic method. *Mechanics of materials* 33, (2001) 741–757.

[7] R. J. Sanford and J. W. Dally. A general method for determining mixed-mode stress intensity factors from isochromatic fringe patterns. *Engineering Fracture Mechanics* 11, (1979) 621–633.

[8] S. Yoneyama, Y. Morimoto, and M. Takashi. Automatic Evaluation of Mixed-mode Stress Intensity Factors Utilizing Digital Image Correlation. *Strain* 42, (2006) 21–29.

[9] R. Harilal, C. Vyasarayani, and M. Ramji. A linear least squares approach for evaluation of crack tip stress field parameters using DIC. *Optics and Lasers in Engineering* 75, (2015) 95–102.

[10] K. Ramesh. Digital photoelasticity: advanced techniques and applications. Springer-Verlag, Berlin, 2000.

[11] M. Ramji, V. Gadre, and K. Ramesh. Comparative study of evaluation of primary isoclinic data by various spatial domain methods in digital photoelasticity. *The Journal of Strain Analysis for Engineering Design* 41, (2006) 333–348.

[12] M. Ramji and R. Prasath. Sensitivity of isoclinic data using various phase shifting techniques in digital photoelasticity towards generalized error sources. *Optics and Lasers in Engineering* 49, (2011) 1153–1167.

[13] J. Dally and R. Sanford. Classification of stress-intensity factors from isochromatic-fringe patterns. *Experimental Mechanics* 18, (1978) 441–448.

[14] R. J. Sanford. Application of the least-squares method to photoelastic analysis. *Experimental Mechanics* 20, (1980) 192–197.

[15] H. Nigam and A. Shukla. Comparison of the techniques of transmitted caustics and photoelasticity as applied to fracture. *Experimental Mechanics* 28, (1988) 123–131.

[16] R. J. Sanford. Determining fracture parameters with full-field optical methods. *Experimental Mechanics* 29, (1989) 241–247.

[17] K. Ramesh, S. Gupta, and A. A. Kelkar. Evaluation of stress field parameters in fracture mechanics by photoelasticityrevisited. *Engineering Fracture Mechanics* 56, (1997) 25–45.

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

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

[20] A. K. S.N. Atluri. Mechanical responses of materials. *In Handbook on Experimental Mechanics* 1–37.

[21] M. Guagliano, M. Sangirardi, A. Sciuccati, and M. Zakeri. Multiparameter analysis of the stress field around a crack tip. *Procedia Engineering* 10, (2011) 2931–2936.

[22] S.-Y. Fu, B. Lauke, and Y.-W. Mai. Science and engineering of short fibre reinforced polymers composites. Woodhead Pub, Cambridge, UK, 2009.

[23] D. Hull and T. Clyne. An introduction to composite materials. Cambridge University Press, Cambridge, UK, 1996.

[24] N. I. Muskhelishvili. Some basic problems of the mathematical theory of elasticity. Springer Science & Business Media, 1989.

[25] C. Atkinson. Some ribbon-like inclusion problems. *International Journal of Engineering Science* 11, (1973) 243–266.

[26] Z. Wang, H. Zhang, and Y. Chou. Characteristics of the elastic field of a rigid line inhomogeneity. *Journal of Applied Mechanics* 52, (1985) 818–822.

[27] R. Ballarini. An integral equation approach for rigid line inhomogeneity problems. *International Journal of Fracture* 33, (1987) R23–R26.

[28] N. Hasebe, L. Keer, and S. Nemat-Nasser. Stress analysis of a kinked crack initiating from a rigid line inclusion. Part 1: Formulation. *Mechanics of Materials* 3, (1984) 131–145.

[29] N. Hasebe, S. Nemat-Nasser, and L. Keer. Stress analysis of a kinked crack initiating from a rigid line inclusion. Part II: direction of propagation. *Mechanics of Materials* 3, (1984) 147–156.

[30] J. Hurtado, J. Dundurs, and T. Mura. Lamellar inhomogeneities in a uniform stress field. *Journal of the Mechanics and Physics of Solids* 44, (1996) 1–21.

[31] Q. Li and T. Ting. Line inclusions in anisotropic elastic solids. *Journal of applied mechanics* 56, (1989) 556–563.

[32] S. Nemat-Nasser and L. Ni. A duality principle and correspondence relations in elasticity. *International journal of solids and structures* 32, (1995) 467–472.

[33] G. Noselli, F. Dal Corso, and D. Bigoni. The stress intensity near a stiffener disclosed by photoelasticity. *International journal of fracture* 166, (2010) 91–103.

[34] M. Ramji and K. Ramesh. Whole field evaluation of stress components in digital photoelasticityissues, implementation and application. *Optics and Lasers in Engineering* 46, (2008) 257–271.

[35] G. Petrucci. Full-field automatic evaluation of an isoclinic parameter in white light. *Experimental Mechanics* 37, (1997) 420–426.

[36] D. Dugdale. Yielding of steel sheets containing slits. *Journal of the Mechanics and Physics of Solids* 8, (1960) 100–104.

[37] A. Shukla, B. Agarwal, and B. Bhushan. Determination of stress intensity factor in orthotropic composite materials using strain gages. *Engineering Fracture Mechanics* 32, (1989) 469–477.

[38] H. Tada, P. Paris, and G. Irwin. The analysis of cracks handbook. New York: ASME Press, 2000.

[39] Y. Murakami and N. Hasebe. Stress intensity factors handbook. Elsevier Science, 2001.

[40] C. Hwu. Anisotropic elastic plates. Springer Science & Business Media, 2010.

[41] T. C. T. Ting. Anisotropic elasticity - Theory and Applications. Oxford University Press, USA, 1996.

[42] L. Ni and S. Nemat-Nasser. A general duality principle in elasticity. *Mechanics of materials* 24, (1996) 87–123.

[43] X. Markenscoff. On the Dundurs correspondence between cavities and rigid inclusions. *Journal of applied mechanics* 60, (1993) 260–264.

[44] A. Akisanya and N. Fleck. Interfacial cracking from the freeedge of a long bi-material strip. *International Journal of Solids and Structures* 34, (1997) 1645–1665.

[45] W. C. Carpenter and C. Byers. A path independent integral for computing stress intensities for V-notched cracks in a bi-material. *International Journal of Fracture* 35, (1987) 245–268.

[46] Z. Qian and A. Akisanya. Wedge corner stress behaviour of bonded dissimilar materials. *Theoretical and Applied Fracture Mechanics* 32, (1999) 209–222.

[47] J. R. Barber. Elasticity. Springer, 2009.

[48] Abaqus. version 6.9. Dassault Systems, Simulia Corporation, Providence, RhodeIsland, USA, 2009.

[49] MATLAB. version 8.0 (R2012b). The MathWorks Inc., Natick, Massachusetts, 2012.

[50] K. Ramesh, A. Yadav, and V. A. Pankhawalla. Plotting of fringe contours from finite element results. *Communications in Numerical Methods in Engineering* 11, (1995) 839–847.

Appendix A: Ten-step PST photoelastic images

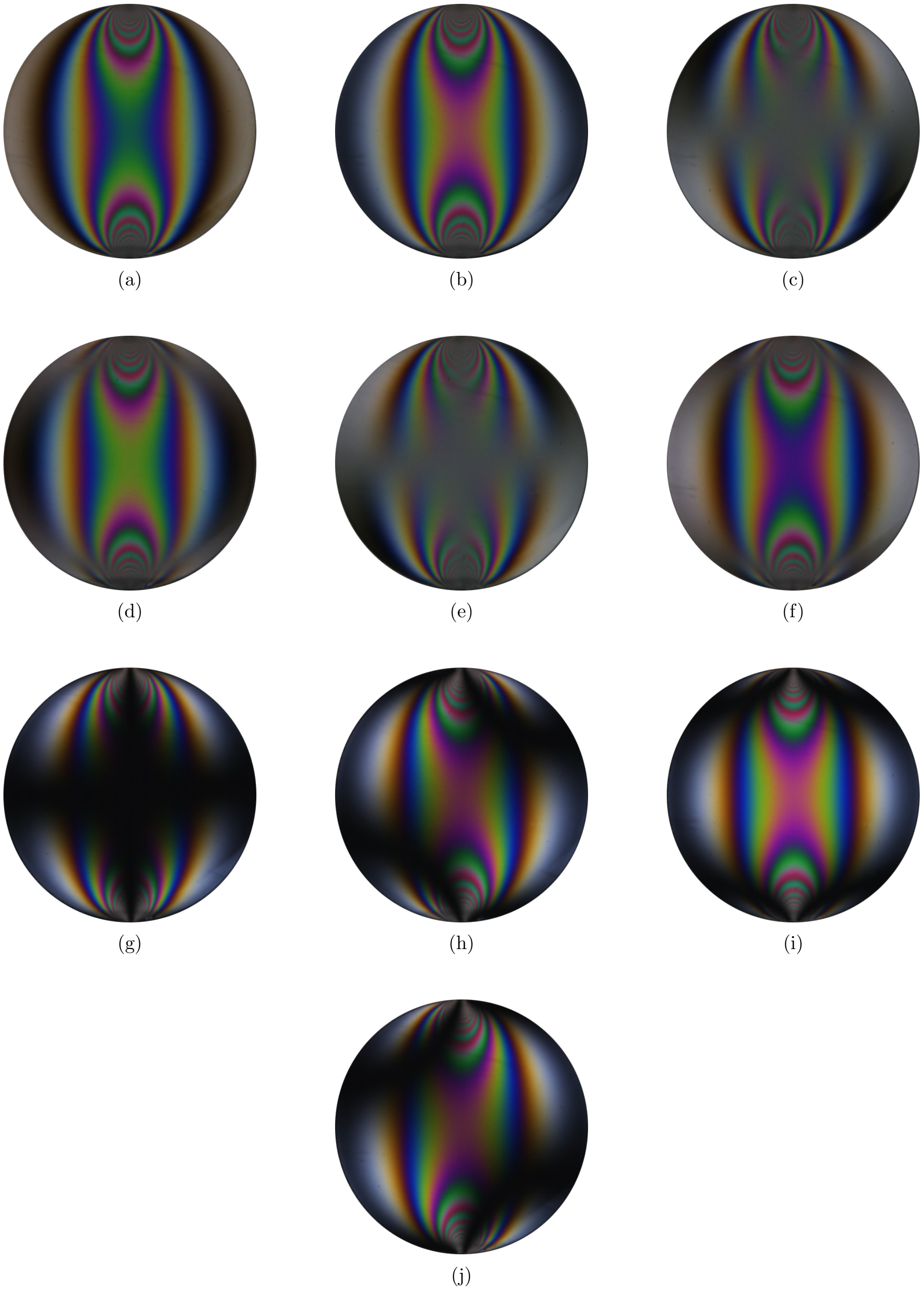
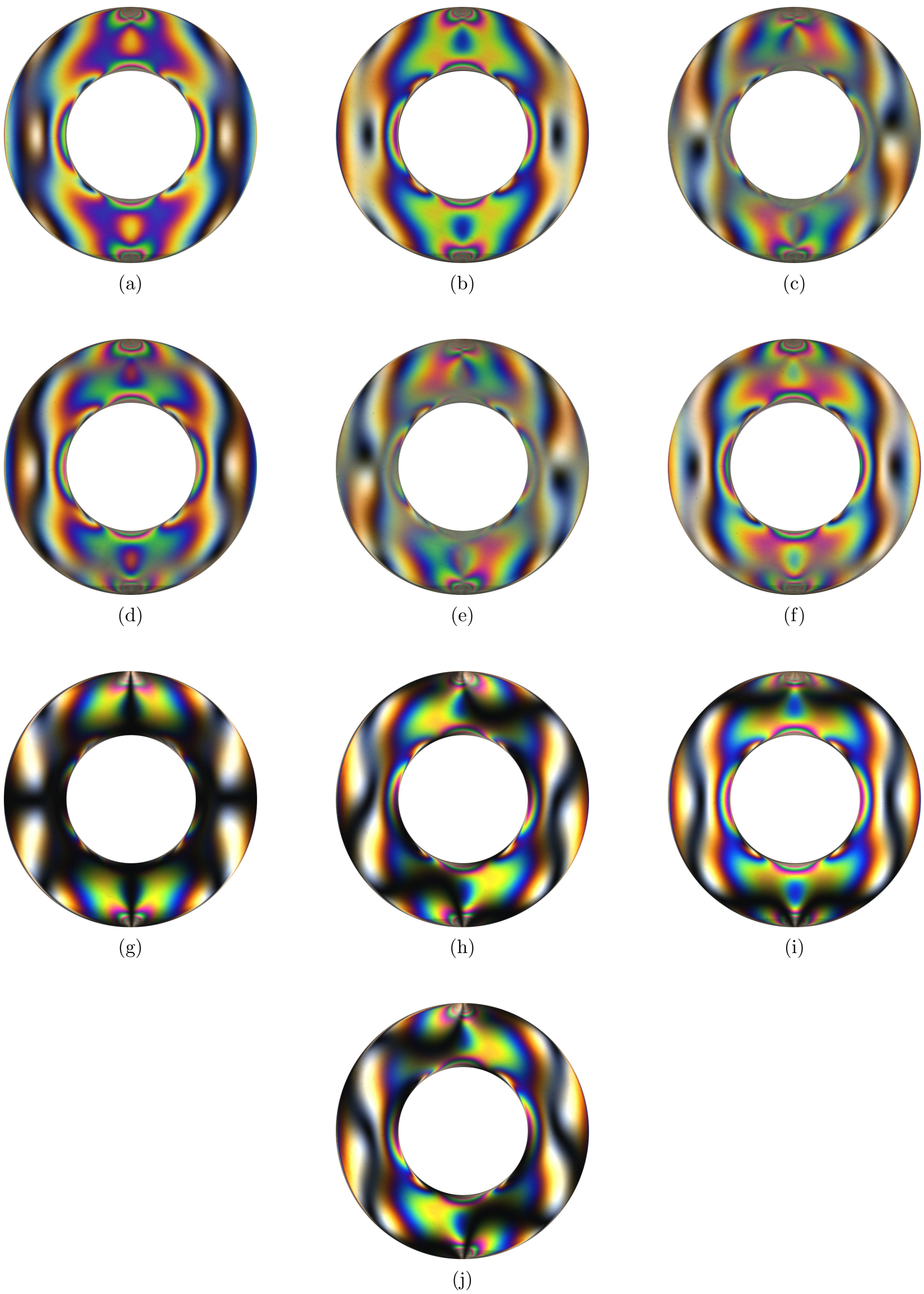


Figure A.1: Experimentally recorded phase shifted images of disc under diametrical compression (500 N) corresponding to ten-step phase shifting algorithm as per the sequence given in table 2.1.

