SingSol: A Web Application for Singularity Analysis in Multi-material Corners with Homogeneous Boundary Conditions

M. A. Herrera-Garrido[1,2(✉)], J. C. Cortés[2], V. Mantič[2], and A. Barroso[2]

[1] Escuela Politécnica Superior, Universidad de Sevilla, C/Virgen de Africa 7, 41011 Sevilla, Spain
mherrera13@us.es

[2] Grupo de Elasticidad y Resistencia de Materiales, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Camino de los Descubrimientos, s/n, 41092 Sevilla, Spain

Abstract. The web application SingSol offers a user-friendly and versatile tool for researchers and engineers that need to determine the singularity exponents and the singular stress and displacement fields in multi-material linear elastic corners with homogeneous boundary conditions. It covers a wide range of configurations, involving homogeneous wedges and various boundary conditions, including perfect bonding or frictionless sliding interfaces between wedges in the corner. The theoretical framework of this powerful tool considers power-law stress singularities, and is based on the Stroh formalism of anisotropic elasticity, assuming generalized plane strain conditions (2.5D), and the concept of transfer matrix for single material wedges, along with the matrix formalism for homogeneous (orthogonal) boundary conditions. This approach is well suited for computational implementations, being particularly efficient in scenarios involving multiple perfectly bonded homogeneous wedges. The fully semi-analytical nature of this tool provides its very high accuracy. SingSol comprises a graphical user interface (GUI) and a Python-based back-end code hosted on the server of the research group. It incorporates and solves limitations found in other currently available codes, providing a flexible solution for isotropic and anisotropic materials while supporting various boundary and interface conditions.

Keywords: web application · singularity · multi-material · elasticity

1 Introduction

When studying stress and strain fields in different structures, geometric, boundary, or material discontinuities leading to zones where linear elasticity predicts unbounded stresses are common. This is why a local analysis of the stress fields in these zones is recommended.

In the past, researchers studying stress singularities in cracks and corners mainly focused on developing methodologies able to solve particular cases, such as that of a single isotropic wedge under different boundary conditions [1–3], two or three perfectly

Y. Torres et al. (Eds.): STEMEPS 2023 Workshops, SPM 50, pp. 106–116, 2024.
https://doi.org/10.1007/978-3-031-64106-0_12

bonded isotropic materials [4–7] or two materials in frictionless or frictional contact [8–11], and stress-free or clamped boundary conditions, among many other cases. Furthermore, as applications of composite materials have considerably increased in last decades, the interest in the characterisation of corner singularities present in anisotropic materials has increased as well [12–15]. Therefore, there is a growing interest to address similar problems that were previously dealt for isotropic materials now involving anisotropic materials.

However, only a limited number of authors [16–22] have developed general methodologies solving a wide range of singularity problems, due to the large variety of possible cases. Nevertheless, upon examining these codes, it was noted that they possess certain limitations, primarily due to their exclusive consideration of free or clamped boundary conditions and perfectly bonded interface. Moreover, some approaches solely consider isotropic materials, not allowing the incorporation of general anisotropic materials.

In [23, 24] a semi-analytical matrix formalism was introduced to analyze displacement and stress fields in multi-material corners. This matrix formalism is based on the Stroh formalism of anisotropic elasticity, assuming generalised plane strain (2.5D) conditions, and on the semi-analytic matrix formalism for wedge transfer matrices and boundary and interface condition matrices. In this formalism the displacement and stress fields can be represented by the following asymptotic series expansions:

$$u_i(r,\vartheta) \approx \sum_{n=1}^{N} K_n r^{\lambda_n} g_i^{(n)}(\vartheta) + \dots, \tag{1}$$

$$\sigma_{ij}(r,\vartheta) \approx \sum_{n=1}^{N} K_n r^{\lambda_n - 1} f_{ij}^{(n)}(\vartheta) + \dots \tag{2}$$

where λ_n are singularity exponents and $g_i^{(n)}$ and $f_{ij}^{(n)}$ characteristic functions.

The main objective of this formalism is to compute the values of λ_n and $g_i^{(n)}$ and $f_{ij}^{(n)}$ that represent singular solutions for each corner or crack problem. Recently, in [25, 26], this formalism has been implemented in a Matlab code and successfully validated by comparing the obtained results with the available solutions for specific cases. Hence, we have opted to provide the scientific community with access to this versatile tool in a user-friendly manner.

