Computational Materials Science 53 (2012) 354–358

Contents lists available at SciVerse ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

Stress intensity factor predictions: Comparison and round-off error

Antonio Carlos de Oliveira Miranda [a,*], Marco Antonio Meggiolaro [b], Luiz Fernando Martha [c], Jaime Tupiassú Pinho de Castro [b]

[a] Department of Civil and Environmental Engineering, University of Brasília, SG-12 Building, Darcy Ribeiro Campus, DF 70.910-900, Brazil
[b] Department of Mechanical Engineering, Pontifical Catholic University of Rio de Janeiro, Rua Marquês de São Vicente 225 – Gávea, Rio de Janeiro, RJ 22453-900, Brazil
[c] Department of Civil Engineering, Pontifical Catholic University of Rio de Janeiro, Rua Marquês de São Vicente 225 – Gávea, Rio de Janeiro, RJ 22453-900, Brazil

ARTICLE INFO

Article history:
Received 12 August 2011
Received in revised form 20 September 2011
Accepted 22 September 2011
Available online 22 October 2011

Keywords:
Curved crack path prediction
Fatigue under complex loading
Stress intensity calculation
Finite element implementation

ABSTRACT

Practical steps required to obtain robust finite element triangular meshes for crack path and stress intensity calculation purposes are evaluated, and techniques to use their predictions to calculate fatigue lives, including load interaction effects, are discussed. These steps address: (a) how to simulate efficiently 2D crack paths under bi-axial loading using automatic remeshing schemes; (b) how to choose the best method to calculate stress intensity factors along the crack path; and (c) how the numerical problems associated with excessive FE mesh refinement along the crack path may affect predictions. Various modeling strategies are compared using different crack geometries and mesh refinements to quantify their performance, particularly when the elements around the crack tip are very small compared with the elements far from it. It is shown that, contrary to many other stress analysis applications, excessive mesh refinement may significantly degrade the calculation accuracy in crack problems. A limit for the elements size ratio is clearly established.

1. Introduction

The theory required to predict the generally curved crack propagation path in two-dimensional (2D) structural components under bi-axial loading is well known, but its implementation in an efficient and reliable code is still far from a trivial task. The purpose of this work is to describe how the difficulties involved in translating such theoretical tools into practical numerical techniques have been solved, and how these techniques were used in a successful special-purpose academic program called *Quebra2D*, which means 2D fracture in Portuguese [1,2]. This software is an interactive graphics program for simulating two-dimensional fracture processes based on numerical finite element (FE) techniques. Three important subjects required to accomplish a complete and robust simulation of fatigue crack growth (FCG) are studied: (a) how to compute efficiently FCG under complex loading; (b) how to deal with numerical problems that come up when the element size at crack tip is very small compared to the entire model size; and (c) how to choose the best method to compute stress intensity factors (SIF).

The complete automatic simulation of crack propagation in 2D using FE method was first presented by Bittencourt et al. [3] and Lim et al. [4]. Bittencourt used an advancing front method to create an initial FE mesh, and then redefined this mesh locally, in a region close to the crack tip, in the subsequent FCG simulation steps, avoiding in this way the need for global remeshing at each FCG step. In his simulation, three methods to compute SIF [5–8] and three techniques to compute the crack path incremental direction were employed [9–11]. Lim also used a local procedure to remesh just close to the crack tip, implementing four displacement-based SIF computation techniques [12]. Many similar works have been published [2,13–22], generally proposing new mesh generation algorithms or new methods to improve the numerical computation of SIF values. For example, previous works by this paper authors [2,16] proposed a FCG under complex loading simulation technique that uses an unbound global–local technique. This is a better approach than simulating FCG considering constant amplitude loading in Paris equation [23], because it only represents part of the material crack growth behavior. However, none of the previous works not provide sufficient information to create an environment to efficiently compute FCG.

This work does not intend to re-analyze the advantages and disadvantages of using FE in computational fracture mechanics, neither to compare FE with other approaches. This task has been recently addressed by Ingraffea [24], who studied the taxonomy of the many methods used to represent cracks in a numerical model, and the fundamental differences between them. Details about the historical development of computational fracture mechanics are not discussed here either, as they can be obtained in the works by Ingraffea and Wawrzynek [25] and Sinclair [26], for instance.