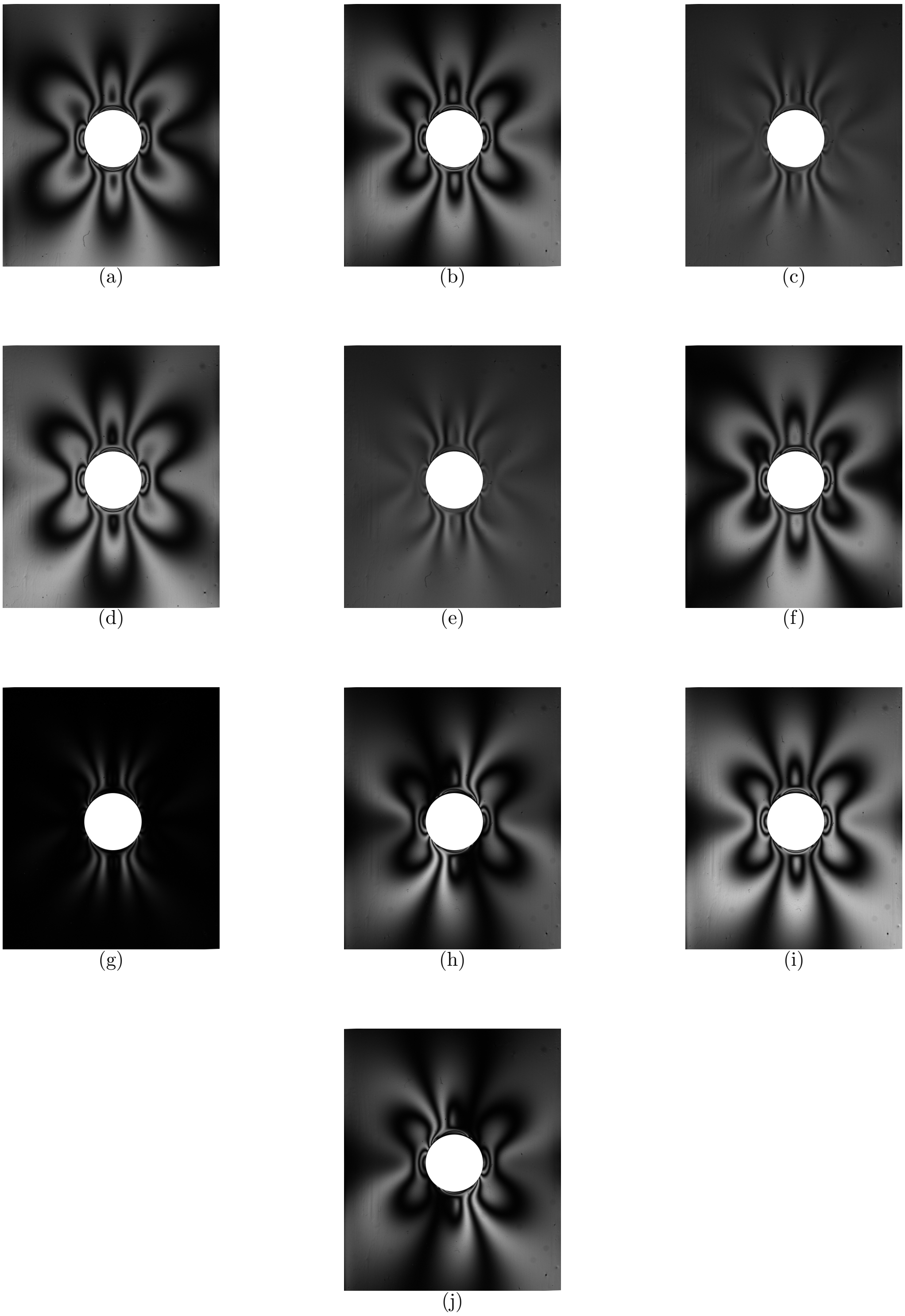

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

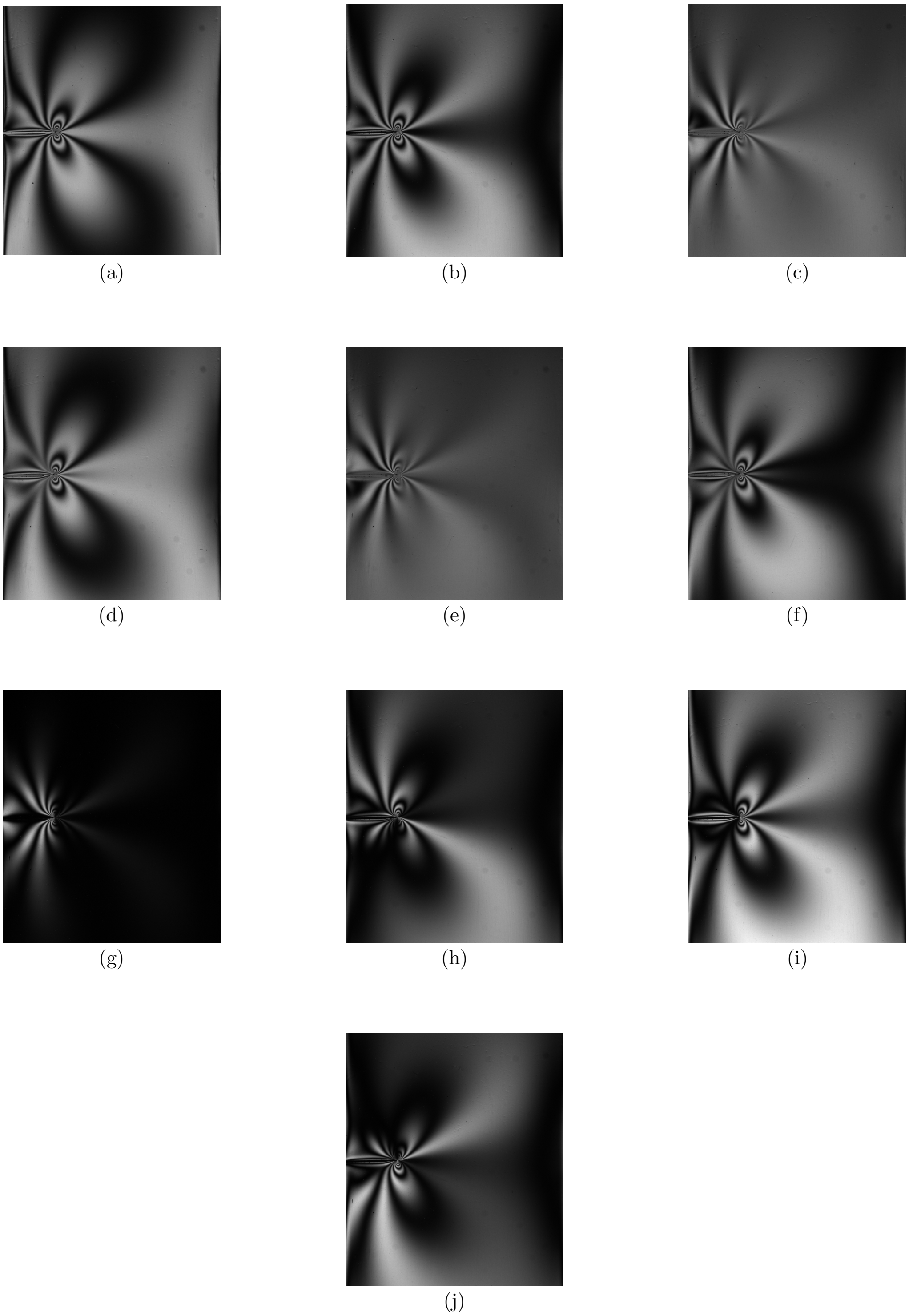
(j)

Figure A.2: Experimentally recorded phase shifted images of ring under diametrical compression (250 N) corresponding to ten-step phase shifting algorithm as per the sequence given in table 2.1.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure A.3: Experimentally recorded phase shifted images of plate with hole subjected to a tensile load (700 N) corresponding to ten-step phase shifting algorithm as per the sequence given in table 2.1.



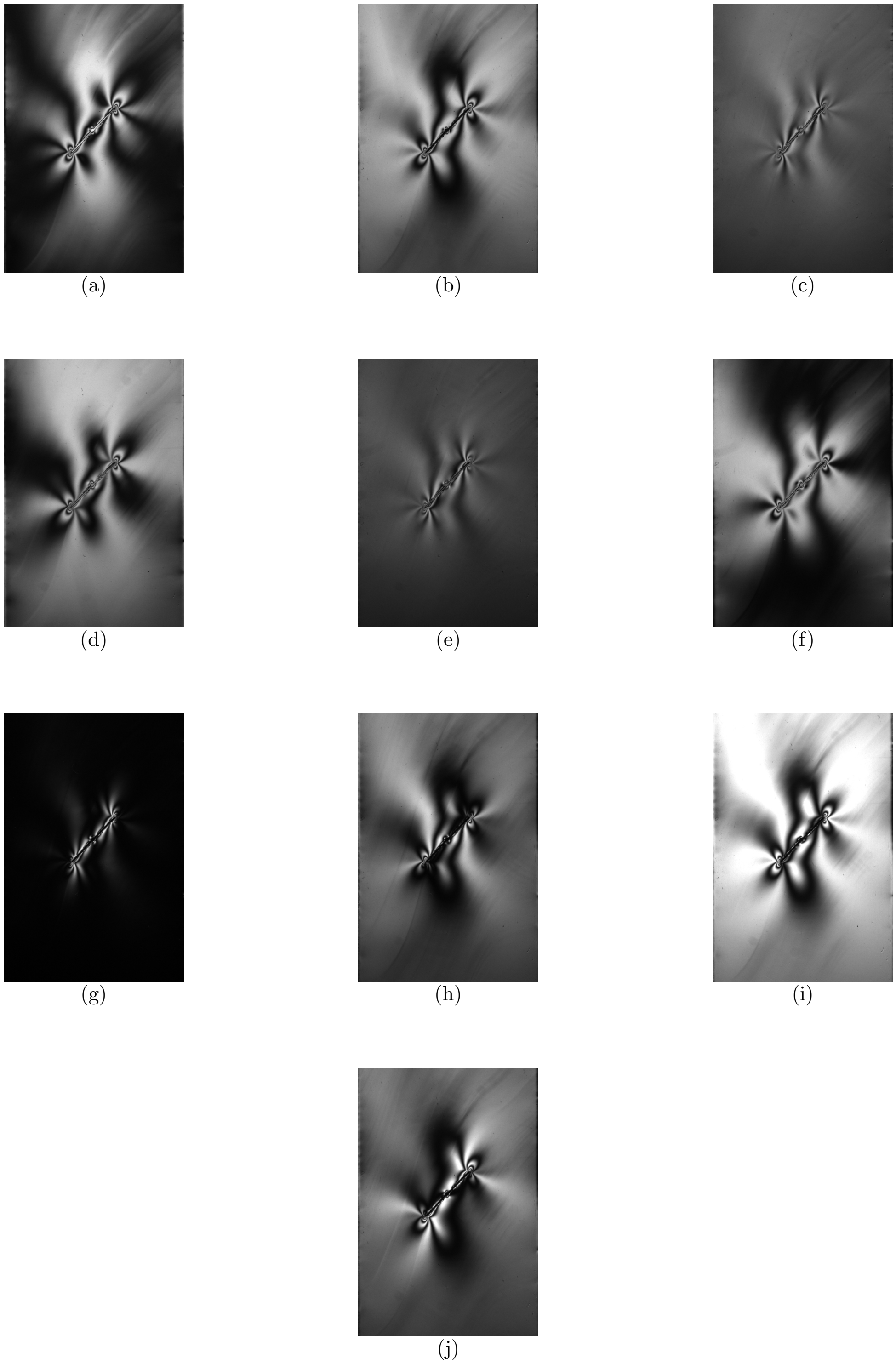
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Figure A.4: Experimentally recorded phase shifted images of single edge notch (SEN) specimen subjected to a tensile load (705 N) corresponding to ten-step phase shifting algorithm as per the sequence given in table 2.1.



