MODELING DIFFERENT FAILURE MECHANISMS IN METALS

A Dissertation

by

LIANG ZHANG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2011

Major Subject: Mechanical Engineering

Modeling Different Failure Mechanisms in Metals

MODELING DIFFERENT FAILURE MECHANISMS IN METALS

A Dissertation

by

LIANG ZHANG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Jyhwen Wang
Committee Members,	Amine Benzerga
	Karl Hartwig
	Xin-Lin Gao
Head of Department,	Jerald Caton

December 2011

Major Subject: Mechanical Engineering

ABSTRACT

Modeling Different Failure Mechanisms in Metals. (December 2011)

Liang Zhang, B.S., University of Science and Technology of China; M.S., Texas A&M University

Chair of Advisory Committee: Dr. Jyhwen Wang

This work consists of three parts corresponding to three different failure mechanisms in metals, i.e., the localized necking in sheet metals, the bifurcation in bulk and sheet metals, and the ductile fracture induced by the void nucleation, growth, and coalescence.

The objective of the first part is to model the localized necking in anisotropic sheet metals to demonstrate that localized geometric softening at a certain stage of deformation is the main cause of localized necking. The sheet is assumed to have no initial geometric defects. Its deformation process is divided into two stages. The neck formation and evolution are considered. A novel failure criterion is proposed. The competition among different types of necks is identified. The predictions are found to fit with the experimental results well. The sheet thickness, the strain hardening behavior, and plastic anisotropy are found to affect the sheet metal formability.

The objective of the second part is to model the bifurcation in anisotropic bulk and sheet metals to couple plastic anisotropy and the strain hardening/softening behavior. The material is assumed to obey a Hill-type Drucker-Prager yield criterion along with a

non associated flow rule. The conditions for bifurcation in bulk and sheet metals are derived. The internal friction coefficient, plastic anisotropy, and other correctional terms are found to affect the onset of bifurcation. Two bifurcation modes are found to exist in sheet metals.

The objective of the third part is to derive the constitutive relations for porous metals using generalized Green's functions to better understand the micromechanism of the ductile fracture in metals. A porous metal is idealized as an isotropic, rigid-perfectly plastic matrix embedded with numerous cylindrical or spherical voids. Two types of hollow cuboid RVEs are employed corresponding to the two void shapes. The microscopic velocity fields are obtained using generalized Green's functions. The constitutive relations are derived using the kinematic approach of the Hill-Mandel homogenization theory and the limit analysis theory. The macroscopic mean stress, the porosity, the unperturbed velocity field, and the void distribution anisotropy are found to affect the macroscopic effective stress and the microscopic effective rate of deformation field.

ACKNOWLEDGEMENTS

I would like to thank my committee chair, Dr. Wang, and my committee members, Dr. Benzerga, Dr. Hartwig, and Dr. Gao, for their guidance and support throughout the course of this research.

Thanks also go to my friends and colleagues and the department faculty and staff for making my time at Texas A&M University a great experience. I also want to extend my gratitude to the National Science Foundation, which provided the research funding.

Finally, thanks to my mother and father for their encouragement and to my girlfriend for her patience and love.

TABLE OF CONTENTS

Page

Page

LIST OF FIGURES

Page

1. INTRODUCTION

Material failure plays an important role in human life. People sometimes expect its occurrence but at other times do not. As indicated by its name, material failure consists of two major parts, i.e., the material involved in it and the failure mechanism behind it. The failure mechanism is affected by many factors, such as the microstructure of the material, the specimen geometry, and the loading condition. As a result, the same material may fail by different failure mechanisms, while different materials may fail by the same failure mechanism. For example, the specimens made of the same material may fail either by microcrack propagation or by geometric softening, whereas the specimens made of different materials may all fail by microcrack propagation. This phenomenon indicates that it may be difficult to determine the failure mechanism behind a certain kind of material failure. Despite difficulties, numerous efforts have been devoted to investigating different failure mechanisms. This is because, by doing this, people can more precisely predict the failure conditions for various materials so as to develop new products, to enhance product performances, and most importantly, to save lives.