* Corresponding author. Tel.: +55 61 3107 0994; fax: +55 61 3273 4644.
E-mail addresses: acmiranda@unb.br (A.C. de Oliveira Miranda), meggi@puc-rio.br (M.A. Meggiolaro), lfm@tecgraf.puc-rio.br (L.F. Martha), jtcastro@puc-rio.br (J.T.P. de Castro).

0927-0256/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.commatsci.2011.09.033

In the following sections, first a brief description of a computational architecture required to implement a robust and efficient FE-based FCG program is presented. Then important details needed to automate crack growth numerical predictions under complex loading are discussed. Finally, based on 864 FE calculations, the various SIF predictions schemes are compared.

2. Automatic crack propagation

The automatic crack propagation strategy adopted in this work is based on a full geometric description of the cracked body two-dimensional model. This means that there is a geometric model that represents the cracked structure, and the FE mesh is attached to this geometric model. Analysis attributes, such as loads, support conditions, and material properties are also attached to the geometric model. The geometric description consists of a set of curves that represent the boundaries of the regions in the model. Boundary conditions (supports and/or loads) are associated with the curves, and domain parameters (such as material properties and thickness) are attached to regions.

This strategy was implemented in the *Quebra2D* [1,2] program. The crack representation scheme used in *Quebra2D* is based on the discrete approach. In this sense, the program is similar to well-known 2D simulators, such as Franc2D [3,27] for instance. *Quebra2D* includes all methods mentioned above to compute crack increment directions and associated SIF along the crack path. Moreover, its adaptive FE analyses are coupled with modern and very efficient automatic remeshing schemes, which substantially decrease the computational effort.

Fig. 1 shows a simplified structure of the *Quebra2D* program. The program does not have a direct user interface. Instead, it provides a medium level C++ application programming interface (API) module. This allows *Quebra2D* to be run without user interference, a desirable situation when it is used within an external code, or when running several examples within a programming loop.

As shown in Fig. 1, *Quebra2D* is composed by several modules controlled by a "Manager" that communicates with the API module. These modules are: *Mesh Structure*, which stores the mesh and the FE results; *Geometric Structure*, which stores the crack and piece geometries; *Mesh Generator*, responsible for the automatic FE generation, including the meshing and re-meshing steps; *Rosette Shapes*, responsible for generating the special finite elements around the crack tip; *Attributes*, where the model attributes such as nodal restrictions, loads and material properties are stored; *Solve*, module responsible for the numerical analysis; *SIF Calculator*, which calculates K_I and K_{II} stress intensity factors using the FE results; *Crack Growth Direction*, which calculates the crack increment angle at each calculation step; and finally *Jobs*, which is the module responsible for determining the type of the static, crack propagation, and mesh adaptive analysis. This program modular structure allows any module to be altered without interfering with the others. For instance, *Quebra2D* can be used with almost any commercial FE code.

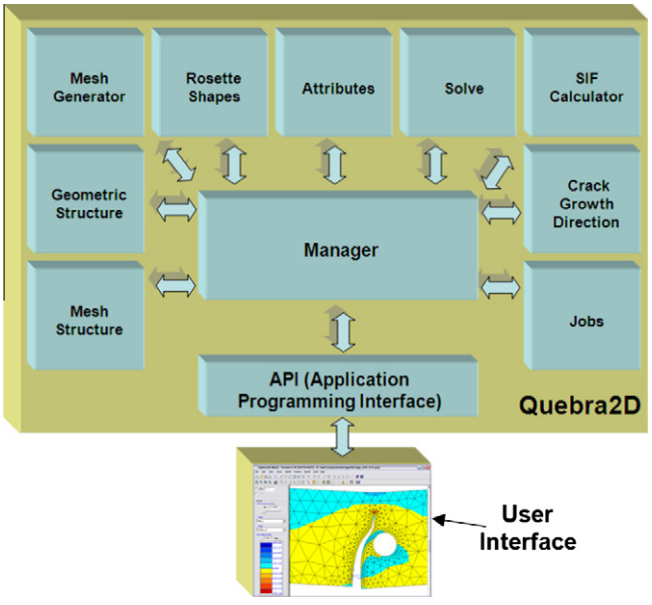

Fig. 1. Internal structure of *Quebra2D*.

The *Mesh Generator* module contains an algorithm to generate triangular elements [28]. The meshing algorithm works both for regions without cracks or with one or multiple cracks, which may be either embedded or surface breaking. This is an adaptation of an algorithm previously proposed for generating unstructured meshes for arbitrarily shaped three-dimensional regions [29]. The same strategy was adopted for surface mesh generation [30].