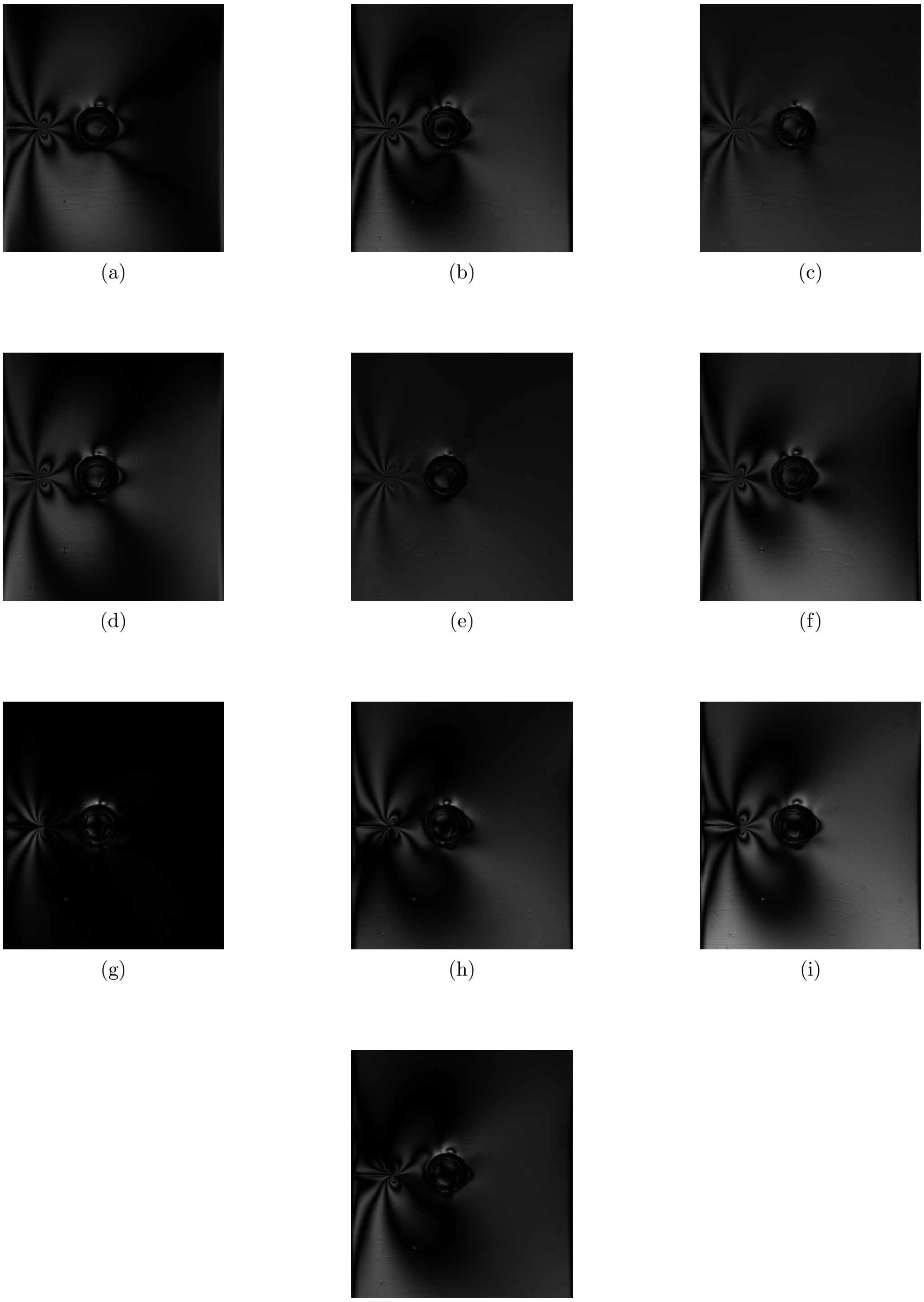
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Figure A.5: Experimentally recorded phase shifted images of center slant crack (CSC) specimen subjected to a tensile load (600 N) corresponding to ten-step phase shifting algorithm as per the sequence given in table 2.1.



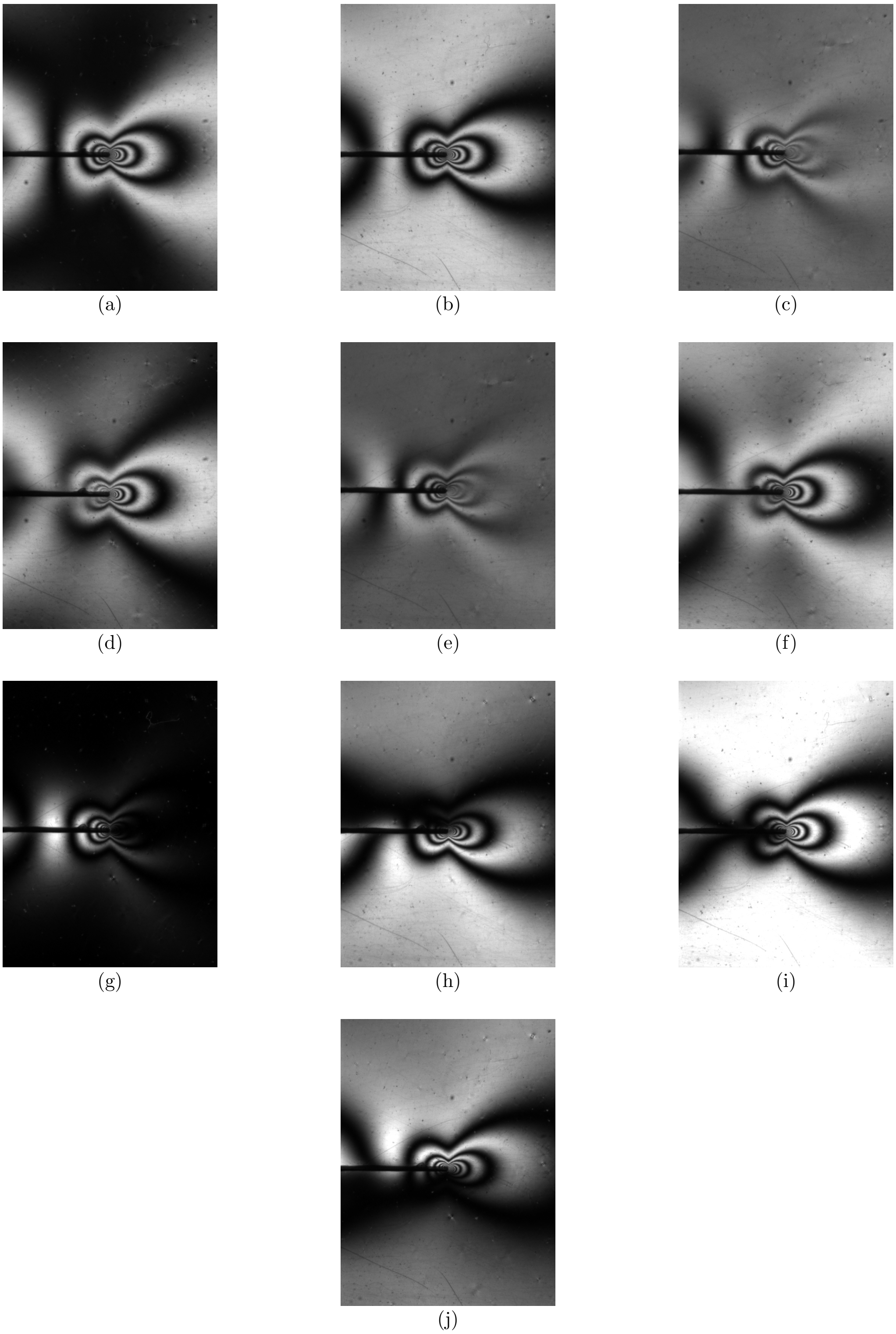
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Figure A.6: Experimentally recorded phase shifted images of crack-inclusion specimen subjected to a tensile load (640 N) corresponding to ten-step phase shifting algorithm as per the sequence given in table 2.1.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure A.7: Experimentally recorded phase shifted images of rigid line inclusion embedded in an elastic matrix subjected to a tensile load (45 N) corresponding to ten-step phase shifting algorithm as per the sequence given in table 2.1.

Appendix B: Python and MATLAB Scripts

MATLAB Codes;

– SIF for cracks with linear least square approach Interface for automatic data collection:

```
%%%%%% Data collection around crack tip for SIF calculation %%%%%%%%%%%%%
%%% Input data files— Unwrapped isoclinic: 'nm_theta.iso', Unwrapped ...
    isochromatic: 'nm_uw∆.iso'
%%% and dark photoelastic field image: 'ajo2.tiff'
clc;
clear all;

%%% Unwrapped isoclinic and isochromatic .iso file reading
∆=importdata('....._uw∆.iso');
theta=importdata('....._uwtheta.iso');

%%% Annular region radius input
Ri=inputdlg('minimum Radius ; maximum radius :','Radius for Data ...
    collection',1);
r = str2num(Ri{1});

%%% Reading photoelastic dark field image
a = imread('ajo2.tiff');
B=a(:,:,2);

f = figure;
h = uicontrol('Position',[200 200 200 40],'String','Proceed for Crack ...
    tip selection',...
              'Callback','uiresume(gcbf)');
disp('This will print immediately');
uiwait(gcf);
disp('This will print after you click Continue');
close(f)
figure(1)
imagesc(B);

%%% Selecting crack tip
crack_tip = round(ginput(1));
close
```

```

f = figure;
h = uicontrol('Position',[200 200 200 40],'String','Proceed for ...
    calibration',...
              'Callback','uiresume(gcbf)');
disp('This will print immediately');
uiwait(gcf);
disp('This will print after you click Continue');
close(f)
figure (2)
imagesc(B);
%%% Calibration of specimen in horizontal direction only
h1=helpdlg('Select first pixel from the figure');
uiwait(h1);
P1=round(ginput(1));
h2=helpdlg('Select second pixel from the figure');
uiwait(h2);
P2=round(ginput(1));
close
ac=inputdlg('Enter Distance in mm:','Two pixel distance calibration',1);
Dist = str2double(ac{1});
scale=Dist/abs(P1(1)-P2(1));
%%%% for vertical, use P1(2) - P2(2) instead of P1(1) - P2(1)
foi=inputdlg('Fringe No:','Fringe No for Data collection',1);

fo = str2num(foi{1});
sf=size(fo);
toli=inputdlg('Enter tolerance:','tolerance for fringe data collection',1);
tol = str2double(toli{1});
s=size(Δ);
[Px Py]=meshgrid(1:1:s(2), 1:1:s(1));

%%% data collection for Δ, theta and co-ordinates
Pxc=(Px-crack_tip(1))*scale;
Pyc=-(Py-crack_tip(2))*scale;

[th Rm]=cart2pol(Pxc,Pyc);
data=[];
data(1,:)=[crack_tip(1) crack_tip(2) Dist 6 11 0];
data(2,:)=[crack_tip(1)*scale crack_tip(2)*scale r(1) r(2) scale 0];
t_lim=7*pi/8 ;
```

```
c=3;
for i=1:1:s(1)

    for j=1:1:s(2)


        if  r(1)≤Rm(i,j)&& Rm(i,j)≤r(2) && abs(th(i,j)) ≤ t_lim
            for k=1:1:sf(2)

                if fo(k)-tol≤ ∆(i,j) && ∆(i,j)≤ fo(k)+tol

                    data(c,:)=[Pxc(i,j) Pyc(i,j) ∆(i,j) i j theta(i,j)];
                    c=c+1;
                end

            end
            k=0;
        end


    end


end
t=size(data);
x_data=data(2:t(1),5);
y_data=data(2:t(1),4);
figure,imagesc(B);axis equal, colormap gray
hold on,
plot(x_data,y_data,'rd','MarkerSize',3);

%%% saving data
save('collection.txt', 'fo', '-ASCII');
save('Data.txt','data','-ASCII');
```