This work consists of three parts corresponding to three different failure mechanisms in metals, i.e., the localized necking in sheet metals, the bifurcation in bulk and sheet metals, and the ductile fracture induced by the void nucleation, growth, and coalescence. The first part provides an alternative approach to modeling the localized

---

This dissertation follows the style of *International Journal of Adhesion and Adhesives.*

necking in anisotropic sheet metals to demonstrate that localized geometric softening at a certain stage of deformation rather than the initial defects is the main cause of localized necking. The second part provides an applicable approach to modeling the bifurcation in anisotropic bulk and sheet metals to couple plastic anisotropy and the strain hardening/softening behavior and also to identify different bifurcation modes in sheet metals. The third part provides an alternative approach to deriving the constitutive relations for porous metals to better understand the micromechanism of the ductile fracture in metals. This work is expected to provide novel insights into several failure mechanisms.

1.1 Localized Necking in Sheet Metals

As a sheet metal is stretched into the plastic range, its deformation often becomes intensively concentrated into a narrow band. This phenomenon is referred to as localized necking. Localized necking is one of the most frequently observed failure mechanisms in many sheet forming processes. The onset of localized necking limits the sheet metal formability.

Keeler [1] and Goodwin [2] first introduced the concept of the forming limit diagram (FLD). The FLD plots the limiting major and minor strains that can be achieved along different stress or strain paths. It characterizes the sheet metal formability and is widely used in industrial practices. Much effort has been devoted to experimentally determining the FLDs of sheet metals of different sheet thicknesses and material

properties [3-10]. However, it is expensive and time consuming to conduct a large number of experiments. There is a need to theoretically predict the FLDs of sheet metals of different sheet thicknesses and material properties.

Hill [11] idealized localized necking as a velocity discontinuity and found that the necks always form along the zero extension directions. Since the zero extension directions do not exist on the right-hand side of the FLD, he only predicted half of the FLD. Swift [12] suggested that instability occurs when the principal stresses attain their maximum and predicted the critical strains for diffuse necking. Although he predicted the entire FLD, his predictions on the right-hand side, together with Hill's predictions, are often used to plot the entire FLD.

Stören and Rice [13] idealized localized necking as a bifurcation from a uniform or smoothly varying deformation field and proposed a bifurcation theory for sheet metals to predict the entire FLD. They suggested that the subsequent yield surfaces of the material would develop a vertex-like structure during continued deformation and found that the vertex-like structure affects the sheet metal formability. Several authors [14, 15] extended their work by taking into account either various yield criterions or some special material behaviors (e.g., strain softening). According to the bifurcation theory, localized necking is rarely related to localized geometric softening. However, its onset is always observed to be accompanied with localized geometric softening. It is hereby questionable whether the pointed vertex on the smooth yield locus is the main cause of localized necking.

Marciniak and Kuczynski [16] proposed the most widely used model, which is referred to as the M-K model, to predict the right-hand side of the FLD. They suggested that it is the initial geometric defects that cause material failure. They idealized a defect as a groove thinner than its neighboring region and assumed that the groove lies perpendicular to the major strain direction. They modeled the groove evolution and calculate the limiting strains at which the groove extends much more quickly than its neighboring region in its width direction. Marciniak et al. [17] later incorporated a rate-dependent behavior into the model and found that this behavior can affect the groove evolution. Hutchinson and Neale [18, 19] improved the M-K model by enabling it to also predict the left-hand side of the FLD. They suggested that the groove does not necessarily lie perpendicular to the major strain direction but can lie inclined by an angle with respect to the major strain direction. They determined the real neck orientation by finding the groove orientation minimizing the major limiting stain. Several authors [20-23] further validated this approach. Barata da Rocha et al. [20] also found that, for anisotropic materials, the groove does not always lies perpendicular to the major stress direction even on the right-hand side of the FLD.

The M-K model provides a relatively simple description of the mechanism of localized necking. It also brings in the important concept of the initial geometric defects because the defects are unavoidable in real materials. However, several authors [24-26] found that the defect sizes measured in the test specimens are much smaller than those selected to fit the predictions to the experimental results. It is hereby questionable whether the initial geometric defects are the main cause of localized necking. Tadros and

Mellor [27] suggested that it is the instable deformation that enables the groove evolution and improved the M-K model by letting the groove start to evolve not at the very beginning of the deformation process but at the onset of diffuse necking. Tadros and Mellor [26] also conducted a series of in-plane stretching tests on sheet metals with premade aligned grooves to validate their hypothesis. They observed that the necks tend to form perpendicular to the major strain direction no matter whether the grooves lie perpendicular or parallel to the major strain direction. This observation again leads one to question whether the initial geometric defects may not be the main cause of localized necking.