The present work introduces the first version of the SingSol Web Application based on the general matrix formalism. The interface conditions include perfectly bonded and frictionless sliding contact, while the implemented boundary conditions include stress-free, clamped, symmetry, antisymmetry, and displacement with one or two specific directions restricted.

Section 2 provides a detailed description of the developed software, covering both the front-end and back-end aspects. Section 3 illustrates how the tool solves an important particular case with minimal user input. Finally, Sect. 4 summarizes the main advantages and limitations of the code, along with planned future developments.

2 Software Description

The Web Application for Singularity Solutions (SingSol) is a user-friendly tool comprising a graphical user interface (GUI) and a Python-based back-end code hosted on a server belonging to the Group of Elasticity and Strength of Materials (GERM) and linked to the group web:

– https://www.germus.es/corner-singularity-app/.

2.1 Front-End

The front-end, built with web technologies such as HTML, CSS, and JavaScript, incorporates a PHP web interface interacting with JavaScript code. The GUI facilitates a step-by-step user interaction: as users input data, the form dynamically adapts, revealing new options based on previous selections. Key front-end tools include Chart.js for dynamic chart generation, jQuery for handling UI updates and efficient data exchange, and Bootstrap for maintaining a consistent and responsive design. Figure 1 illustrates the flow of SingSol's front-end. The steps in the flowchart are explained below:

– To start, the user must choose between an *open corner* or a *closed corner*. The wedges in a multi-material corner are introduced in counterclockwise order, see [25, Fig. 1].

  • For an **open corner**: proceed to select the material type and set two boundary conditions, in case that another material is planned to be add to the problem, only 1 boundary condition is needed. When choosing the material type (isotropic, transversely isotropic, or orthotropic material), the user must introduce the geometry of the wedge and elastic constants of the material.
  • For a **closed corner**, the material type is selected, and one interface condition. When choosing the material type, user must introduce the elastic constants of the material and the geometry of the wedge.

– Decide whether to **add another material** or not

• If adding another material, the form checks if the corner is open or closed

  ∗ If the corner is closed, the user is asked to choose the new material type and additionally its interface condition with the previous material.
  ∗ If the corner is open, the user is prompted to select the new material type and specify its interface condition with the preceding material, along with the boundary condition of the new material. Notably, if the intention is to add another material, there is no need to include the boundary condition at this stage.

• If no more materials need to be added

  ∗ Input the interval for the real part of the singularity exponent λ and the number of points in that interval.
  ∗ Provide name, organization, and email for submission.
  ∗ Click the *Submit* button to automatically generate the *.JSON file* and send it to the cluster.

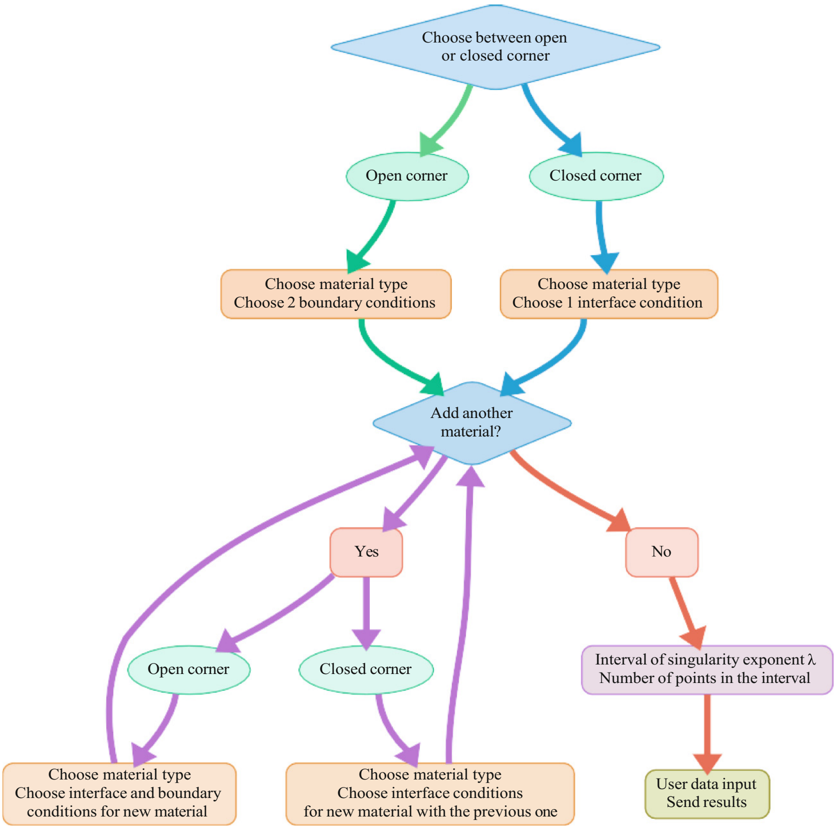

Fig. 1. Front-end SingSol form option flow.