Experimental SIF determination:

```

%%%% SIF estimation by linear least square approach%%%%%%%%%%%%
%%% Input data files- collected data: 'Data.txt' and collected input ...
```

```
    data file: 'collection.txt'
clc;
clear all;
close all;
%%% Reading collected data
colle=importdata('collection.txt');
data=importdata('Data.txt');
crack_tip_x=-0.4:0.01:0.4;  %%% selection of Grid around crack
crack_tip_y=0.4:-0.01:-0.4;
r1=data(2,3):0.1:data(2,4);
fsigma=11.0;%data(1,5);   %%% material stress fringe value
h=data(1,4); %%% thickness of specimen

n_start=7;  %%% number of parameter : starting and total
n_total=7;
[xx,yy]=meshgrid(crack_tip_x,crack_tip_y);
data_crc=data(3:size(data,1),:);
a=size(data_crc);

trans=ones(a(1),1);

    main_count=1;
    len=length(crack_tip_x)*length(crack_tip_y)*(n_total-n_start+1);
    saved_data = zeros(len,7);
    A=cell(len,1);
%progressbar('Crack tip x coordinate search','Crack tip y coordinate ...
    search','Annular radii search','Parameter Search');
%%% loop over grids in x-direction
for ctx=1:1:length(crack_tip_x)
    %progressbar([],0,0,0);
    %%% loop over grids in y-direction
    for cty=1:1:length(crack_tip_y)
     translated_coord_x=data_crc(:,1:2)-[xx(ctx,cty)*trans ...
         yy(ctx,cty)*trans];
     % progressbar([],[],0,0);
     %for r1c=1:1:length(r1)
     x_data=translated_coord_x(:,1);
     y_data=translated_coord_x(:,2);
     ps_data=data_crc(:,3);
     theta_data=data_crc(:,6);
     [theta,r] = cart2pol(x_data,y_data);
```

```
%progressbar([],[],[],0);
%%% loop over number of parameters
   for n=n_start:1:n_total
    F1 = zeros(size(data_crc,1),n);
    F2 = zeros(size(data_crc,1),n);
    G1 = zeros(size(data_crc,1),n);
    G2 = zeros(size(data_crc,1),n);

    for i=1:1:n
           %%% multi-parameter stress field equation for crack
           %%% sigma_xx-sigma_yy= r^((n/2-1)) (A_In * F1 + A_IIn * F2)
           %%% Tau_xy= r^((n/2-1)) (A_In * G1 + A_IIn * G2)
    F1(:,i)=i/2*r.^((i-2)/2).*(2*((-1)^i+i/2)*cos((i/2-1)*theta)...
                               -2*(i/2-1)*cos((i/2-3)*theta));
    F2(:,i)=i/2*r.^((i-2)/2).*(2*(-(-1)^i+i/2)*sin((i/2-1)*theta)...
                                -2*(i/2-1)*sin((i/2-3)*theta));
    G1(:,i)=i/2*r.^((i-2)/2).*(-((-1)^i+i/2)*sin((i/2-1)*theta)...
                                        +(i/2-1)*sin((i/2-3)*theta));
    G2(:,i)=i/2*r.^((i-2)/2).*(-((-1)^i-i/2)*cos((i/2-1)*theta)...
                                        -(i/2-1)*cos((i/2-3)*theta));
    end
           C_Assembled=[F1 -F2;G1 -G2];
           s1_s2=ps_data*fsigma/h;
           sigma_Assembled=[s1_s2.*cos(2*(theta_data)) ...
               ;s1_s2.*sin(2*(theta_data))/2];
           unknowns =  pinv(C_Assembled) * sigma_Assembled;
           %%% SIF definition in terms of fracture parameter
           K_I = unknowns(1,1)*sqrt(2*pi);
           K_II = -unknowns(n+1,1)*sqrt(2*pi);

           sigma_reconstructed = C_Assembled * unknowns;
           error_store = (sigma_reconstructed-sigma_Assembled).^2;
           convergence_error = ...
               sum(error_store)/size(sigma_reconstructed,1);

               A{main_count,1}=unknowns;
               saved_data(main_count, 1) = crack_tip_x(cty);
               saved_data(main_count, 2) = crack_tip_y(ctx);
               %saved_data(main_count, 3) = r1(r1c);
               saved_data(main_count, 4) = n;
               saved_data(main_count, 5) = convergence_error;
```

```
                    error(ctx,cty)=convergence_error;
                    saved_data(main_count, 6) = K_I;
                    saved_data(main_count, 7) = K_II;
                    main_count = main_count+1;
                    %progressbar([],[],[],n/n_total);
            end
           % progressbar([],[],r1c/length(r1));
        %progressbar([],cty/length(crack_tip_y));
     end
    % progressbar(ctx/length(crack_tip_x));
end
[m ,i]=min((saved_data(1:length(A),5)));
z=i;
saved_data(z,:)
saved_data(z,6:7)
saved_data(z,5)
saved_data(z,1)
saved_data(z,2)
A_coef=A{z};
xx=data_crc(:,1)-saved_data(z,1)*trans;
yy=data_crc(:,2)-saved_data(z,2)*trans;

%%%Reconstructing fringe data
scale=data(2,5);
s = (min(xx):scale:max(xx));       % limits for plotting in x-direction
o = (min(yy):scale:max(yy));       % limits for plotting in y-direction
[X,Y] = meshgrid(s,o);
[t,r] = cart2pol(X,Y);

sx_syp = zeros(length(o),length(s));
sxyp = zeros(length(o),length(s));
Np = zeros(length(o),length(s));

for p=1:1:length(A_coef)/2
 sx_syp=sx_syp+A_coef(p)*p/2*r.^((p-2)/2).*(2*((-1)^p+p/2)*cos((p/2-1)*t)...
 -2*(p/2-1)*cos((p/2-3)*t))-A_coef(p+length(A_coef)/2)*p/2*r.^((p-2)/2).*(2*...
 (-(-1)^p+p/2)*sin((p/2-1)*t)-2*(p/2-1)*sin((p/2-3)*t));
 sxyp=sxyp+A_coef(p)*p/2*r.^((p-2)/2).*(-((-1)^p+p/2)*sin((p/2-1)*t)...
 +(p/2-1)*sin((p/2-3)*t))-A_coef(p+length(A_coef)/2)*p/2*r.^((p-2)/2).*(...
 -((-1)^p-p/2)*cos((p/2-1)*t)-(p/2-1)*cos((p/2-3)*t));
end
```

```
Np = (2*h/fsigma)*abs(sqrt((sx_syp/2).^2 + sxyp.^2));
Ndark = 0:1:20;
Nbright = 0.5:1:20.5;
Nmixed = 0:0.5:15;
figure;
[cs1,H] = contour(X,Y,Np,Ndark);
colormap jet
clabel(cs1,H,[0:1:5])
clabel(cs1,H,'FontSize',12,'Color','k')
grid
axis('equal')
hold on
plot(xx,yy,'rd','MarkerSize',3);
hold off
xlim([min(s) max(s)])
ylim([min(o) max(o)])
title('Plot of Dark field Fring order')
colorbar('location','Eastoutside')
path='mention path to save plot';
name2=['error_plot' num2str(n_start)];
name1=['mixed_frng_plot' num2str(n_start)];
fig1=figure;
set(gca,'FontSize',13)
contour(X,Y,Np,[0:0.5:15],'--')
colorbar('location','Eastoutside')
hold on
[cs1,H] = contour(X,Y,Np,Nmixed);
clabel(cs1,H,Nmixed,'FontSize',12,'LabelSpacing',200)
set(H,'LineWidth',1.35)
grid off
axis('equal')
hold on
plot(xx,yy,'rd','MarkerSize',3);
hold off
xlim([min(s) max(s)])
ylim([min(o) max(o)])

print(fig1,fullfile(path,name1),'-djpeg','-r300')
print(fig1,fullfile(path,name1),'-depsc','-r300')

fig2=figure;
```

```
contourf(crack_tip_x,crack_tip_y,error/max(max(error)),20)
colorbar('location','Eastoutside')
set(gca,'FontSize',13)
xlabel('x cordinates'), ylabel('y cordinates')

print(fig2,fullfile(path,name2),'-djpeg','-r300')
print(fig2,fullfile(path,name2),'-depsc','-r300')
```

- Fringe plotting form FEA solution

```
%%%%%%%%%%%program to plot fringe pattern from FEM results %%%%%%%%%%%%
%%% input data from FEM- xy coordinate data:'Inclusion_x.dat', ...
    connectivity data='Inclusion_icon.dat'
%%% and stress data= 'stress.xlsx'

%%% functions involved in code-shfunction : function file 'shfunction.m'
clc;
clear all;

path='mention path';

%%% reading saved FEM data: xy-coordinate data, connectivity data and ...
    stress data
xycoor=importdata([path '/Inclusion_x.dat']);
conn=importdata([path '/Inclusion_icon.dat']);
stress=xlsread([path '/stress.xlsx'],'A:C');


t=6;    %%% thickness of specimen
fsig=0.24;  %%% fringe value
nelnode=8;  % no of nodes per element written for 4 nodes per element
[p q]=size(xycoor);
[r s]=size(conn);

nnod=p;
nel=r;
%%fringe data computation at node
stress=stress(1:nel*8,:);
frnt=sortrows(stress,2);
ph=size(frnt);
```

```
k=1;
l=0;
avg=0;
stresst=zeros(p,2);
for i=2:1:ph(1)

    if frnt(i,2)==frnt(i−1,2)

        avg=avg+frnt(i−1,3);
        l=l+1;
    else
        avg=avg/l;
        stresst(k,:)=[frnt(i−1,2),avg];
        k=k+1;
        l=0;
        avg=0;
    end

end


frn(:,1)=stresst(:,1);
frn(:,2)=stresst(:,2)*t/fsig;

%%%% fixing image resolution
xresolution=2000;
yresolution=2000;

xmax=max(xycoor(:,1));
xmin=min(xycoor(:,1));
ymax=max(xycoor(:,2));
ymin=min(xycoor(:,2));

gpixx=round(xresolution/(xmax−xmin));

gpiyy=round(yresolution/(ymax−ymin));
tic;
m=1;
xx1=cell(nel,1);
yy1=cell(nel,1);
frn1=cell(nel,1);
```

```

for i=1:nel
    pcoor=zeros(nelnode,q);
    for j=1:nelnode

        pcoor(j,:)=xycoor(conn(i,j+1),:);

    end

    xext1=pcoor(1,1);           yext1=pcoor(1,2);
    xext2=pcoor(1,1);           yext2=pcoor(1,2);
    frng=zeros(nelnode,1);
    tx=zeros(nelnode,1);
    ty=zeros(nelnode,1);
    for ja=1:nelnode
        if pcoor(ja,1)>xext1
            xext1=pcoor(ja,1);
        end
        if pcoor(ja,1)<xext2
            xext2=pcoor(ja,1);
        end
        if pcoor(ja,2)>yext1
            yext1=pcoor(ja,2);
        end
        if pcoor(ja,2)<yext2
            yext2=pcoor(ja,2);
        end


        frng(ja)=frn(conn(i,ja+1),2);
        tx(ja)=xycoor(conn(i,ja+1),1);
        ty(ja)=xycoor(conn(i,ja+1),2);
    end

    if xext1<xmin
        continue;
    end

    if xext2>xmax
        continue;
    end

    if yext1<ymin
        continue;
    end
    if yext2>ymax
        continue;
    end


    xext=xext1-xext2; yext=yext1-yext2;

    npixx=round(gpixx*xext+1);
    npiyy=round(gpiyy*yext+1);
    ri=2/npixx;   si=2/npiyy;

    s=-1;

    xx11=zeros(npiyy*npixx,1);
    yy11=zeros(npiyy*npixx,1);
    frn11=zeros(npiyy*npixx,1);
    for ii=1:npiyy
      r=-1;
      for jj=1:npixx
       shf=shfunction(r,s);
        for iii=1:nelnode
         xx11((ii-1)*npixx+jj,1)=xx11((ii-1)*npixx+jj,1)+shf(iii)*tx(iii);
         yy11((ii-1)*npixx+jj,1)=yy11((ii-1)*npixx+jj,1)+shf(iii)*ty(iii);
         frn11((ii-1)*npixx+jj,1)=frn11((ii-1)*npixx+jj,1)+shf(iii)*frng(iii);
        end
        r=r+ri;
      end
      s=s+si;
    end
    xx1{i,1}=xx11;
    yy1{i,1}=yy11;
    frn1{i,1}=frn11;
end

toc;
x=cell2mat(xx1);
y=cell2mat(yy1);
```

```
z=cell2mat(frn1);
a=abs(z-round(z))≤0.01 ;
b=find(a);
plot(x(b),y(b),'.','MarkerSize', 5);
axis equal;

%%% plotting dark field fringe pattern
F = scatteredInterpolant(x,-y,z);

scale=5.5/240;
% fsigma=0.25;
% h=6;
xst=-2;
yst=-13;

 x1=xst:scale:768*scale+xst;
 y1=yst:scale:768/1.1*scale+yst;

[xx, yy]=meshgrid(x1,y1);
zz=F(xx,yy);
figure;

pp=zz-fix(zz);

I=255/2*(1-cos((pp)*2*pi));
imagesc(I),axis equal ,colormap gray
```

```
function [shf]=shfunction(r,s)
%%%% function required to in code for fringe plotting from FEM
    a1=1-r;a2=1+r;a3=1-s;a4=1+s;a5=(1-r^2)/2;a6=(1-s^2)/2;
    shf(1)=a1*a3*(-r-s-1)/4; shf(2)=a2*a3*(r-s-1)/4; shf(3)=a2*a4*(r+s-1)/4;
    shf(4)=a1*a4*(-r+s-1)/4; shf(5)=a5*a3;shf(6)=a6*a2; shf(7)=a5*a4; ...
        shf(8)=a6*a1;
end
```