Several authors implemented different yield criterions into the M-K model to incorporate special material properties and found that the predictions are quite sensitive to the selection of the yield criterion. Sowerby and Duncan [28] implemented the quadratic Hill criterion into the M-K model to incorporate plastic anisotropy and found overestimates in the predictions. Several authors [4, 29-35] implemented different nonquadratic yield criterions into the M-K model and found their predictions to better fit the experimental results. Meanwhile, several authors [21, 29, 36-41] employed a Taylor type model to characterize the microstructure of polycrystalline, simulated the texture evolution during continued deformation, and found that their predictions also fit the experimental results well. Although nowadays the nonquadratic yield criterions are more favorable, the quadratic yield criterions are still often employed to examine the validity of new models.

Keeler and Brazier [7] observed that, for steel sheets, the limiting major strain under plane strain conditions increases with increasing sheet thickness. They also provided empirical equations to approximate this effect. Smith and Lee [9] observed that this effect also exists in aluminum sheets but is not as prominent as that in steel sheets. Wilson et al. [42, 43] observed that the sheet thickness effect also exists in copper and brass sheets. In industries, this sheet thickness effect enables people to increase the sheet metal formability or to reduce the weight and cost of sheet metal products by varying the sheet thickness. Therefore, it is of great practical value to theoretically predict the sheet thickness effect.

Recently, several authors studied localized necking in different ways. Kuwabara et al. [44] incorporated the strain path change into the M-K model and found the sheet metal formability is path-dependent. Korkolis and Kyriakides [45] later experimentally demonstrated this finding. Eyckens et al. [46, 47] and Allwood and Shouler [48] incorporated the out-of-plane normal and shear stresses into the M-K model and found these stresses may help to enhance the sheet metal formability. Wu et al. [49] incorporated the hydrostatic pressure into the M-K model and drew a similar conclusion. They all found these stresses may help to enhance the sheet metal formability. Stoughton [50] and Stoughton and Yoon [51, 52] introduced the concept of the stress-based FLD and also proposed the associated stress-based approach to surface cracking, and etc.. Aretz [53] proposed a simple isotropic-distortional hardening model and applied it to predict the onset of localized necking in orthotropic sheet metals, while Kobayashi [54]

also proposed an acceleration wave theory to predict the onsets of diffuse and localized necking.

1.2 Bifurcation in Bulk and Sheet Metals

As a ductile solid is deformed into the plastic range, its deformation often becomes intensively concentrated into a narrow band. The examples of localization of deformation include the Lüders bands in metals, the rock faults in marble and sandstone, the shear bands in soils, and etc. [55]. Such localization is one of the most frequently observed failure mechanisms in engineering structures. It is hereby of great significance to theoretically predict the onset of localization. The onset of localization is believed to be affected the by the defects in the material. Unfortunately, it is almost impossible to accurately represent these defects. However, it has been proven to be doable to idealize an imperfect material as homogeneous on a large scale and also to treat localization occurs as a bifurcation from uniform deformation [56-59]. The problem can hereby be simplified as seeking for the conditions for the onset of bifurcation.

Several authors [56-59] constructed the fundamental theory of bifurcation. Hill [56] and Thomas [59] proposed an acceleration wave theory to predict the onsets of bifurcation in dynamically deformed solids, while Rudnicki and Rice [57, 58] proposed an alternative theory to predict the onsets of bifurcation in quasistatically deformed solids. Rudnicki and Rice [57, 58] solved the problem as follows: first, they proposed the constitutive relations including a tangent tensor relating the stress rates to the rates of

deformation; second, they idealized bifurcation as a singularity in the uniform rate of deformation field; third, they obtained the conditions for the onset of bifurcation as a function of the tangent tensor. Their works provided a solid framework for modeling bifurcation and enable the successive researchers to implement more elaborate considerations into the model.