A typical computational procedure to predict propagation of curved cracks adopts the following sequence of steps. Given an initial geometric model with cracks and attributes, a few options must be chosen or read from an input data file: the method to obtain K_I and K_{II}; the crack increment direction criterion; the equivalent SIF K_{eq} criterion; the propagation threshold ΔK_{th}; the material or structure toughness K_C; the load ratio $R = K_{min}/K_{max}$; the constants of the material FCG rule, such as Paris' coefficient and exponent A and m (when dealing with multiple cracks that can interact); the maximum crack increment size Δa; and the number of steps or increments required to simulate the crack path. The next simulation step is to create a mesh in the model domain, followed by the FE analysis. Then, the equivalent SIF ΔK_i and the global angle of crack propagation direction θ_i are obtained from the FE results, for each crack in the model. In addition, the maximum equivalent SIF ΔK_{max} is obtained searching through all ΔK_i, and the equivalent SIF at the maximum load is found from the load ratio R by $K_{max} = \Delta K_{max}/(1 - R)$. After finishing all these tasks, the entire process is restarted for the next crack growth simulation step.

However, this global sequential procedure, in spite of being adopted by most FCG simulation programs, is not efficient for modeling long variable amplitude loading histories. Indeed, such complex load histories usually cause different crack increments at each load cycle, which require remeshing and time-consuming FE recalculations. Moreover, load interaction effects such as FCG retardation after overloads compromise even more the computational efficiency of this approach.

To cope with this problem, the authors have proposed an unbound global–local approach in which crack increments are locally calculated at each load cycle in an efficient manner, even considering crack retardation effects if necessary. The local approach does not require a global solution of the whole structure stress field. It is based on the direct integration of the fatigue crack propagation rule of the material, $da/dN = F(K, K_{max}, K_{th}, K_C, \ldots)$, where K and K_{max}, the stress intensity range and maximum, are the two fatigue cracking driving forces; K_{th} and K_C, the fatigue crack propagation threshold and the toughness of the material-structure set, are the material related properties that induce the sigmoidal shape of typical da/dN curves; whereas the ellipsis represent the possible influence of other parameters on FCG, such as the opening SIF K_{op}, or non-mechanical (e.g. environmental or chemical) driving forces. Appropriate expressions for K and for the da/dN rule must be used to obtain satisfactory cracking life predictions.

The need for good SIF expressions is a major drawback of the local approach, because they simply are not available for most real components, in which cracks tend to curve while they cross non-uniaxial stress fields. In such cases, designers must use engineering

common sense to choose approximate K_I handbook expressions to simulate FCG real problems by the local approach. The error involved in such approximations obviously increases as the real crack deviates from the modeled crack, and in such cases the accuracy of the local approach is questionable and its predictions unreliable. In spite of this drawback, this problem can be efficiently solved as follows.

Since the advantages of the global and local approaches are complementary, the problem can be successfully divided into two parts. First, an appropriate FE program (such as *Quebra2D*) is used to calculate the generally curved crack path and its associated Mode I stress intensity factor $K_I(a)$ along the crack length a, under constant amplitude loading. Then, an analytical expression is adjusted to the discrete $K_I(a)$ values calculated at as many points as required along the crack path, to be used as input to a local approach program (such as the *ViDa* program, described below). Finally, the actual variable amplitude loading can be efficiently treated by the direct integration of the crack propagation rule, considering retardation effects if required. Neither the ΔK expression nor the type of crack propagation rule need to have their accuracy compromised when using this dual approach, a hypothesis experimentally verified in many cases.

The local approach program used in this work is named *ViDa* (which means life in Portuguese). This piece of software has been developed to automate all the traditional local approach methods used in fatigue design [31,32], including the *SN*, the *IIW* (for welded structures) and the εN methods for crack initiation, and the *da/dN* method for crack propagation. FCG lives can be calculated by the ΔK_{rms} or the cycle-by-cycle methods, for either planar or 3D problems, as long as proper SIF expressions are provided (e.g. which can be calculated by *Quebra2D*.).

To properly simulate FCG under variable amplitude loading problems, several load interaction models are included in the *ViDa* software [33–36]. For example, Wheeler's model, perhaps one of the simplest and most well known [33], introduces a crack-growth reduction factor bounded by zero and unity. This factor is calculated for each cycle to predict retardation as long as the current plastic zone Z_i is contained within a previously overload-induced plastic zone Z_{ol}. The retardation is maximum just after the overload, and it ends when the border of Z_i touches the border of Z_{ol}.