– Strain intensity factor estimation for rigid line inclusion

```
%%% program to calculate the strain intensity factor for rigid line ...
    inclusion problem %%%%
```

```
%%%% input data file=python script file "full_model_rigid.py",
%%%% saved data files from abaqus (FEM) — data related to ...
    inclusion:'Inclusion_data.dat',
%%%% displacement data : 'Inclusion_disp.dat', stress data : ...
    'Inclusion_stress.dat',
%%%% xy coordinate data : 'Inclusion_x.dat', connectivity data : ...
    'Inclusion_icon.dat',
%%%% strain data : 'Inclusion_strain.dat' and epsilon_11 component data ...
    on line : 'Inclusion_dataE11.dat'.

%%%% functions required to run this files are
%%% shape functions for 8—nodded element: 'shapefunction.m'
%%% shape functions derivative for 8—nodded element: ...
    'shapefunctionderivative.m'
%%% singular stress field functions : 'get_f_and_g.m'
clc;
clear all;
close all;

fid = fopen('full_model_rigid_free.txt','r') ;
X = fread(fid) ;

fclose(fid) ;
xc=char(X.');

S1='w=100.00;';
S2='b=100.00;';
%wb=[100];
%bb=[30];
wb=[100 80 50 40 30 25 20 18 15 13];
bb=[100 80 50 40 30 25 20 18 15 13 10 5];

rH1=zeros(length(wb),length(bb));
rH2=zeros(length(wb),length(bb));
rHa1=zeros(length(wb),length(bb));
rstrn=zeros(length(wb),length(bb));
rwb=zeros(length(wb),length(bb));
ravstrn=zeros(length(wb),length(bb));

for ii=1:1:length(wb)
    for jj=1:1:length(bb)
    s1r=['w=' num2str(wb(ii)) '.00;'];
    s2r=['b=' num2str(bb(jj)) '.00;'];
    Yy = strrep(xc,S1 ,s1r) ;
    Y = strrep(Yy,S2 ,s2r) ;

    fid2 = fopen('full_model_rigid.py','wt') ;
    fwrite(fid2,Y) ;
    fclose (fid2) ;

    npoints=9;
    ncoord=2;
    nelnodes=8;
  %%% abaqus running with no GUI with input python script
    unix(['C:/SIMULIA/Abaqus/Commands/abaqus cae noGUI' ...
        '=D:/Prat_abaqus/Paper_res/full_model_rigid.py'])

    path='mention path';

    %%% reading data
    data=importdata([path '/Inclusion_data.dat']);
    disp=importdata([path '/Inclusion_disp.dat']);
    stress=importdata([path '/Inclusion_stress.dat']);
    xy=importdata([path '/Inclusion_x.dat']);
    connt=importdata([path '/Inclusion_icon.dat']);
    strain=importdata([path '/Inclusion_strain.dat']);
    E11=importdata([path '/Inclusion_dataE11.dat']);
    connt=sortrows(connt,1);

    l=data(1);                 %% Half inclusion length
    r1=data(2);                %% radius inside and outside for closed path
    r2=data(3);
    rot=data(4);               %% inclusion inclination wrt load  (angle)
    E=data(5);
    nu=data(6);
    pressure=data(7);          %% applied load
    tipnodno=data(8);
    intip=[data(9),data(10)];  %%  inclusion tip co-ordinates
    disptip=[disp(tipnodno,1);disp(tipnodno,2)];     %%  inclusion tip ...
        displacement
    wd=data(11);        % half width of specimen
    bd=data(12);        %half breadth of specimen
    dispb=data(13);     %applied disp
    kappa=3-4*nu;
    mu=E/(2*(nu+1));

    trn=[cos(rot) sin(rot);-sin(rot) cos(rot)];

    ab=sqrt(3/5);

xilist =[  -ab      0     ab     -ab     0    ab   -ab    0    ab;
           -ab    -ab    -ab       0     0     0    ab   ab    ab ];

%plot(xilist(1,:),xilist(2,:),'o')

% weight for corresponding gauss point
w=zeros(1,9);
w1D = [0.555555555,0.888888888,0.55555555555];
for j = 1:3
    for i = 1:3
        n = 3*(j-1)+i;
        w(n) = w1D(i)*w1D(j);
    end
end

p=size(stress);

nelmn=p(1)/9; % no of element inside the contour


ceH1=0;
caH1=0;

ceH2=0;
caH2=0;
aSE=0;
Area1=0;
m=1;

for ielmn=1:nelmn
    ceH1=0;
    ceH2=0;
    eSE=0;
```

```
   s=(ielmn-1)*npoints+1;
   elmnno=stress(s,1);
   nconnect=connt(elmnno,2:9);
   coord=zeros(ncoord,nelnodes);
   u=zeros(ncoord,nelnodes);

if nconnect(1,1)==0
       continue
end

for i=1:nelnodes
 coord(:,i)=trn*[xy(nconnect(i),1)-intip(1);xy(nconnect(i),2)-intip(2)];

 u(:,i)=[disp(nconnect(i),1)-disptip(1);disp(nconnect(i),2)-disptip(2)];
end

   rsq=(coord(1,:)).*(coord(1,:))+(coord(2,:)).*(coord(2,:));
   r=sqrt(rsq);
   q=(r-r1)/(r2-r1);

   for intpt = 1:npoints

       % Compute shape functions && derivatives wrt local coords  %
       xi=zeros(1,ncoord);
       for i = 1:ncoord
           xi(i) = xilist(i,intpt);
       end
       N = shapefunction(xi);
       dNdxi = shapefunctionderivative(xi);

       %     Compute the jacobian matrix && its determinant

       dxdxi=zeros(ncoord,ncoord);
       for i = 1:ncoord
           for j = 1:ncoord
               dxdxi(i,j) = 0.;
               for a = 1:nelnodes
                   dxdxi(i,j) = dxdxi(i,j) + coord(i,a)*dNdxi(a,j);
               end
           end
       end
```

```
        dxidx = inv(dxdxi);
        dt = det(dxdxi);

        %%Convert shape function derivatives:derivatives wrt global coords

        dNdx=zeros(nelnodes,ncoord);

        for a = 1:nelnodes
            for i = 1:ncoord
                dNdx(a,i) = 0.;
                for j = 1:ncoord
                    dNdx(a,i) = dNdx(a,i) + dNdxi(a,j)*dxidx(j,i);
                end
            end
        end

        c=(ielmn-1)*npoints+intpt;
        Q=q*dNdx;
        norm(Q);
        xyg=coord*N;

        u;
        ugo=u*N;
        ug=ugo;
        sigmag=[stress(c,3) stress(c,5); stress(c,5) stress(c,4)]; %fea ...
            value
        rg=sqrt(xyg(1)*xyg(1)+(xyg(2))*(xyg(2)));
        theta=atan2(xyg(2),xyg(1));

        %%%% singular values from function
        lambda=1/2;
        [sigmagh1, ugh1,sigmagh2, ugh2] = get_f_and_g(theta, lambda, ...
            kappa, rg, mu);

        lambda=-1/2;
        [sigmastarh1, ustarh1,sigmastarh2, ustarh2] = get_f_and_g(theta, ...
            lambda, kappa, rg, mu);

        sigmag1=trn*sigmag*trn';
        ug1=trn*ug;

        %%%% area integral for symmetric part %%%%
        sigmastar=sigmastarh1+sigmastarh2*0;
        ustar=ustarh1+ustarh2*0;
        nH1=w(intpt)*(sigmag1*ustar-sigmastar*ug1)'*Q'*dt;
        ceH1=ceH1+ nH1;

        %%%% area integral for unsymmetric part %%%%
        sigmastar=sigmastarh1*0+sigmastarh2;
        ustar=ustarh1*0+ustarh2;
        nH2=w(intpt)*(sigmag1*ustar-sigmastar*ug1)'*Q'*dt;
        ceH2=ceH2+ nH2;

        sigma11=stress(c,3);sigma22= stress(c,4); sigma12=stress(c,5);
        strn11=strain(c,3);strn22= strain(c,4); strn12=strain(c,5);
        SE=1/2*(sigma11*strn11+sigma22*strn22+sigma12*strn12);
        nSE=w(intpt)*SE*dt;
        eSE=eSE+ nSE;
        Area1 = Area1 + w(intpt)*dt;
    end
    caH1=caH1+ceH1;
    caH2=caH2+ceH2;
    aSE=aSE+eSE;
end
    rot*180/pi;
    FeH1=caH1/(4*pi*(kappa^2+kappa)*mu);
    FeH2=caH2/(4*pi*(kappa^2+kappa)*mu);
    %AnlH1=pressure/4*(kappa-(1-2*cos(2*rot)))/kappa*sqrt(1/2)
    format long
    nu;
    avstrn=mean(E11,1);
    AnlHe1a=avstrn(2)*sqrt(1/2)  %%% analytical strain intensity factor
    He1=FeH1*kappa               %%% Numerical strain intensity factor
    He2=FeH2*kappa;
    He1/AnlHe1a
    strain=dispb/(2*wd);
    AnlHe1=strain*sqrt(1/2)
    He1/AnlHe1
    rH1(ii,jj)=He1;
    rH2(ii,jj)=He2;
    rHa1(ii,jj)=AnlHe1;
    rstrn(ii,jj)=strain;
    ravstrn(ii,jj)=avstrn(2);
    wd
    bd
    end
end

save('result_free_rHa1.txt','rHa1')
save('result_free_rstrn.txt','rstrn')
save('result_free_ravstrn.txt','ravstrn')
save('result_free_rH1.txt','rH1')
```

```

function [sigmacarth1, ucarth1,sigmacarth2, ucarth2] = ...
    get_f_and_g(theta, lambda, kappa, rg, mu)
%%%% Singular elastic stress and displacement field equations for rigid ...
    line inclusion problem
%%% This function required for Strain intensity factor estimation for ...
    rigid line inclusion program
%%%% save this function as 'get_f_and_g.m'

         %%%%symmetric part
    fs11=-(-(kappa+lambda)*cos((lambda+1)*theta)+(lambda-3)...
                *cos((lambda-1)*theta));
    fs12=(-(kappa+lambda)*sin((lambda+1)*theta)+(lambda-1)...
                *sin((lambda-1)*theta));
    fs22=(-(kappa+lambda)*cos((lambda+1)*theta)+(lambda+1)...
                *cos((lambda-1)*theta));

    gs1=((kappa-lambda)*cos((lambda-1)*theta)+(kappa+lambda)...
                *cos((lambda+1)*theta));
    gs2=((kappa+lambda)*sin((lambda-1)*theta)-(kappa+lambda)...
                *sin((lambda+1)*theta));
    sigmapol=2*mu*rg^(lambda-1)*lambda*[fs11 fs12;fs12 fs22];
    upol=rg^lambda*[gs1;gs2];

    poltocart=[cos(theta) -sin(theta); sin(theta) cos(theta)];
    sigmacarth1=poltocart*sigmapol*poltocart';
    ucarth1=poltocart*upol;

    %%%% anti-symmetric part
    fu11=-((kappa-lambda)*sin((lambda+1)*theta)+(lambda-3)...
                *sin((lambda-1)*theta));
    fu12=-((kappa-lambda)*cos((lambda+1)*theta)+(lambda-1)...
                *cos((lambda-1)*theta));
    fu22=((kappa-lambda)*sin((lambda+1)*theta)+(lambda+1)...
                *sin((lambda-1)*theta));

    gu1=((kappa-lambda)*sin((lambda-1)*theta)-(kappa-lambda)...
                *sin((lambda+1)*theta));
    gu2=-((kappa+lambda)*cos((lambda-1)*theta)+(kappa-lambda)...
                *cos((lambda+1)*theta));
    sigmapol=2*mu*rg^(lambda-1)*lambda*[fu11 fu12;fu12 fu22];
    upol=rg^lambda*[gu1;gu2];

    sigmacarth2=poltocart*sigmapol*poltocart';
    ucarth2=poltocart*upol;
return
end
```

```
function N =shapefunction(xi)
%shape function for quad element%
%%%%% Shape functions for 8-nodded element
%%% This function required for Strain intensity factor estimation for ...
    rigid line inclusion program
%%%% save this function as 'shapefunction.m'
        N=zeros(8,1);
       N(1)  = -0.25*(1.-xi(1))*(1.-xi(2))*(1.+xi(1)+xi(2));
       N(2)  = 0.25*(1.+xi(1))*(1.-xi(2))*(xi(1)-xi(2)-1.);
       N(3)  = 0.25*(1.+xi(1))*(1.+xi(2))*(xi(1)+xi(2)-1.);
       N(4)  = 0.25*(1.-xi(1))*(1.+xi(2))*(xi(2)-xi(1)-1.);
       N(5)  = 0.5*(1.-xi(1)*xi(1))*(1.-xi(2));
       N(6)  = 0.5*(1.+xi(1))*(1.-xi(2)*xi(2));
       N(7)  = 0.5*(1.-xi(1)*xi(1))*(1.+xi(2));
       N(8)  = 0.5*(1.-xi(1))*(1.-xi(2)*xi(2));
end
```

```
function dNdxi=shapefunctionderivative(xi)
```

```
%%%%% Shape functions derivatives for 8—nodded element
%%% This function required for Strain intensity factor estimation for ...
    rigid line inclusion program
%%%% save this function as 'shapefunctionderivative.m'
       dNdxi(1,1)  = 0.25*(1.—xi(2))*(2.*xi(1)+xi(2));
       dNdxi(1,2)  = 0.25*(1.—xi(1))*(xi(1)+2.*xi(2));
       dNdxi(2,1)  = 0.25*(1.—xi(2))*(2.*xi(1)—xi(2));
       dNdxi(2,2)  = 0.25*(1.+xi(1))*(2.*xi(2)—xi(1));
       dNdxi(3,1)  = 0.25*(1.+xi(2))*(2.*xi(1)+xi(2));
       dNdxi(3,2)  = 0.25*(1.+xi(1))*(2.*xi(2)+xi(1));
       dNdxi(4,1)  = 0.25*(1.+xi(2))*(2.*xi(1)—xi(2));
       dNdxi(4,2)  = 0.25*(1.—xi(1))*(2.*xi(2)—xi(1));
       dNdxi(5,1)  = —xi(1)*(1.—xi(2));
       dNdxi(5,2)  = —0.5*(1.—xi(1)*xi(1));
       dNdxi(6,1)  = 0.5*(1.—xi(2)*xi(2));
       dNdxi(6,2)  = —(1.+xi(1))*xi(2);
       dNdxi(7,1)  = —xi(1)*(1.+xi(2));
       dNdxi(7,2)  = 0.5*(1.—xi(1)*xi(1));
       dNdxi(8,1)  = —0.5*(1.—xi(2)*xi(2));
       dNdxi(8,2)  = —(1.—xi(1))*xi(2);
end
```