2.2 Back-End

Once the *.JSON file* is created, it serves as the input for the Python-based back-end code. The Python code developed utilizes key libraries, namely Sympy, NumPy, and Matplotlib. Sympy is employed for symbolic mathematics, NumPy for numerical operations, and Matplotlib for creating visualizations. These libraries collectively enhance the functionality and efficiency of the code, enabling seamless integration of symbolic computation, numerical analysis, and data visualization in the Python environment. This code is hosted on a cluster and follows a modular structure, as depicted in Fig. 2.

- **Main** reads the *.JSON file*. Creates as many objects of the class *Material* and class *Interface* as needed. Calls the different modules as needed.
- **Material characterization** is called for each material in the wedge. It has three different functions, one per each type of allowed material, isotropic, transversely isotropic, or orthotropic material. These functions introduce as features of each object of class *Material* all the material properties and the geometry of the wedge, the starting and ending angles of the material wedge in the corner. The main matrices

that characterise each material, the Stroh matrices **A** and **B** and the transfer matrix **E** are saved as features of the *Material* object too.

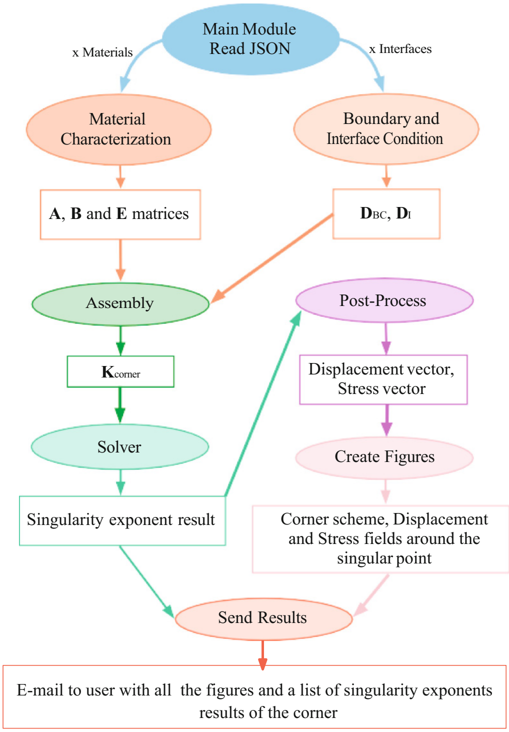

Fig. 2. Back-end SingSol chart flow.

- **Boundary and interface condition** This module is called for each boundary or interface in the corner. It calculates the Boundary condition and Interface condition matrices, $\mathbf{D}_{\mathrm{BC}}$ and $\mathbf{D}_{\mathrm{I}}$ respectively, defined in [23–25] that are features of the *Interface* class objects.
- **Assembly** creates the main matrix $\mathbf{K}_{\mathrm{Corner}}$ of the corner, feature of the *Corner* class object.
- **Solver** applies the Müller method [27] to obtain λ solutions of $\det(\mathbf{K}_{\mathrm{Corner}}(\lambda))$. It saves them in the Corner object.
- **Displacement and stress fields** computes stress and displacement fields around the singular point at a distance $r = 1$ for each λ solution. It saves them in the Corner object.
- **Figures** generates a scheme of the problem and graphs of displacement and stress fields, saving them in the PDF report.
- **Send results** sends generated PDF files with the solution of the problem to the user's email address.

3 Graphical Example

In this section, we present a practical example featuring a cutting tool sliding under a semiplane with frictionless contact. Material 1 and 2 have the same elastic properties. Figure 3 presents a screenshot of the SingSol Web-App, illustrating the steps taken to solve this specific problem.

Let's remind that certain form fields become visible after a step has been completed. As mentioned earlier, the form dynamically updates as the user progresses through the completion process.