Several authors [55, 57, 58, 60] found that the constitutive relations for the material can significantly affect the onset of bifurcation. Specifically, plastic anisotropy, plastic non-normality, dilatancy, and the strain hardening behavior are found to all affect the onset of bifurcation. Rudnicki and Rice [57, 58] assumed that the material exhibits a plastically isotropic, dilatant, non-linear hardening behavior and also suggested that the subsequent yield surfaces of the material would develop a vertex-like structure during the continued deformation. They found these factors differently affect the onset of bifurcation. Steinmann et al. [60] later incorporated plastic anisotropy into the model. They found plastic anisotropy can also affect the onset of bifurcation. However, they assumed the material to be rigid-perfectly plastic. Therefore, there is a need for a model truly conjugating the effects of plastic anisotropy and the strain hardening behavior. Borja [55] assumed different materials obeying different yield criterions (i.e., the Drucker-Prager criterion [61], the Labe-Ducan criterion [62], and the Matusuoka-Nakai criterion [63]) along with the corresponding non-associated flow rules. He also found that different yield criterions and non-associated flow rules can also affect the onset of bifurcation.

Earlier works on the bifurcation theory primarily focused on the geometrically linear case corresponding to small deformation. Ogden [64] first found that geometric nonlinearity also plays an important role in the bifurcation theory. In fact, geometric nonlinearity has a profound effect on the onset of bifurcation. Specifically, if the stresses are comparable to the tangent modulus, geometric nonlinearity tends to cause the tangent tensor to be asymmetric and hereby to promote the onset of bifurcation [55]. Several authors [55, 65-67] incorporated geometric nonlinearity into the bifurcation theory using the finite deformation elasto-plasticity theory. Armero and Garikipati [65] first derived the unregularized conditions for the onset of bifurcation. Larsson and co-workers [66, 67] later derived the regularized ones. However, these two groups of authors obtained different predictions of the onset of bifurcation. Borja [55] pointed out that the discrepancy arises from the constitutive relations.

Stören and Rice [13] proposed the bifurcation theory for sheet metals. They took into account the restrictions introduced by plane stress conditions and also suggested a vertex-like structure of the yield surface. They succeeded in plotting the forming limit diagrams of the sheet metals. Similarly to Rice and Rudncki [58], they found that the vertex structure can affect the sheet metal formability. Several authors [14, 15] extended their work by taking into account either various yield criterions or some special material behaviors, e.g., strain softening. However, these authors all required the bifurcation band to only lie in the sheet plane. Further investigation in this part will show that this is not realistic.

1.3 Deformation of Porous Metals

Ductile fracture in metals is often caused by the void nucleation, growth, and coalescence. As a metal is deformed into the plastic range, numerous voids tend to nucleate around the second-phase particles by the particle cracking and debonding [68]. The metallic matrix flow is closely connected with the void evolution because: first, it plays an important role in the particle cracking and debonding and hereby affects the void nucleation; second, it also accounts for dilatation; last, it even has an impact on the ligament growth and breakage. Therefore, studying the micromechanics of the matrix helps to predict ductile fracture in metals. It is doable to either experimentally monitoring or numerically simulating the matrix deformation. However, it is expensive and time consuming to conduct a large number of experiments or simulations. Therefore, there is a need to predict the matrix flow and thereafter to derive the constitutive relations for porous metals.

The micromechanical approach is the most widely used approach to predicting the matrix flow. Several earlier authors [69-73] first studied the growth of an isolated void in an infinite medium to evaluate the effects of the void shape and the matrix properties. McClintock [71] first assumed an elliptic cylindrical void under generalized plane strain conditions. Tracey [73] also assumed a cylindrical void under generalized plane strain conditions. Rice and Tracey [72] then assumed a spherical void subject to combined uniaxial and hydrostatic extension. Budiansky et al. [69] and Lee and Mear [70] later assumed an elliptical void subject to axisymmetric extension. In addition,

McClintock [71], Tracey [73], and Rice and Tracey [72] assumed an isotropic perfectly plastic medium, while Budiansky et al. [69] and Lee and Mear [70] assumed a viscous nonlinear medium. These works provided valuable insights into the micromechanics of the problem and also prepared the successive researchers for studying the micromechanics of porous metals.