Python Script Abaqus modeling of rigid line inclusion

```
### abaqus  python script for problem of straight rigid line inclusion###

import numpy as np
import numpy
import os
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *
from abaqusConstants import*
from math import*
import sys
from abaqus import *
from odbAccess import *
import xyPlot
import displayGroupOdbToolset as dgo
# os.getcwd()
os.chdir("mention path")
Mdb()
numpy.set_printoptions(threshold='nan')

w=13.00;  ### half width of specimen
b=5.00;   ### half height of specimen
l=10.00;  ### inclusion length
p=0.00;   ### inclusion centre (p,l)

r1=0.005;
r2=1.0;    ### Circular Mesh outside radius at inclusion
c=2.00;     ### Half Square edge at inclusion
m1=0.25;    ###mesh size of square
m2=0.1*w;   #### max Mesh size in x-direction
m3=0.1*b;     ### max Mesh size in y-direction
m4=0.1;
m5=0.005        ### Mesh size at r1
m6=0.005   ### Mesh size at tip
#ka=0
pressure=0.075;
dispbv=1;
alpha=0 *pi/180;
E=7.98   ### Material Properties
nu=0.45
##### sketch part
myModel = mdb.Model(name='Model A')
mySketch = myModel.ConstrainedSketch(name='Sketch A', sheetSize=200.0)
session.journalOptions.setValues(replayGeometry=COORDINATE,recoverGeometry=COORDINATE)
xyCoords = ((-w,-b),(w,-b),(w,b),(-w,b),(-w,-b))
for i in ...
    range(len(xyCoords)-1):mySketch.Line(point1=xyCoords[i],point2=xyCoords[i+1])
```

```
myPart = myModel.Part(name='Part A', ...
     dimensionality=TWO_D_PLANAR,type=DEFORMABLE_BODY)
myPart.BaseShell(sketch=mySketch)
del mdb.models['Model-1']

### material definition
mdb.models['Model A'].Material(name='Matrix')
mdb.models['Model A'].materials['Matrix'].Elastic(table=((E, nu), ))
mdb.models['Model A'].HomogeneousSolidSection(material='Matrix', ...
     name='matrixSection', thickness=None)
## section definition
p1 = mdb.models['Model A'].parts['Part A']
p1.Set(faces=p1.faces.findAt(((w/2, b/2, 0.0), )), name='Set-1')
p1.SectionAssignment(offset=0.0, offsetField='', offsetType=MIDDLE_SURFACE, ...
     region=
    mdb.models['Model A'].parts['Part A'].sets['Set-1'], sectionName=
     'matrixSection', thicknessAssignment=FROM_SECTION)
### assembly definition
mdb.models['Model A'].rootAssembly.DatumCsysByDefault(CARTESIAN)
mdb.models['Model A'].rootAssembly.Instance(dependent=ON, name='Part A-1',
    part=mdb.models['Model A'].parts['Part A'])

### partitioning sketch for meshing
myModel.ConstrainedSketch(name='PartitionSketch', sheetSize=200)
p2=mdb.models['Model A'].sketches['PartitionSketch']
p2.Line(point1=(-1.01*w,(p-c)), point2=(w*1.01, (p-c)))
p2.Line(point1=(-(l+c),p), point2=((l+c),p))
p2.Line(point1=(-1.01*w,(p+c)), point2=(w*1.01, (p+c)))
p2.Line(point1=(l-c,-1.01*b), point2=(l-c,1.01*b))
p2.Line(point1=(l+c,-1.01*b), point2=(l+c,1.01*b))
p2.Line(point1=(-(l-c),-1.01*b), point2=(-(l-c),1.01*b))
p2.Line(point1=(-(l+c),-1.01*b), point2=(-(l+c),1.01*b))
p2.Line(point1=(l,p), point2=(l+c,p+c))
p2.Line(point1=(l, p), point2=(l-c,p+c))
p2.Line(point1=(l, p), point2=(l+c,p-c))
p2.Line(point1=(l, p), point2=(l-c,p-c))
p2.Line(point1=(-l,p), point2=(-(l+c),p+c))
p2.Line(point1=(-l, p), point2=(-(l-c),p+c))
p2.Line(point1=(-l, p), point2=(-(l+c),p-c))
p2.Line(point1=(-l, p), point2=(-(l-c),p-c))
p2.CircleByCenterPerimeter(center=(l,p), point1=(l+r1,p))
```

```
p2.CircleByCenterPerimeter(center=(l,p), point1=(l+r2,p))
p2.CircleByCenterPerimeter(center=(-l,p), point1=(-(l+r1),p))
p2.CircleByCenterPerimeter(center=(-l,p), point1=(-(l+r2),p))
p1.PartitionFaceBySketch(faces=p1.faces.findAt(((w/2, b/2, 0.0), )), sketch=p2)

##### Seeding part
p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end1Edges=
    p1.edges.findAt(((0.99*w,-b , 0.0), ),((0.99*w, p-c,
    0.0), ),((0.99*w, p+c, 0.0), ),((-0.99*w, b, 0.0), ...
        )), end2Edges=p1.edges.findAt( ((0.99*w, b, 0.0), ),((-0.99*w,-b , ...
        0.0), ),((-0.99*w, p-c, 0.0), ),
    ((-0.99*w, p+c, 0.0), )), maxSize=m2, minSize=m1)

p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end2Edges=
    p1.edges.findAt(((l-c, -0.99*b, 0.0), ), ((l+c, -0.99*b, 0.0), ),((w, ...
        -0.99*b, 0.0), ),
    ((-(l-c), -0.99*b, 0.0), ), ((-(l+c), -0.99*b, 0.0), ),((-w, 0.99*b, ...
        0.0), ) ),
    end1Edges=p1.edges.findAt( ((l-c, 0.99*b, 0.0), ), ((l+c, 0.99*b, 0.0), ...
        ),((w, 0.99*b, 0.0), ),
    ((-(l-c), 0.99*b, 0.0), ), ((-(l+c), 0.99*b, 0.0), ),((-w, -0.99*b, 0.0), ))
    , maxSize=m3, minSize=m1)

p1.seedEdgeBySize(constraint=FINER,deviationFactor=0.1, edges=
    p1.edges.findAt( ((l-c, (p-c*0.99), 0.0), ), ((l+c, (p-c*0.99), 0.0), ...
        ),((l+c, (p+c*0.99), 0.0), ), ((w, (p-c*.099), 0.0), ),
    ((l-c, (p+c*0.99), 0.0), ), ((l, (p+c), 0.0), ), ((l, (p-c), 0.0), ),((l, ...
        -b, 0.0), ), ((l, b, 0.0), ) ,
    ((-(l-c), (p-c*0.99), 0.0), ), ((-(l+c), (p-c*0.99), 0.0), ),((-(l+c), ...
        (p+c*0.99), 0.0), ), ((-w, (p-c*.099), 0.0), ),
    ((-(l-c), (p+c*0.99), 0.0), ), ((-l, (p+c), 0.0), ), ((-l, (p-c), 0.0), ...
        ),((-l, -b, 0.0), ), ((-l, b, 0.0), ) ),
    size=m1)

p1.seedEdgeByBias(biasMethod=DOUBLE,constraint=FINER, endEdges=
    p1.edges.findAt(((0.01*w, -b, 0.0), ), ((0.01*w, (p-c), 0.0), ),((0.01*w, ...
        p, 0.0), ), ((0.01*w, p+c, 0.0), ),((0.01*w,b, 0.0), ) ),
     maxSize=m1*2, minSize=m1)

p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end2Edges=
    p1.edges.findAt((((l-c*0.99), p, 0.0), ),(((l+r2), (p+r2), 0.0), ...
        ),(((l+r2), (p-r2), 0.0), )),
    end1Edges=p1.edges.findAt((((l-r2), (p+r2), 0.0), ),(((l+c*0.99), p, ...
        0.0), ),(((l-r2), (p-r2), 0.0), ),)
    , maxSize=m1, minSize=m4)

p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end2Edges=
    p1.edges.findAt(((-(l+c*0.99), p, 0.0), ),((-(l+r2), (p+r2), 0.0), ...
        ),((-(l+r2), (p-r2), 0.0), )),
    end1Edges=p1.edges.findAt(((-(l-r2), (p+r2), 0.0), ),((-(l-c*0.99), p, ...
        0.0), ),((-(l-r2), (p-r2), 0.0), ),)
    , maxSize=m1, minSize=m4)

p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end2Edges=
    p1.edges.findAt((((l-r2*0.99), p, 0.0), ),(((l+r1), (p+r1), 0.0), ...
        ),(((l+r1), (p-r1), 0.0), )),
    end1Edges=p1.edges.findAt((((l-r1), (p+r1), 0.0), ),(((l+r2*0.99), p, ...
        0.0), ),(((l-r1), (p-r1), 0.0), ),)
    , maxSize=m4, minSize=m5)

p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end2Edges=
    p1.edges.findAt(((-(l+r2*0.99), p, 0.0), ),((-(l+r1), (p+r1), 0.0), ...
        ),((-(l+r1), (p-r1), 0.0), )),
    end1Edges=p1.edges.findAt(((-(l-r1), (p+r1), 0.0), ),((-(l-r2*0.99), p, ...
        0.0), ),((-(l-r1), (p-r1), 0.0), ),)
    , maxSize=m4, minSize=m5)

p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end2Edges=
    p1.edges.findAt((((l-r1*0.99), p, 0.0), ),(((l+r1/2), (p+r1/2), 0.0), ...
        ),(((l+r1/2), (p-r1/2), 0.0), )),
    end1Edges=p1.edges.findAt((((l-r1/2), (p+r1/2), 0.0), ),(((l+r1*0.1), p, ...
        0.0), ),(((l-r1/2), (p-r1/2), 0.0), ),)
    , maxSize=m5, minSize=m6)

p1.seedEdgeByBias(biasMethod=SINGLE,constraint=FINER, end2Edges=
    p1.edges.findAt(((-(l+r1*0.99), p, 0.0), ),((-(l+r1/2), (p+r1/2), 0.0), ...
        ),((-(l+r1/2), (p-r1/2), 0.0), )),
    end1Edges=p1.edges.findAt(((-(l-r1/2), (p+r1/2), 0.0), ),((-(l-r1*0.1), ...
        p, 0.0), ),((-(l-r1/2), (p-r1/2), 0.0), ),)
    , maxSize=m5, minSize=m6)