The step-by-step process is outlined below:

- **Step 1** Choose between Spanish or English language. This preference is saved for future sessions.
- **Step 2** Select between an open or closed corner. For this example, an open corner is chosen.
- **Step 3** Choose the material type from options such as isotropic, orthotropic, or transversely isotropic. In this example, orthotropic material is used for the semiplane composite laminate.
- **Step 4** Enter the wedge angle in the corner and the material properties. For orthotropic materials, specify 3 elastic moduli E_{11}, E_{22} and E_{33}, 3 shear moduli G_{12}, G_{13} and G_{23} and 3 Poisson ratios ν_{12}, ν_{13} and ν_{23}. Define the material orientation within the corner using the ω angles. Define the first material position as the upper material with angles 0° and 180°. Angles must be entered in degrees, while the unit for the elasticity and shear moduli is at the user discretion, although the use of GPa is recommended. It is crucial to maintain consistency between the chosen units.
- **Step 5** Introduce the boundary condition for the first material, in this case *Stress Free* boundary condition.
- **Step 6** Add a second material by clicking the *Add material* button.
- **Step 7** Choose the type of the second material. In this example, an orthotropic material is selected again.
- **Step 8** Enter the second material position in the corner and its properties. Define the second material position as the lower wedge material with angles 180° and 300°.
- **Step 9** Specify the interface condition between material 1 and material 2. In this case, a frictionless contact interface is chosen. Also the boundary condition of the second material is chosen in this step, in this case we choose *Stress free* boundary condition too.
- **Step 10** Optionally, introduce additional parameters for the search interval of the singularity exponent λ and the resolution for the stress and displacement computation, or use predefined values.
- **Step 11** Provide your email address together with your name, surname and institution, to receive the results.
- **Step 12** Click the *Submit* button to submit the problem for solving in the cluster and to receive the results.

Once the user submits the problem, the *.JSON file* is sent to the cluster. Depending on the complexity of the problem, results will be delivered to the user's provided email address within minutes for a simple problem or within hours for a more complex one.