Therefore, if a_{ol} and a_i are the crack sizes at the instant of the overload and at the (later) ith cycle, and $(da/dN)_{ret,i}$ and $(da/dN)_i$ are the retarded and the corresponding non-retarded crack growth rates (at which the crack would be growing in the ith cycle if the overload had not occurred), then, according to Wheeler:

$$\left(\frac{da}{dN}\right)_{ret,i} = \left(\frac{da}{dN}\right)_i \cdot \left(\frac{Z_i}{Z_{ol} + a_{ol} - a_i}\right)^{\beta}, \quad \text{if } a_i + Z_i < a_{ol} + Z_{ol} \qquad (1)$$

where β is a fitting constant, obtained by selecting the closest match between an experimentally measured curve under variable amplitude loading and the predicted FCG curves using several β-value candidates. However, this model cannot predict crack arrest because the resulting $(da/dN)_{ret,i}$ is always positive. Cut-off values have been proposed to include crack arrest in the original Wheeler model, however this approach results in discontinuous $(da/dN)_{ret,i}$ equations. Meggiolaro and Castro [37] used a simple but effective modification to the original Wheeler model in order to predict both crack retardation and arrest. This approach, called the *Modified Wheeler* model, uses a Wheeler-like parameter to multiply ΔK instead of *da/dN* after the overload

$$\Delta K_{ret}(a_i) = \Delta K(a_i) \cdot \left(\frac{Z_i}{Z_{ol} + a_{ol} - a_i}\right)^{\gamma}, \quad \text{if } a_i + Z_i < a_{ol} + Z_{ol} \qquad (2)$$

where $\Delta K_{ret}(a_i)$ and $\Delta K(a_i)$ are the values of the SIF ranges that would be acting at a_i with and without retardation due to the overload, and γ is an experimentally adjustable constant, usually different from the original Wheeler model exponent β. This simple modification can be used with any of the FCG rules that recognize ΔK_{th} to predict both retardation and arrest of fatigue cracks after an overload, the arrest occurring if $\Delta K_{ret}(a_i) \leqslant \Delta K_{th}$.

Another crack retardation model included in *ViDa* is the Constant Closure model, originally developed at Northrop for use in their classified programs [34]. It is based on the observation that for some variable amplitude load histories the closure stress does not deviate significantly from a certain stabilized value (if the spectrum has a "controlling overload" and a "controlling underload" that occur often enough to keep the residual stresses constant, and thus the closure level constant).

In the constant closure model, the opening SIF K_{op} is the only empirical parameter, with typical values estimated between 20% and 50% of the maximum overload stress intensity factor. The value of K_{op}, calculated for the controlling overload event, is then applied to the following (smaller) loads to compute crack growth, recognizing crack retardation and even crack arrest (if $K_{max} \leqslant K_{op}$).

The main limitation of the Constant Closure model is that it can only be applied to loading histories with "frequent controlling overloads," because it does not model the decreasing retardation effects as the crack tip cuts through the overload plastic zone. In this model, it is assumed that a new overload zone, with primary plasticity, is formed often enough before the crack can significantly propagate through the previous plastic zone, thus not modeling secondary plasticity effects by keeping K_{op} constant.

3. Stress intensity factor predictions

Sinclair [26] presents an extensive review of SIF numerical prediction models. However, most comparisons are somewhat incomplete in at least one of three important aspects. First, they discuss numerical results generated by "artificial" meshes that do not adapt to the (growing) crack path, contrary to the automatically generated meshes presented in this paper. Second, they do not show convergence analysis of the results when the elements size is decreased around the crack tip, increasing the number of elements in the analyzed numerical model. Third, they present only Mode I results.