```

```
mdb.models['Model A'].rootAssembly.Instance(dependent=OFF, name='Part A-1',
    part=mdb.models['Model A'].parts['Part A'])

p3=mdb.models['Model A'].rootAssembly.regenerate()
p3=mdb.models['Model A'].rootAssembly
### quarter point definition at both tip of rigid line inclusion
p3.engineeringFeatures.ContourIntegral(
    collapsedElementAtTip=SINGLE_NODE, crackFront=Region(
    edges=p3.instances['Part A-1'].edges.findAt((((l-c*1.01),p, 0.0), ), ...
        (((l-c*0.99),p , 0.0), ), (((l-r2*0.99), p, 0.0), ), (((l-r1*0.99),p, ...
        0.0), ),
    ((-(l-c*0.99),p , 0.0), ), ((-(l-r2*0.99), p, 0.0), ), ((-(l-r1*0.99),p, ...
        0.0), ))),
    crackTip=Region(vertices=p3.instances['Part A-1'].vertices.findAt(
    ((l, p, 0.0), ), )), extensionDirectionMethod=Q_VECTORS,
    midNodePosition=0.25, name='Crack-1', qVectors=((
    p3.instances['Part A-1'].vertices.findAt((-l, p, 0.0), ),
    p3.instances['Part A-1'].vertices.findAt((l, p, 0.0), )), ), symmetric=OFF)

p3.engineeringFeatures.ContourIntegral(
    collapsedElementAtTip=SINGLE_NODE, crackFront=Region(
    edges=p3.instances['Part A-1'].edges.findAt((((l-c*1.01),p, 0.0), ), ...
        (((l-c*0.99),p , 0.0), ), (((l-r2*0.99), p, 0.0), ), (((l-r1*0.99),p, ...
        0.0), ),
    ((-(l-c*0.99),p , 0.0), ), ((-(l-r2*0.99), p, 0.0), ), ((-(l-r1*0.99),p, ...
        0.0), ))),
    crackTip=Region(vertices=p3.instances['Part A-1'].vertices.findAt(
    ((-l, p, 0.0), ), )), extensionDirectionMethod=Q_VECTORS,
    midNodePosition=0.25, name='Crack-2', qVectors=((
    p3.instances['Part A-1'].vertices.findAt((-l, p, 0.0), ),
    p3.instances['Part A-1'].vertices.findAt((l, p, 0.0), )), ), symmetric=OFF)

    #### setting mesh parameters
mdb.models['Model A'].rootAssembly.setMeshControls(elemShape=QUAD, regions=
    mdb.models['Model A'].rootAssembly.instances['Part A-1'].faces.findAt(((
    0, 0.99*b, 0.0), ), ((l, 0.99*b, 0.0), ), ((0.99*w, 0.99*b, 0.0), ), ((0, ...
        -0.99*b, 0.0), ),
    ((l, -0.99*b, 0.0), ), ((0.99*w, -0.99*b, 0.0), ),((-l, 0.99*b, 0.0), ), ...
        ((-0.99*w,0.99*b, 0.0), ), ((-l, -0.99*b, 0.0), ),((-0.99*w,-0.99*b, ...
        0.0), ),
    ((-0.99*w, 0, 0.0), ), ((0.99*w,0, 0.0), ), ((0, 0.99*c, 0.0), ...
        ),((0,-0.99*c, 0.0), )),
    technique=STRUCTURED)

mdb.models['Model A'].rootAssembly.setMeshControls(regions=
   mdb.models['Model A'].rootAssembly.instances['Part A-1'].faces.
   getByBoundingBox( l-1.01*c,p-1.01*c, -100, l+1.01*c ,p+1.01*c, 100), ...
       technique=SWEEP)

mdb.models['Model A'].rootAssembly.setMeshControls(regions=
   mdb.models['Model A'].rootAssembly.instances['Part A-1'].faces.
   getByBoundingBox( -(l+1.01*c),p-1.01*c, -100, -(l-1.01*c) ,p+1.01*c, 100), ...
       technique=SWEEP)

mdb.models['Model A'].rootAssembly.Set(faces=
    mdb.models['Model A'].rootAssembly.instances['Part A-1'].faces.
    findAt(((l+0.99*r2*cos(7*pi/8),p+r2*0.99*sin(7*pi/8), 0.0), ...
        ),((l+r2*0.99*r2*cos(-7*pi/8),p-r2*0.99*sin(7*pi/8), 0.0), ), (( ...
        l,p+r2*.99, 0.0), ),
    (( l,p-r2*0.99, 0.0), ), ((l+r2*0.99*cos(pi/8),p+r2*0.99*sin(pi/8), 0.0), ...
        ),((l+r2*0.99*cos(-pi/8),p+r2*0.99*sin(-pi/8), 0.0), ), ), name='Area')
   #### meshing part
mdb.models['Model A'].rootAssembly.generateMesh(regions=(
    mdb.models['Model A'].rootAssembly.instances['Part A-1'], ))

mdb.models['Model A'].rootAssembly.setElementType(elemTypes=(ElemType(
    elemCode=CPE8H, elemLibrary=STANDARD), ElemType(elemCode=CPE6H,
    elemLibrary=STANDARD)), regions=(
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].faces.getByBoundingBox(-1.01*w, -1.01*b, -100,   w*1.01, ...
        b*1.01, 100), ))

mdb.models['Model A'].StaticStep(name='Step-1', previous='Initial')

∆ = 1.0e-4
mdb.models['Model A'].rootAssembly.Set(name='Node-A', nodes=(
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].nodes.getByBoundingSphere(center=((l,p,0)),radius=∆),))

mdb.models['Model A'].rootAssembly.Set(name='Node-B', nodes=(
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].nodes.getByBoundingSphere(center=((-l,p,0)),radius=Δ),))

mdb.models['Model A'].rootAssembly.Set(edges=(mdb.models['Model ...
    A'].rootAssembly.instances['Part ...
    A-1'].edges.getByBoundingBox(-l-Δ,p-Δ,-10,l+Δ, p+Δ,10), ), name='Inclusion');

xya=mdb.models['Model A'].rootAssembly.sets['Node-A'].nodes[0].coordinates
xyb=mdb.models['Model A'].rootAssembly.sets['Node-B'].nodes[0].coordinates
la=mdb.models['Model A'].rootAssembly.sets['Node-A'].nodes[0].label

# s=abs(xya[0]-xyb[0])
# t=abs(xya[1]-xyb[1])
# mdb.models['Model A'].Equation(name='Constraint-3', terms=((-s, 'Node-A', ...
    1), (s, 'Node-B', 1)))
# st=sqrt((xya[0]-xyb[0])*(xya[0]-xyb[0])+(xya[1]-xyb[1])*(xya[1]-xyb[1]))
# lb=mdb.models['Model A'].rootAssembly.sets['Node-B'].nodes[0].label
# incdata=mdb.models['Model A'].rootAssembly.sets['Inclusion'].nodes
# nond=len(incdata)
# for i in range(0,nond):
    # acd=mdb.models['Model A'].rootAssembly.sets['Inclusion'].nodes[i].label
    # if(acd==la or acd==lb):
        # print(i)
        # continue
    # else:
        # Nodename='Node-i'+str(i)
        # mdb.models['Model ...
            A'].rootAssembly.SetFromNodeLabels(name=Nodename,nodeLabels=(('Part ...
            A-1', (acd,)),))
        # xyi=mdb.models['Model ...
            A'].rootAssembly.sets[Nodename].nodes[0].coordinates
        # s=sqrt((xya[0]-xyi[0])*(xya[0]-xyi[0])+(xya[1]-xyi[1])*(xya[1]-xyi[1]))
        # t=st-s
        # consrteq='Constr-inc-'+str(i)
        # mdb.models['Model A'].Equation(name=consrteq +'-2', terms=((1.0, ...
            Nodename, 2),(-s/st, 'Node-B', 2), (-t/st, 'Node-A', 2)))
        # mdb.models['Model A'].Equation(name=consrteq +'-1', terms=((1.0, ...
            Nodename, 1),(-s/st, 'Node-B', 1), (-t/st, 'Node-A', 1)))

RP1=mdb.models['Model A'].rootAssembly.ReferencePoint(point=(0.0, 0.0, 0.0))

```

```
mdb.models['Model A'].RigidBody(name='Constraint-1', pinRegion=
    mdb.models['Model A'].rootAssembly.sets['Inclusion'], refPointAtCOM=ON, ...
        refPointRegion=Region(referencePoints=(mdb.models['Model ...
        A'].rootAssembly.referencePoints[RP1.id], )))

mdb.models['Model A'].rootAssembly.Set(edges=
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].edges.findAt(((-w,-0.99*b, 0.0), ), ((-w, p, 0.0), ),((-w, ...
        b*0.99, 0.0), ) ), name='Set-6')

mdb.models['Model A'].rootAssembly.Set(edges=
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].edges.findAt(((w,-0.99*b, 0.0), ), ((w, p, 0.0), ),((w, b*0.99, ...
        0.0), ) ), name='Set-7')

mdb.models['Model A'].rootAssembly.Set(edges=
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].edges.findAt(((-w*0.99,-b, 0.0), ), ((-l, -b, 0.0), ),((0,-b, ...
        0.0), ) ,((l, -b, 0.0), ),((w*0.99,-b, 0.0), )) , name='Set-10')

mdb.models['Model A'].rootAssembly.Set(edges=
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].edges.findAt(((-w*0.99,b, 0.0), ), ((-l, b, 0.0), ),((0,b, ...
        0.0), ) ,((l, b, 0.0), ),((w*0.99,b, 0.0), )) , name='Set-11')

mdb.models['Model A'].DisplacementBC(amplitude=UNSET, createStepName='Step-1',
    distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=None, name=
    'BC-1', region=mdb.models['Model A'].rootAssembly.sets['Set-6'], u2=UNSET,
    u1=0.0, ur3=UNSET)

mdb.models['Model A'].rootAssembly.Set(name='Set-206', vertices=
    mdb.models['Model A'].rootAssembly.instances['Part ...
        A-1'].vertices.findAt(((-w, -b, 0.0), )))

#mdb.models['Model A'].DisplacementBC(amplitude=UNSET, createStepName='Step-1',
    #distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=None, name=
    #'BC-2', region=mdb.models['Model A'].rootAssembly.sets['Set-206'], u1=UNSET
    #, u2=0.0, ur3=UNSET)

mdb.models['Model A'].DisplacementBC(amplitude=UNSET, createStepName='Step-1',
    distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=None, name=
    'BC-2', region=mdb.models['Model A'].rootAssembly.sets['Set-7'], u1=dispbv
    , u2=UNSET, ur3=UNSET)

mdb.models['Model A'].DisplacementBC(amplitude=UNSET, createStepName='Step-1',
    distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=None, name=
    'BC-3', region=mdb.models['Model A'].rootAssembly.sets['Set-206'], u1=UNSET
    , u2=0.0, ur3=UNSET)

#mdb.models['Model A'].DisplacementBC(amplitude=UNSET, createStepName='Step-1',
    #distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=None, name=
    #'BC-4', region=mdb.models['Model A'].rootAssembly.sets['Set-11'], u1=UNSET
    #, u2=-dispbv, ur3=UNSET)

#mdb.models['Model A'].DisplacementBC(amplitude=UNSET, createStepName='Step-1',
     #distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=None, name=
     #'BC-5', region=mdb.models['Model A'].rootAssembly.sets['Set-10'], u2=0.0,
     #u1=UNSET, ur3=UNSET)

mdb.models['Model A'].rootAssembly.Surface(name='Surf-1', ...
    side1Edges=mdb.models['Model A'].rootAssembly.sets['Set-7'].edges)

#mdb.models['Model A'].Pressure(amplitude=UNSET, createStepName='Step-1',
    #distributionType=UNIFORM, field='', magnitude=-0.075, name='Load-1', region=
    #mdb.models['Model A'].rootAssembly.surfaces['Surf-1'])

mdb.models['Model A'].rootAssembly.Surface(name='Surf-2', ...
    side1Edges=mdb.models['Model A'].rootAssembly.sets['Set-11'].edges)

#mdb.models['Model A'].Pressure(amplitude=UNSET, createStepName='Step-1',
    #distributionType=UNIFORM, field='', magnitude=-pressure, name='Load-2', ...
        region=
    #mdb.models['Model A'].rootAssembly.surfaces['Surf-1'])

mdb.Job(atTime=None, contactPrint=OFF, description='', echoPrint=OFF,
    explicitPrecision=SINGLE, getMemoryFromAnalysis=True, historyPrint=OFF,
    memory=90, memoryUnits=PERCENTAGE, model='Model A', modelPrint=OFF,
    multiprocessingMode=DEFAULT, name='Inclusion', nodalOutputPrecision=SINGLE,
    numCpus=1, numGPUs=0, queue=None, scratch='', type=ANALYSIS,
    userSubroutine='', waitHours=0, waitMinutes=0)

mdb.jobs['Inclusion'].submit(consistencyChecking=OFF)
mdb.jobs['Inclusion'].waitForCompletion()
#       ################### POST PROCESSING #########

jname = 'Inclusion'
odbname= jname+'.odb'
odb = openOdb (odbname)
#odb = session.openOdb(name=odbname)
assembly   = odb.rootAssembly
nodes=odb.rootAssembly.instances['Part A-1'].nodes
elements=odb.rootAssembly.instances['Part A-1'].elements
nelm = len(elements)
nnod = len(nodes)
session.viewports['Viewport: 1'].setValues(displayedObject=odb)
pathp=l+c+10
path = session.Path(name='path', type=POINT_LIST,
    expression=((l,p, 0.0), (pathp, p, 0.0)))

data = session.XYDataFromPath(name='data', path=path,
    includeIntersections=True, shape=UNDEFORMED, labelType=TRUE_DISTANCE,
    variable=('S',INTEGRATION_POINT,((COMPONENT,'S11'),),))
dataS11 = numpy.zeros(shape=(0,2))
temp = numpy.zeros(shape=(2))

for value in range(0,len(data)):
    temp[0] =  data[value][0]
    temp[1] =  data[value][1]
    dataS11=numpy.vstack((dataS11,temp))

data = session.XYDataFromPath(name='data',
     path=path,
     includeIntersections=True,
     shape=UNDEFORMED,
     labelType=TRUE_DISTANCE,
     variable= ('S',INTEGRATION_POINT, ((COMPONENT, 'S22' ), )),)

dataS22 = numpy.zeros(shape=(0,2))
temp = numpy.zeros(shape=(2))

for value in range(0,len(data)):
    temp[0] =  data[value][0]
    temp[1] =  data[value][1]
```

```python
    dataS22=numpy.vstack((dataS22,temp))

data = session.XYDataFromPath(name='data',
     path=path,
     includeIntersections=True,
     shape=UNDEFORMED,
     labelType=TRUE_DISTANCE,
     variable=('S',INTEGRATION_POINT,((COMPONENT,'S12'),),))

dataS12 = numpy.zeros(shape=(0,2))
temp = numpy.zeros(shape=(2))

for value in range(0,len(data)):
    temp[0] =  data[value][0]
    temp[1] =  data[value][1]
    dataS12=numpy.vstack((dataS12,temp))

numpy.savetxt(jname+'_dataS11.dat', dataS11, fmt='%15.7e %15.7e ')
numpy.savetxt(jname+'_dataS22.dat', dataS22, fmt='%15.7e %15.7e ')
numpy.savetxt(jname+'_dataS12.dat', dataS12, fmt='%15.7e %15.7e ')

path1 = session.Path(name='path1', type=POINT_LIST,expression=((w,b, 0.0)
    , (w, -b, 0.0)))
data = session.XYDataFromPath(name='data', path=path1,
    includeIntersections=True, shape=UNDEFORMED, labelType=TRUE_DISTANCE,
    variable=('E',INTEGRATION_POINT,((COMPONENT,'E11'),),))
dataE11 = numpy.zeros(shape=(0,2))
temp = numpy.zeros(shape=(2))

for value in range(0,len(data)):
    temp[0] =  data[value][0]
    temp[1] =  data[value][1]
    dataE11=numpy.vstack((dataE11,temp))

numpy.savetxt(jname+'_dataE11.dat', dataE11, fmt='%15.7e %15.7e ')

mylist=[]
for inode in range(0,nnod):
    mylist.append(nodes[inode].coordinates)

x= numpy.array(mylist)
```

```
mylist=[]
disp_big = odb.steps['Step-1'].frames[1].fieldOutputs['U']

for inode in range(1,nnod):
    mylist.append(disp_big.values[inode].data )

disp= numpy.array(mylist)

mylist=[]
for ielement in range(0,nelm):
    if len(elements[ielement].connectivity) == 8:
        ab=list(elements[ielement].connectivity)
        ab.insert(0,elements[ielement].label)
        mylist.append(ab)
    else:
        a=[0,1,2,3,4,5,6,7]
        a.insert(0,elements[ielement].label)
        mylist.append(a)

icon= numpy.array(mylist)
nelsets = len(odb.rootAssembly.elementSets['AREA'].elements[0])
mylist=[]
elset_big = odb.rootAssembly.elementSets['AREA']
for i in range(0,nelsets):
    mylist.append(elset_big.elements[0][i].label)

elset= numpy.array(mylist)
iintp=0;
ngp=9;
stress_big = odb.steps['Step-1'].frames[1].fieldOutputs['S']
strain_big = odb.steps['Step-1'].frames[1].fieldOutputs['E']
stress = numpy.zeros(shape=(0,5))
strain = numpy.zeros(shape=(0,5))
temp = numpy.zeros(shape=(5))

for ielement in  elset:
    #print ielement
  for iintp in range(0,9):
    index=(ielement-1)*ngp+iintp
    #index=(ielement)*ngp+iintp
    temp[0] =  ielement
    temp[1] =  iintp+1
    temp[2] = stress_big.values[index].data[0]
    temp[3] = stress_big.values[index].data[1]
    temp[4] = stress_big.values[index].data[3]
    stress=numpy.vstack((stress,temp))
    temp[0] =  ielement
    temp[1] =  iintp+1
    temp[2] = strain_big.values[index].data[0]
    temp[3] = strain_big.values[index].data[1]
    temp[4] = strain_big.values[index].data[3]
#            temp[2] = ...
    odb.steps['Step-1'].frames[1].fieldOutputs['E'].values[index].data[0]
#            temp[3] = ...
    odb.steps['Step-1'].frames[1].fieldOutputs['E'].values[index].data[1]
#            temp[4] = ...
    odb.steps['Step-1'].frames[1].fieldOutputs['E'].values[index].data[3]
    strain=numpy.vstack((strain,temp))

numpy.savetxt(jname+'_x.dat', x)
numpy.savetxt(jname+'_disp.dat', disp)
numpy.savetxt(jname+'_stress.dat', stress, fmt='%8i %8i %15.7e %15.7e %15.7e ')
numpy.savetxt(jname+'_strain.dat', strain, fmt='%8i %8i %15.7e %15.7e %15.7e ')
numpy.savetxt(jname+'_icon.dat', icon, fmt='%8i')
prtc=nodes[la-1].coordinates
#prtd=disp_big.values[la-1].data
data1=[l,r1,r2,alpha,E,nu,pressure,la,prtc[0],prtc[1],w,b,dispbv]
data1= numpy.array(data1)
numpy.savetxt(jname+'_data.dat', data1, fmt='%15.7e')
odb.close()
####### End Of Program ############
```