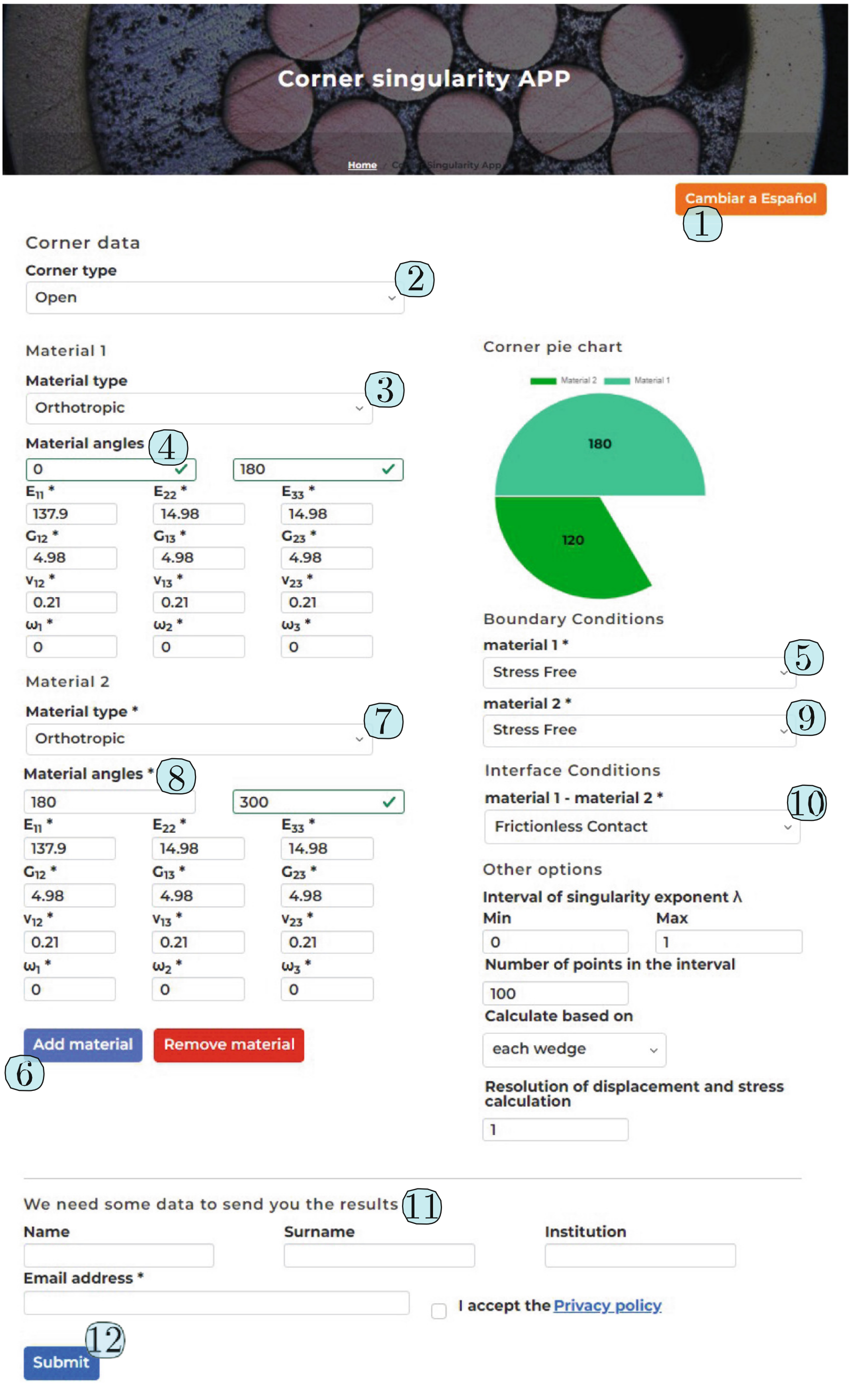

Fig. 3. SingSol Web-App: Steps to solve a problem with the frictionless contact between a wedge and a semiplane.

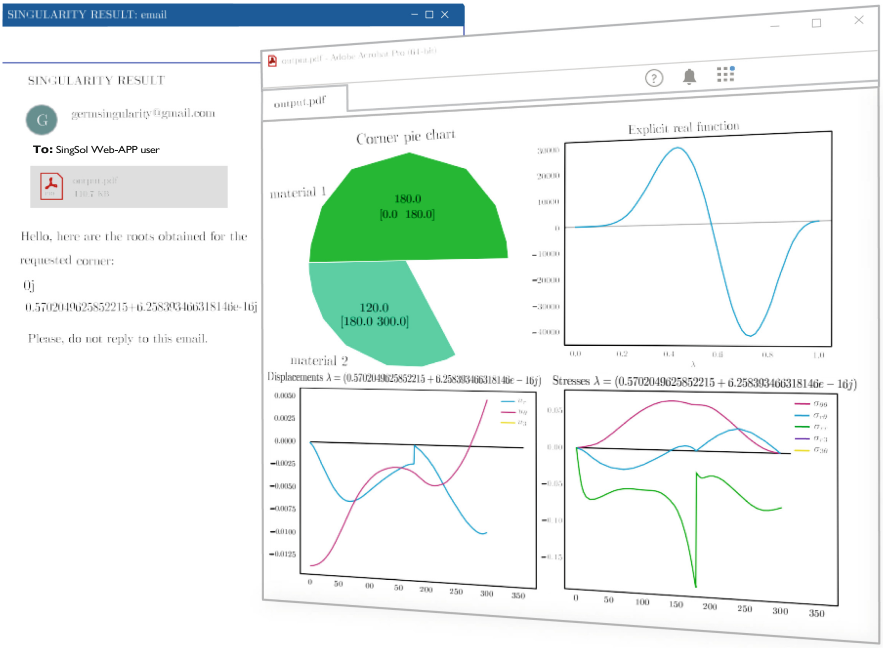

Fig. 4. Email notification: SingSol results for the solved problem.

In Fig. 4, an example of email generated as a solution for a solved problem involving a frictionless contact problem is shown.

4 Conclusions and Future Developments

To summarize, SingSol, the web application for singularity solution analysis in multi-material corners with homogeneous boundary conditions, stands as a userfriendly and versatile tool for researchers and engineers engaged in analyzing stress and strain fields in complex structural configurations. Several key points highlight its strength:

User-friendly Interface: SingSol stands out for its user-friendly and highly intuitive interface. The platform guides users through a step-by-step process, enabling them to input problem details efficiently. From choosing between an *open corner* or *closed corner* to the detailed specification of materials, boundary conditions, and interfaces, SingSol dynamically adapts the form based on user selections. These characteristics ensure that the data entry process is clear and straightforward. The user-friendly interface makes SingSol an accessible tool even for those not experts in the field, providing an eficient user experience.