This section compares SIF values obtained by three methods using the *Quebra2D* methodology: the displacement correlation technique (DCT) [5]; the potential energy release rate, computed by means of a modified crack-closure integral technique (MCC) [6,7]; and the *J*-integral, computed by means of the equivalent domain integral (EDI) together with a mode decomposition scheme [8,38–41]. The objective of this study is to overcame the three limitations of previous comparisons available in the literature. The model shown in Fig. 2a is employed to compare SIF calculated by using the three different methods in FE analysis. This model is a representation of a very large plate with a small crack, aiming to reproduce the infinite plate solution. The plate has 2000×2000 mm with an inclined central crack of length $2a = 2$ mm. The plate is loaded by a uniform stress in the vertical direction; see Fig. 2a. This example is studied varying the crack angle with the horizontal axis from 0° to 80°, at 10° steps. To analyze the SIF calculation convergence, the density of the FE mesh is modified varying the number n of the elements at each of the two crack faces, with $n = 4$, 8, 16, 32, 64, 128, 256 and 512 elements. In all models, eleven nodes are used at each edge of the plate. Fig. 2b shows an example of a mesh generated by *Quebra2D* for the 80° crack with four elements on each of the two crack faces.

Four different strategies for mesh generation were considered in the model, featuring meshes with or without additional refinement to force equal size between neighboring cells near the crack

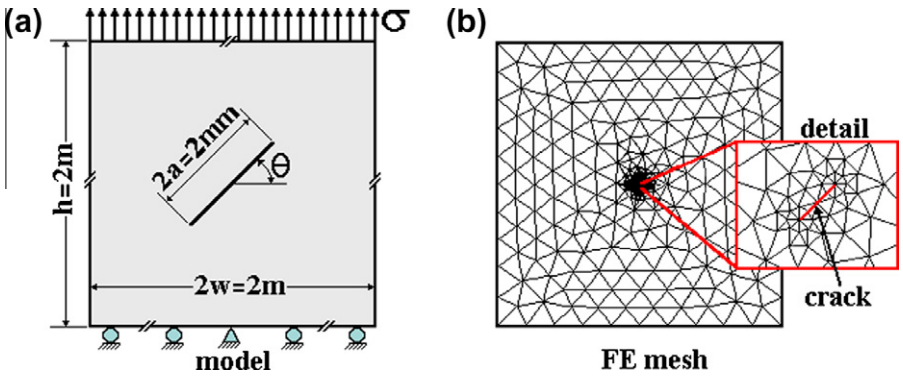

Fig. 2. (a) Model used to compare the SIF calculated by the three different methods and (b) typical FE mesh.

tip and, for either case, with or without performing additional smoothing on the final meshes. In this way, elements around the crack tip ended up with different shapes and arrangements. FE analyses were performed for each of the four meshing strategies, for eight different numbers of crack face elements (n = 4, 8, 16, 32, 64, 128, 256 and 512), nine considered crack angles (0°, 10°,. . ., 80°), and three K_I calculation models (DCT, MCC and EDI), resulting in 864 calculations. It was found that the K_I calculation errors do not significantly depend on the meshing strategy or on the crack angle, but they are very much influenced by the number of crack face elements and calculation model. Figs. 3 and 4 show the ratio between the FE-calculated and the actual (theoretical) Mode I or II SIF for different crack angles and calculation methods, as a function of the number of crack face elements.

It is clearly seen from Figs. 3 and 4 graphs that, in average, the EDI SIF calculation method results in better predictions than the MCC, which in turn is usually better than the DCT. Note from Fig. 4 that the value of K_{II} calculated by the DCT method does *not* converge to the theoretical value as the mesh is refined. For the other cases, the calculation precision tends to improve (i.e., the ratio between the calculated and actual values tends to 1) as the number of crack face elements is increased.

Note, however, that the calculation errors and associated standard deviations tend to increase for a number of crack face elements greater than 16. These increasing errors are a result of an ill-conditioned numerical problem [42]. Formally, the condition number of a matrix is defined as the ratio between its largest eigenvalue (in absolute value) and its eigenvalue closest to zero. A matrix is singular if its condition number is infinite, and it is ill-conditioned if its condition number is too large, that is, if its reciprocal approaches the machine's floating-point precision [43] (e.g., less than 10^{-6} for single or 10^{-12} for double precision computer variables).