Gurson [74] proposed the most widely used model, which is referred to as the Gurson model, to study the micromechanics of porous metals. He employed a hollow cylindrical and a hollow spherical RVE to represent to typical properties of porous metals with cylindrical and spherical voids, respectively. He also assumed isotropic perfectly plastic subject to axisymmetric extension. He succeeded in deriving the macroscopic yield criterions and the porosity evolution laws for porous metals. Leblond [75] later found that his derivation is amendable to the kinematic approach of the Hill-Mandel [76, 77] homogenization theory and the limit analysis [78] of the selected RVE subject arbitrary extension. Leblond [75] also demonstrated that, for the selected RVE, the Gurson model provides a rigorous upper bound.

Several authors [79-85] incorporated the void nucleation, growth, and coalescence into the Gurson model. Tvergaard [84, 85] introduced a parameter with a value of 1.5 to account for the void interactions based on his numerical studies. Needleman and Tvergaard [82] later proposed a phenomenological description of the void coalescence including two parameters, i.e., the critical porosity and the acceleration rate, to account for the fracture behavior. Several authors [79-81, 83] further looked into the micromechanism of the void coalescence and quantified the effect of the

microstructure on the two aforementioned parameters. These improvements enable the Gurson-type models to provide more accurate predictions.

Several authors [79, 80, 86-93] also incorporated the void shape effect and the plastic anisotropy of the matrix into the Gurson model. Gologanu et al. [88-90] and Garajeu [87] assumed either spherical or ellipsoidal voids, while Benzerga and Besson [86] assumed an plastic anisotropic matrix. They found that the void shape effect and plastic anisotropy both affect the macroscopic yield criterions and the porosity evolution laws. Benzerga et al. [79, 80] then incorporated both initial and induced anisotropy and also combined the void shape effect and plastic anisotropy. Monchiet et al. [93] and Keralavarma and Benzerga [91] later derived different sets of macroscopic yield criterions truly coupling the void shape effect and plastic anisotropy. Keralavarma and Benzerga [92] recently derived a new set of macroscopic yield criterions and porosity evolution laws applicable to more general deformations. These improvements enable the Gurson-type models to have broader applications.

1.4 Research Objectives

The objective of Section 2 is to model the localized necking in anisotropic sheet metals to demonstrate that the main cause of localized necking is localized geometric softening at a certain stage of deformation rather than the initial defects. The quadratic Hill yield criterion is employed to characterize plastic anisotropy. The sheet is assumed to have no initial geometric defects. The deformation process is divided into two stages.

A Considère-type criterion is proposed to determine the critical strains for a neck to form. An energy-based hypothesis is proposed to quantify the defect ratio at the neck formation. The neck evolution is considered. A novel failure criterion is proposed. Two types of necks are found to be most competitive to cause material failure during continued deformation. The forming limit curves are hereby found to exhibit different characteristics in different region. The predicted forming limit curve of 2036-T4 aluminum is found to fit the experimental results well. The sheet thickness, the strain hardening behavior, and plastic anisotropy are found to affect the sheet metal formability. More realistic yield criterions and strain hardening behaviors can be implemented into the proposed model. This part provides an alternative approach to modeling the localized necking in anisotropic sheet metals.

The objective of Section 3 is to model the bifurcation in anisotropic bulk and sheet metals to couple plastic anisotropy and the strain hardening/softening behavior. The material is assumed to obey a Hill-type Drucker-Prager yield criterion along with a non associated flow rule. The constitutive relations for the material under consideration are derived. The conditions for bifurcation in bulk metals proposed by Rudnicki and Rice are employed, while the conditions for bifurcation in sheet metals are obtained with consideration of the restrictions imposed by the stress resultant equilibrium. Different bifurcation modes in sheet metals are identified. The critical tangent modulus at the onset of bifurcation and the corresponding bifurcation band orientation are obtained. A series of so-called bifurcation curves are generated to represent to which extent a metal can be deformed without causing the onset of bifurcation. The effects of the internal

friction coefficient, plastic anisotropy, the terms introduced by the co-rotational stress rates, and the terms introduced by the stress resultant equilibrium on the onset of bifurcation are evaluated.