Versatility in Boundary Conditions: SingSol supports a broad range of boundary conditions, offering displacement and stress fields around singular points. However, it is important to acknowledge the current limitation of addressing only frictionless contact conditions.

While SingSol's strengths lie in the handling of different scenarios, its current version lacks the capability to address cases involving frictional contact conditions. The Matlab-based code, used in the offline version, that is the base of the actual Python code back-end of this application, displays a color-coded map to the user. This visual representation aids the user in understanding the solution landscape, providing insights into both the number of solutions and their approximate values. Subsequently, these approximate values serve as the initial point in the search for solutions through a minimization approach employing least squares fitting. However, adapting this visualization-driven approach to the online, unsupervised version introduces some challenges.

As in the online version, users can not manually provide initial points for the solver when dealing with frictional contact conditions. Current efforts are focused on refining the solver module to handle frictional cases, eliminating the need for user-provided initial points. The objective is to develop a solver capable of autonomously identifying and presenting all possible solutions.

The challenge increases as the dimension of the problem increases with each additional frictional contact face introduced to the corner. While the offline version effectively employs a two-dimensional plane to represent the solution space for a single face with friction, advanced automation is necessary when multiple frictional faces are involved. Anticipated enhancements aim to optimize the analysis for cases with a single face with friction and to extend the capability of the tool to handle scenarios involving multiple faces with friction contact.

Nevertheless, despite this pointed weakness, the SingSol web application in its current state offers more versatility than most methodologies and codes currently available in the literature.

Robust and Reliable: The matrix formalism and Matlab-based code underlying SingSol have undergone rigorous verification, comparing results with various methodologies from the literature. This extensive validation process confirms the reliability of the tool.

Modular Organization: SingSol's modular structure allows seamless integration of additional modules for extended functionality. This approach ensures flexibility and facilitates the incorporation of future features and improvements.

Looking forward, we are actively working on improving the tool, and together with the need discussed above, our plans include addressing the following improvements:

Frictional Contact: As mentioned above, unlike the offline version of this tool, the web application does not include yet the ability to solve frictional contact cases. To address this weakness in the code, work is underway to improve the *solver* module of the code. New methods for the code to autonomously search for possible solutions are being introduced and tested. The final strategy chosen will be one that allows the code to autonomously find at least all the solutions that were found manually, although it is expected that some additional solutions may appear that were not detected due to the complexity of the problem, especially in cases with more than one friction surface.

File Loading Capability: Amplifying user functionality by enabling direct import of *.JSON file* into the application. This feature enables users to make modifications to previously generated *.JSON file* from similar cases, eliminating the need to input parameters manually. Furthermore, the direct import functionality allows users to explore solutions for the same problem with variations in singularity exponent interval or higher resolutions in stress and displacement parameters. This flexibility enhances the versatility of the application, addressing a broader range of user needs.

Parameterization of Variables: Incorporating advanced parameterisation capabilities to explore the effect of varying variables defining the corner problem.

These ongoing improvements and planned enhancements underline SingSol's commitment to providing a robust and user-friendly platform for the analysis of singular stress problems, addressing current challenges and anticipating future needs in the field of structural mechanics and material science.

Acknowledgements. Discussions with Prof. Luis Távara and Dr. Mar Muñoz-Reja (Universidad de Sevilla) are highly appreciated. The research was carried out with the support of the Spanish Ministry of Science, Innovation, and Universities: PID2021- 123325OB-I00, PGC2018-099197-B-I00, Consejería de Transformación Económica, Industria, Conocimiento y Universidades, Junta de Andalucía: P18-FR-1928, US-1266016, and European Regional Development Fund: PID2021-123325OB- I00, PGC2018-099197-B-I00, P18-FR-1928, US-1266016.