The linear algebraic system solution from the linear FE analysis used by *Quebra2D* is obtained by the direct Gauss–Jordan elimination method. This method itself is not applicable to

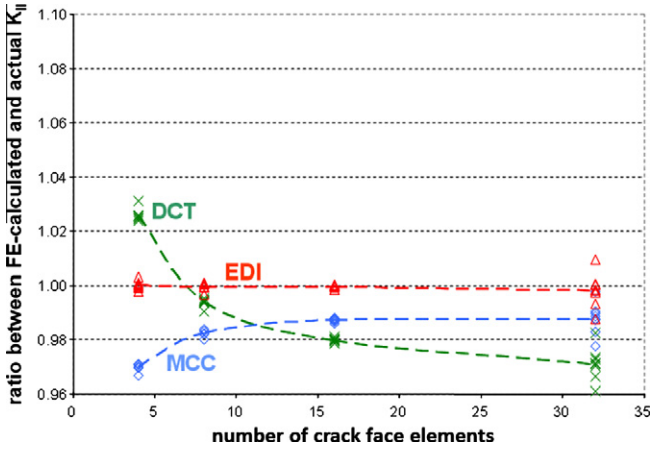

Fig. 4. Ratio between FE-calculated and actual Mode II SIF for different crack angles and calculation methods, as a function of the number of crack face elements; the dashed lines are averages over the nine analyzed crack angles (0°, 10°,. . .,80°).

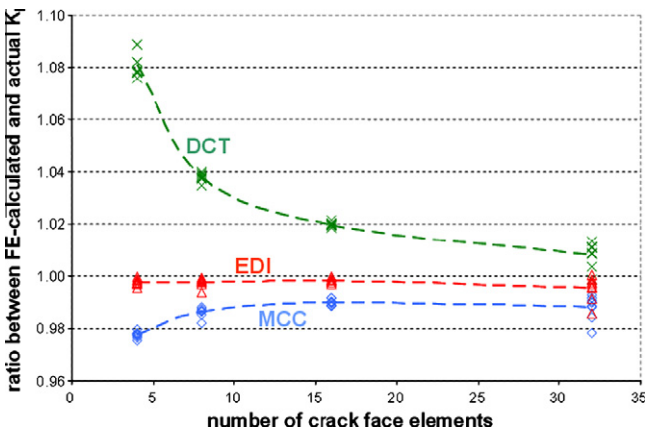

Fig. 3. Ratio between the FE-calculated and the actual Mode I SIF for different crack angles and calculation methods, as a function of the number of crack face elements; the dashed lines are averages over the nine analyzed crack angles (0°, 10°,. . .,80°).

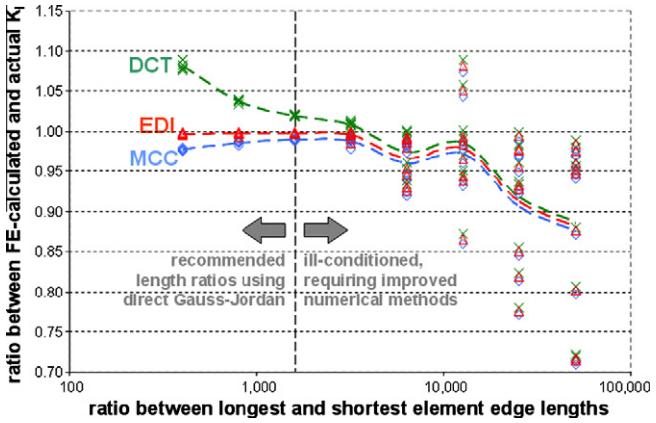

Fig. 5. FE-calculated to actual Mode I SIF ratio for different crack angles and calculation methods, as a function of the longest-to-shortest element edge length ratio; the dashed lines are averages over the nine analyzed crack angles (0°, 10°,. . .,80°).

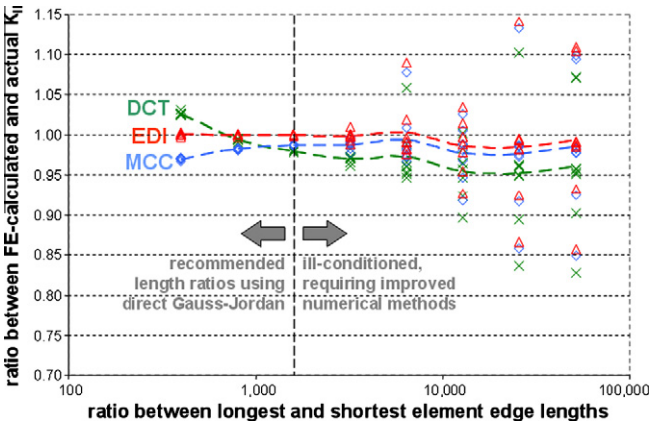

Fig. 6. FE-calculated to actual Mode II SIF ratio for different crack angles and calculation methods, as a function of the longest-to-shortest element edge length ratio; the dashed lines are averages over the studied crack angles (0°, 10°,...,80°).

ill-conditioned matrices. This ill-conditioned problem is caused here by the large disparity between the sizes of elements in the mesh. In all the studied examples, the longest element edge has length 200 mm, located at the plate border. The shortest edge occurs at each crack face, with length $2a/n$, where $2a = 2$ mm is the considered crack length and n is the number of elements at each crack face. Thus, the ratio between the longest and shortest element edges is given by $200/(2/n) = 100 \cdot n$, which varies between 400 and 51,200 for the considered values of n between 4 and 512.