The objective of Section 4 is to derive the constitutive relations for porous metals using generalized Green's functions to better understand the micromechanism of the ductile fracture in metals. The porous metals are assumed to consist of an isotropic, rigid-perfectly plastic matrix and numerous periodically distributed voids and to be subject to non-equal biaxial or triaxial extension. Two types of hollow cuboid RVEs are employed represent the typical properties of porous metals with cylindrical and spherical voids. The microscopic velocity fields are obtained using generalized Green's functions. The constitutive relations are derived using the kinematic approach of the Hill-Mandel homogenization theory and the limit analysis theory. The effects of the macroscopic mean stress, the porosity, the unperturbed velocity field, and the void distribution anisotropy on the macroscopic effective stress and the microscopic effective rate of deformation field are evaluated. The potential applications of the predicted microscopic effective rate of deformation field are addressed. The results obtained using the cuboid RVEs are compared with those obtained using the non-cuboid RVEs.

2. MODELING THE LOCALIZED NECKING IN ANISOTROPIC SHEET METALS

2.1 Theoretical Considerations

2.1.1 Yield Criterion and Constitutive Relations

For plastically orthotropic, isotropic strain hardening materials, the yield function takes the form of

$$F\left(\sigma_e, \epsilon_e^p\right) = \sigma_e - \sigma_Y\left(\epsilon_e^p\right) = 0, \tag{2.1}$$

where σ_e denotes the Hill effective stress; ϵ_e^p denotes the effective plastic strain; σ_Y denotes the current uniaxial strength increasing with increasing ϵ_e^p. Assume that the material exhibits a rate-dependent strain hardening behavior that can be represented by

$$\sigma_Y = K\dot{\epsilon}_e^m \epsilon_e^n. \tag{2.2}$$

Following Benzerga and Besson [86], σ_e can be expressed as

$$\sigma_e^2 = \frac{3}{2}\sigma'_{ij}\mathscr{A}_{ijkl}\sigma'_{kl}, \tag{2.3}$$

where $(\cdot)'$ denotes the deviatoric tensor and $\mathscr{A}_{ijkl}$ denotes the fourth-order anisotropy tensor. $\mathscr{A}_{ijkl}$ can be expressed in the Voigt notation as

$$(A_{IJ})_{I,J=\mathrm{I},\ldots,\mathrm{VI}} = \mathscr{A}_{ijkl} \quad \text{and} \quad (A_{IJ})_{I\neq J} = 0. \tag{2.4}$$

The effective plastic strain increment, $d\epsilon_e^p$, can be expressed as [86]

$$d\epsilon_e^{p2} = \frac{2}{3}d\epsilon_{ij}^p\hat{\mathscr{A}}_{ijkl}d\epsilon_{kl}^p, \tag{2.5}$$

where $\hat{\mathscr{A}}_{ijkl}$ denotes the formal inverse of $\mathscr{A}_{ijkl}$. $\hat{\mathscr{A}}_{ijkl}$ is related to $\mathscr{A}_{ijkl}$ by [86]

$$\hat{A}_{\mathrm{I,I}} = \frac{-A_{\mathrm{I,I}} + 2A_{\mathrm{II,II}} + 2A_{\mathrm{III,III}}}{A_{\mathrm{I,I}}A_{\mathrm{II,II}} + A_{\mathrm{II,II}}A_{\mathrm{III,III}} + A_{\mathrm{III,III}}A_{\mathrm{I,I}}},$$
$$\hat{A}_{\mathrm{II,II}} = \frac{2A_{\mathrm{I,I}} - A_{\mathrm{II,II}} + 2A_{\mathrm{III,III}}}{A_{\mathrm{I,I}}A_{\mathrm{II,II}} + A_{\mathrm{II,II}}A_{\mathrm{III,III}} + A_{\mathrm{III,III}}A_{\mathrm{I,I}}},$$
$$\hat{A}_{\mathrm{III,III}} = \frac{2A_{\mathrm{I,I}} + 2A_{\mathrm{II,II}} - A_{\mathrm{III,III}}}{A_{\mathrm{I,I}}A_{\mathrm{II,II}} + A_{\mathrm{II,II}}A_{\mathrm{III,III}} + A_{\mathrm{III,III}}A_{\mathrm{I,I}}},$$
$$\hat{A}_{\mathrm{IV,IV}} = \frac{1}{A_{\mathrm{IV,IV}}}, \quad \hat{A}_{\mathrm{V,V}} = \frac{1}{A_{\mathrm{V,V}}}, \quad \hat{A}_{\mathrm{VI,VI}} = \frac{1}{A_{\mathrm{VI,VI}}}. \tag{2.6}$$