Appendix C: Stress intensity factor for rigid line inclusion

In this appendix, we discuss the consequences of quantifying the singularity in the elastic fields near the inclusion tip in terms of a stress intensity factor. The stress intensity factor for a rigid line inclusion under tensile load is defined as (see for example [33])

$$K_I = \lim_{r\to 0}(2\pi r)^{1/2}\sigma_{22}(\theta = 0^\circ), \tag{C.1}$$

where σ_{22} is the stress ahead of a rigid line inclusion tip. On substituting the value of σ_{22} from Eqn. (4.42.2) in Eqn. (C.1), the stress intensity factor can be written as

$$K_I = \frac{\mu\epsilon_{11}^{\infty}(1-\kappa)}{\kappa} \tag{C.2}$$

where ϵ_{11}^{∞} is a remote strain applied. Note that the above mentioned SIF expression turns out to be a function of Poisson's ratio of the matrix, as discussed in the main text. Therefore, the stress intensity factor cannot be used as a loading parameter for the present inclusion problem.

For completeness, we discuss the expression for the stress intensity factor as a function of remotely applied stress σ_{11}^{∞}. The remote strain ϵ_{11}^{∞} can be expressed in terms of the applied stress σ_{11}^{∞} as $\epsilon_{11}^{\infty} = (\sigma_{11}^{\infty}/(8\mu))(\kappa - 1 + 2\cos(2\alpha))/\kappa$, where α is the orientation of the inclusion with respect to the x_1 direction. The obtained SIF expression, shown in Eqn. (C.2), turns out to be a function of Poisson's ratio. Figure (C.1) shows the variation of SIF with respect to increasing Poisson's ratio for $\alpha = 0°$. The SIF is negative for all values of Poisson's ratio, except at 0.5, for which the SIF vanishes.

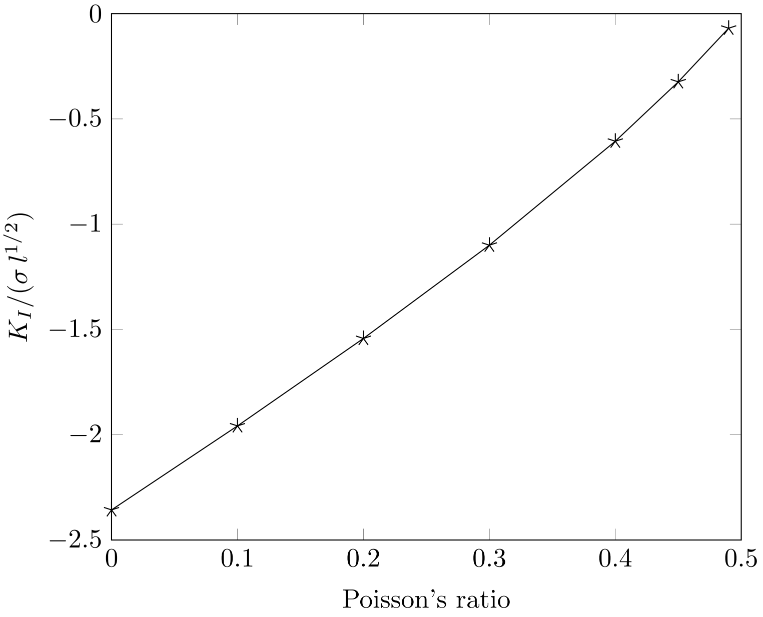


Figure C.1: Normalized analytical SIF value versus Poisson's ratio obtained in case of an inclusion kept parallel to the remotely applied stress, i.e., $\alpha = 0$.

Figure (C.2) shows the SIF versus Poisson's ratio obtained for a rigid line inclusion with $\alpha = 90°$. i.e., the inclusion is transverse to the applied stress. Now Eqn. (C.2) can be written as

$$K_I = \frac{\sigma_{11}^{\infty}}{8}\frac{(\kappa-3)(1-\kappa)}{\kappa}\sqrt{\frac{l}{2}} = \frac{2\sigma_{11}^{\infty}}{24}\frac{(3-\kappa)}{\kappa}\frac{E}{K}\sqrt{\frac{l}{2}}, \tag{C.3}$$

where E is Young's modulus and K is the bulk modulus defined as $K = (2E)/(3(\kappa-1))$. As the Poisson's ratio is increased from zero, there are two counteracting contributions to the SIF. The factor $(3-\kappa)/\kappa$ vanishes for $\nu = 0$ and increases with increasing Poisson's ratio value. On the other hand the factor $1/K$ decreases with an increase in the Poisson's ratio and vanishes for $\nu = 1/2$. As a consequence the SIF vanishes for $\nu = 0$ and $1/2$. The reason for vanishing of the SIF for $\nu = 0$ can be understood as follows. The rigid line inclusion tries to prevent the lateral contraction of matrix under a transverse loading, which leads to singularity. In case of zero Poisson's ratio, the lateral contraction is absent and therefore the SIF vanishes.

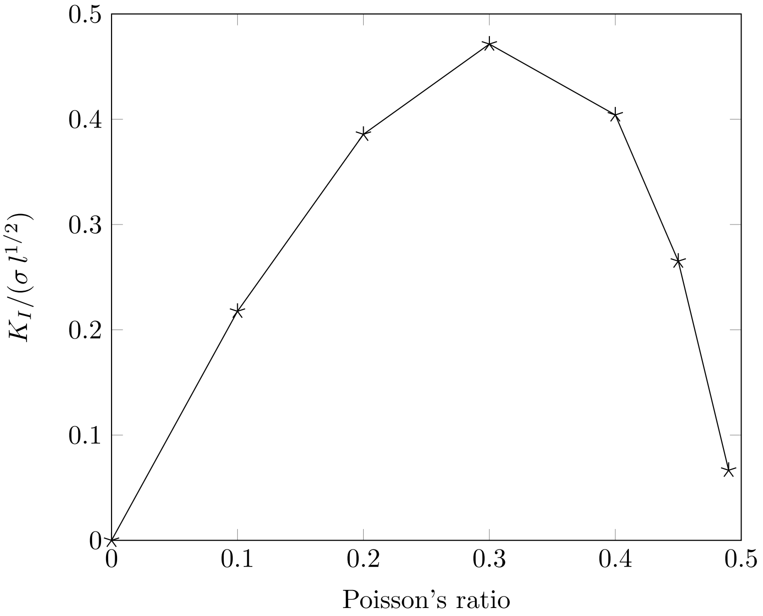


Figure C.2: Normalized analytical SIF value versus Poisson's ratio obtained in case of an inclusion kept transverse to the applied tensile load, i.e., $\alpha = 90°$.

Appendix D: Fringe pattern plotting from FEA results [50]

In this appendix, procedure to plot dark field photoelastic fringe pattern from FEA results is explained. It is adopted from the work of Ramesh et al. [50]. The optical technique photoelasticity gives the whole field information in the form of fringe contours corresponding to principal stress difference and principal stress orientation. These fringe orders are related to principal stress difference as shown in Eqn. 4.87. The FEA solution provides the displacement and stress value at nodes. The value of principal stress difference $(\sigma_1 - \sigma_2)$ has to be calculated from σ_x, σ_y and τ_{xy} at all the nodes. The fringe order N are then calculated at all the nodes using Eqn. (4.87). The fringe order N_i, nodal co-ordinates and nodal connectivity data are considered as input from FEA. The entire element in the model is scanned discretely and field variable is calculated at each point. Here, field variable is the fringe order. The quality of fringes depends on the scanning interval. Scanning interval of 0.01 mm is fixed to get a good quality fringe plot. Figure (D.1) shows the schematic of element with fringe and scanning interval. Further, the intensity value at pixel is plotted by checking whether fringe order is equal to an integer contour value. For each point, the global co-ordinates and fringe order is estimated using the shape function as interpolation function which is given below

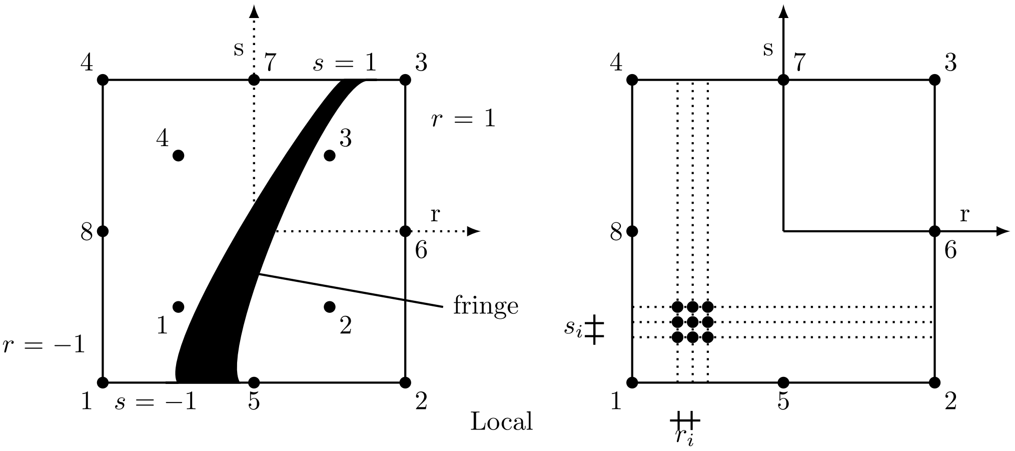


Figure D.1: The element in natural coordinate system (a) fringe contour (b) scanning interval.

$$\begin{aligned} x_g &= N_1 x_1 + N_2 x_2 + \cdots + N_8 x_8 \\ y_g &= N_1 y_1 + N_2 y_2 + \cdots + N_8 y_8 \\ frn_g &= N_1 frn_1 + N_2 frn_2 + \cdots + N_8 frn_8 \end{aligned} \tag{D.1}$$

where x_i, y_i are co-ordinates of the nodes of the element, x_g, y_g are global co-ordinates of the point, $N_1, \cdots, N_8$ are the shape functions, $frn_1, \cdots, frn_8$ are the fringe order at node and frn_g is fringe order of the point.