Figs. 5 and 6 show the ratio between the FE-calculated and the actual Mode I or II SIF, including results from numbers n of crack face elements above 32. It is possible to verify from these figures that the calculation errors dramatically increase for high longest-to-shortest element edge length ratios $100 \cdot n$. This work does not intend to solve this numerical problem, however many solutions can be found in literature [43]. Therefore, unless such ill-conditioning problem is addressed, it is recommended to perform the calculations using meshes with length ratios $100 \cdot n$ up to $100 \times 16 = 1600$.

4. Conclusions

In this work, computational algorithms used to calculate the curved path of fatigue cracks in generic 2D structural components were presented. The actual performance of the three main SIF calculation methods was evaluated by comparing their predictions on 864 FE calculations with different crack geometries and mesh refinements on a very large plate, with the idealized analytical solution for this problem. It was found that the EDI method results in better predictions than the MCC, which in turn is usually better than the DCT. In some cases, the DCT method does not converge to the theoretical value of SIF.

Using the direct Gauss–Jordan elimination method in the linear FE analysis, it was found that the ratio between the longest and shortest element edge lengths should be kept below 1600 to avoid calculation errors in SIF calculations. For meshes with length ratios higher than 1600, improved numerical methods to deal with ill-conditioned matrices would be necessary to not compromise the calculation accuracy of the calculated SIF.

The main models to predict the crack incremental direction were reviewed. An automatic crack propagation algorithm was presented to calculate the curved crack path and associated SIF.