References

1. Williams, M.L.: Stress singularities resulting from various boundary conditions in angular corners of plates in extension. J. Appl. Mech. **19**, 526–528 (1952)
2. Vasilopoulos, D.: On the determination of higher order terms of singular elastic stress fields near corners. Numer. Math. **53**, 51–95 (1988)
3. Sinclair, G.B.: Stress singularities in classical elasticity–I: removal, interpretation and analysis, –II: Asymptotic identification. Appl. Mech. Rev. **57**, 251–297 and 385–439 (2004)
4. Bogy, D.B.: Two edge-bonded elastic wedges of different materials and wedge angles under surface tractions. J. Appl. Mech. **32**, 377–386 (1971)
5. Theocaris, P.S., Gdoutos, E.E.: Stress singularities in cracked composite fullplanes. Int. J. Fract. **13**, 763–773 (1977)
6. Dempsey, J.P., Sinclair, G.B.: On the singular behavior at the vertex of a bi-material wedge. J. Elast. **11**, 373–391 (1981)
7. Sator, C., Becker, W.: Closed-form solutions for stress singularities at plane bi- and trimaterial junctions. Arch. Appl. Mech. **82**, 643–658 (2012)
8. Comninou, M.: Stress singularity at a sharp edge in contact problems with friction. J. Appl. Math. Phys. **27**, 493–499 (1976)
9. Comninou, M.: The interface crack. J. Appl. Mech. **44**, 631–636 (1977)
10. Comninou, M.: Interface crack with friction in the contact zone. J. Appl. Mech. **44**, 780–781 (1977)
11. Arias, R., Madariaga, R., Adda-Bedia, M.: Singular elasto-static field near a fault kink. Pure Appl. Geophys. **168**, 2167–2179 (2011)
12. Mantič, V., París, F., Cañas, J.: Stress singularities in 2D orthotropic corners. Int. J. Fract. **83**, 67–90 (1997)
13. Chen, H.-P.: Stress singularities in anisotropic multimaterial wedges and junctions. Int. J. Solids Struct. **35**, 1057–1073 (1998)
14. Poonsawat, P., Wijeyewickrema, A.C., Karasudhi, P.: Singular stress fields of angle-ply and monoclinic bimaterial wedges. Int. J. Solids Struct. **38**, 91–113 (2001)

15. Chen, H.-P., Guo, Z., Zhou, X.: Stress singularities of contact problems with a frictional interface in anisotropic bimaterials. Fatigue Fract. Eng. Mater. Struct. **35**, 718–731 (2012)
16. Costabel, M., Dauge, M., Lafranche, Y.: Fast semi-analytic computation of elastic edge singularities. Comput. Methods Appl. Mech. Eng. **190**, 2111–2134 (2001)
17. Barroso, A., Mantič, V., París, F.: Singularity analysis of anisotropic multimaterial corners. Int. J. Fract. **119**, 1–23 (2003)
18. Yin, W.L.: Anisotropic elasticity and multi-material singularities. J. Elast. **71**, 263–292 (2003)
19. Hwu, C.: Anisotropic Elasticity with Matlab. Springer, New York (2021). https://doi.org/10.1007/978-3-030-66676-7
20. Papadakis, P.J., Babuska, I.: A numerical procedure for the determination of certain quantities related to the stress intensity factors in two-dimensional elasticity. Comput. Methods Appl. Mech. Eng. **122**, 69–92 (1995)
21. Lee, D., Barber, J.R.: An automated procedure for determining asymptotic elastic stress fields at singular points. J. Strain. Anal. **41**, 287–295 (2006)
22. Yosibash, Z.: Singularities in Elliptic Boundary Value Problems and Elasticity and their Connection with Failure Initiation. Springer, New York (2012). https://doi.org/10.1007/978-1-4614-1508-4
23. Mantič, V., Barroso, A., París, F.: Singular elastic solutions in anisotropic multimaterial corners. Applications to composites. In: Mantič, V. (ed.), Mathematical Methods and Models in Composites, Imperial College Press, pp. 425–495 (2014)
24. Mantič, V., Barroso, A., París, F.: Computational procedure for singularity analysis of anisotropic elastic multimaterial corners - applications to composites and their joints. In: Mantič, V. (ed.), Mathematical Methods and Models in Composites (Second Edition), World Scientific, pp. 613–695 (2023)
25. Herrera-Garrido, M.A., Mantič, V., Barroso, A.: A powerful matrix formalism for stress singularities in anisotropic multimaterial corners. Homogeneous (orthogonal) boundary and interface conditions. Theor. Appl. Fract. Mech. **119**, 103271 (2021)
26. Herrera-Garrido, M.A., Mantič, V., Barroso, A.: A semi-analytical matrix formalism for stress singularities in anisotropic multi-material corners with frictional boundary and interface conditions. Theor. Appl. Fract. Mech. **127**, 104160 (2023)
27. Muller, D.E.: A method for solving algebraic equations using an automatic computer. Math. Tables Aids Comput. **10**, 208–215 (1956)