References

[1] A.C.O. Miranda, M.A. Meggiolaro, J.T.P. Castro, L.F. Martha, T.N. Bittencourt, in: M.P. Braun AA, Lohr RD (Eds.), Applications of Automation Technology in Fatigue and Fracture Testing and Analysis, 2002, pp. 120–146.
[2] A.C.O. Miranda, M.A. Meggiolaro, J.T.P. Castro, L.F. Martha, T.N. Bittencourt, Engineering Fracture Mechanics 70 (10) (2003) 1259–1279.
[3] T.N. Bittencourt, P.A. Wawrzynek, A.R. Ingraffea, J.L. Sousa, Engineering Fracture Mechanics 55 (2) (1996) 321–334.
[4] I.L. Lim, I.W. Johnston, S.K. Choi, Engineering Fracture Mechanics 53 (2) (1996) 193–211.
[5] C.F. Shih, H.G. de Lorenzi, M.D. German, International Journal of Fatigue 12 (1976) 647–651.
[6] E.F. Rybicki, M.F. Kanninen, Engineering Fracture Mechanics 9 (4) (1977) 931–938.
[7] I.S. Raju, Engineering Fracture Mechanics 28 (3) (1987) 251–274.
[8] H.D. Bui, Journal of Mechanics & Physics Solids 13 (1983) 439–448.
[9] F. Erdogan, G.C. Sih, Journal of Basic Engineering 85 (1963) 519–527.
[10] M.A. Hussain, S.U. Pu, J. Underwood, in: ASTM STP 560, 1974, pp. 2–28.
[11] G.C. Sih, International Journal of Fracture 10 (3) (1974) 305–321.
[12] I.L. Lim, I.W. Johnston, S.K. Choi, International Journal of Fracture 58 (3) (1992) 193–210.
[13] P.O. Bouchard, F. Bay, Y. Chastel, I. Tovena, Computer Methods in Applied Mechanics and Engineering 189 (3) (2000) 723–742.
[14] J.M. Alegre, I.I. Cuesta, International Journal of Fatigue 32 (7) (2010) 1090–1095.
[15] P.O. Bouchard, F. Bay, Y. Chastel, Computer Methods in Applied Mechanics and Engineering 192 (35–36) (2003) 3887–3908.
[16] A.C.O. Miranda, M.A. Meggiolaro, J.T.P. de Castro, L.F. Martha, International Journal of Fatigue 25 (9–11) (2003) 1157–1167.
[17] S. Phongthanapanich, P. Dechaumphai, Finite Elements in Analysis and Design 40 (13–14) (2004) 1753–1771.
[18] P. Heintz, International Journal for Numerical Methods in Engineering 65 (2) (2006) 174–189.
[19] A. Alshoaibi, M. Hadi, A. Ariffin, Journal of Zhejiang University – Science A 8 (2) (2007) 228–236.
[20] A.R. Khoei, H. Azadi, H. Moslemi, Engineering Fracture Mechanics 75 (10) (2008) 2921–2945.
[21] D. Azócar, M. Elgueta, M.C. Rivara, Advances in Engineering Software 41 (2) (2010) 111–119.
[22] D. Rozumek, C.T. Lachowicz, E. Macha, Engineering Fracture Mechanics 77 (11) (2010) 1808–1821.
[23] P.C. Paris, M.P. Gomez, W.E. Anderson, The Trend in Engineering 13 (1961) 9–14.
[24] A.R. Ingraffea, in: Encyclopedia of Computational Mechanics, John Wiley and Sons, 2004.
[25] A.R. Ingraffea, P.A. Wawrzynek, in: R.d.B.a.H. Mang (Ed.), Comprehensive Structural Integrity, Elsevier Science Ltd., Oxford, England, 2003.
[26] G. Sinclair, Applied Mechanics Reviews 57 (4) (2004) 251–298.
[27] P.A. Wawrzynek, A.R. Ingraffea, Theoretical and Applied Fracture Mechanics 8 (2) (1987) 137–150.
[28] A.C. Miranda, J.B. Cavalcante Neto, L.F. Martha, in: XII Brazilian Symposium on Computer Graphics and Image Processing (Cat. No.PR00481), 1999, pp. 29–38.
[29] J.B. Cavalcante Neto, P.A. Wawrzynek, M.T.M. Carvalho, L.F. Martha, A.R. Ingraffea, Engineering with Computers 17 (1) (2001) 75–91.
[30] A.C.O. Miranda, L.F. Martha, P.A. Wawrzynek, A.R. Ingraffea, Engineering with Computers 25 (2) (2009) 207–219.
[31] J.T.P. Castro, M.A. Meggiolaro, Revista Brasileira de Ciências Mecânicas, 21 (1999) 294–312 (in Portuguese).
[32] M.A. Meggiolaro, J.T. Pinho de Castro, Journal of the Brazilian Society of Mechanical Sciences 20 (4) (1998) 666–685.
[33] O.E. Wheeler, Transactions of the ASME Series D, Journal of Basic Engineering 94 (1) (1972) 181–186.
[34] J.O. Bunch, R.T. Trammell, P.A. Tanouye, Structural Life Analysis Methods used on the B-2 Bomber, ASTM Special Technical Publication, 1996.
[35] J. Willenborg, R.M. Engle, H.A. in: Wood, Crack Growth Retardation Model Using an Effective Stress Concept, A.F.F.D. Laboratory, Wreight Patterson Air Force Base, 1971.
[36] R.H. Gallagher, A Generalized Development of Yield Zone Models, Wright Patterson Air Force Laboratory, 1974.
[37] M.A. Meggiolaro, J.T.P. Castro, in: II Seminário Internacional de Fadiga (SAE-Brasil), São Paulo, 2001, pp. 207–216.
[38] R.H.J. Dodds, P.M. Vargas, Numerical Evaluation of Domain and Contour Integrals for Nonlinear Fracture Mechanics, University of Illinois, Urbana-Champaign, 1988.
[39] L. Banks-Sills, D. Sherman, International Journal of Fracture Mechanics 32 (1986) 127–140.
[40] K.L. Chen, N. Atluri, Engineering Fracture Mechanics 34 (1989) 935–956.
[41] G.P. Nikishkov, S.N. Atluri, International Journal for Numerical Methods in Engineering 24 (9) (1987) 1801–1821.
[42] J.R. Rice, Matrix Computations and Mathematical Software, McGraw-Hill Computer Science Series, McGraw-Hill, 1981.
[43] W.H. Press, B.P. Flannery, S.A. Teukolsky, Numerical Recipes in C: The Art of Scientific Computing, Cambridge University Press, New York, 1997.