Eqs. (2.3) and (2.5) provide an appropriate measure of work equivalence that can be expressed as

$$\sigma_{ij} d\epsilon^p_{ij} = \sigma_e d\epsilon^p_e. \tag{2.7}$$

The R-value is widely used to characterize the plastic anisotropy of sheet metals. It is defined as the mean value of the ratios of the plastic strain in the width direction to that in the thickness direction. The R-values are generally obtained from the uniaxial tensile tests on specimens cut at angels $0°$, $45°$ and $90°$ to the rolling direction. An average R-value is often used and can be defined as

$$R = \frac{R_0 + 2R_{45} + R_{90}}{4}. \tag{2.8}$$

If the orthotropic axes are set to coincide with the principal stress and strain directions, the crossed stress and strain components vanish. The components $(A_{IJ})_{I,J\geq \mathrm{IV}}$ hereby do not appear. $(A_{IJ})_{I,J=\mathrm{I,II,III}}$ can be expressed in terms of R_0 and R_{90} as [86]

$$A_{\mathrm{I,I}} = \frac{2}{3}\frac{-R_0R_{90} + 2R_0 + 2}{R_0 + 1}, \quad A_{\mathrm{II,II}} = \frac{2}{3}\frac{2R_0R_{90} + 2R_0 - 1}{R_0 + 1},$$
$$A_{\mathrm{III,III}} = \frac{2}{3}\frac{2R_0R_{90} - R_0 + 2}{R_0 + 1}. \tag{2.9}$$

The associated flow rule can be expressed in terms of $\mathscr{A}_{ijkl}$ as [86]

$$d\epsilon_{ij}^{p} = \frac{3}{2}\frac{d\epsilon_{e}^{p}}{\sigma_{Y}}\left(\mathscr{A}_{ijkl}\sigma_{kl}^{\prime}\right)^{\prime}. \tag{2.10}$$

Eq. (2.10) can be formally inverted to

$$\sigma_{ij}^{\prime} = \frac{2}{3}\frac{\sigma_{Y}}{d\epsilon_{e}^{p}}\left(\hat{\mathscr{A}}_{ijkl}d\epsilon_{kl}^{p}\right)^{\prime}. \tag{2.11}$$

Eq. (2.11) can be rewritten in terms of the principal components as

$$\sigma_{I}^{\prime} = \frac{2}{3}\frac{\sigma_{Y}}{d\epsilon_{e}^{p}}\left(\hat{A}_{II}d\epsilon_{I}^{p}\right)^{\prime} \quad (\text{no sum on } I). \tag{2.12}$$

Assume the out-of-plane stresses to vanish. This implies that $\sigma_{\mathrm{III}} = 0$ and $\sigma_{m} = -\sigma_{\mathrm{III}}^{\prime}$, where σ_{m} denotes the mean stress. Eq. (2.12) can then be further modified as

$$\sigma_{\mathrm{I}} = \frac{2}{9}\frac{\sigma_{Y}}{d\epsilon_{e}^{p}}\left[2\hat{A}_{\mathrm{I,I}} - \rho\hat{A}_{\mathrm{II,II}} + 4\left(1+\rho\right)\hat{A}_{\mathrm{III,III}}\right]d\epsilon_{\mathrm{I}}^{p}, \tag{2.13}$$

$$\sigma_{\mathrm{II}} = \frac{2}{9}\frac{\sigma_{Y}}{d\epsilon_{e}^{p}}\left[-\hat{A}_{\mathrm{I,I}} + 2\rho\hat{A}_{\mathrm{II,II}} + 4\left(1+\rho\right)\hat{A}_{\mathrm{III,III}}\right]d\epsilon_{\mathrm{I}}^{p}. \tag{2.14}$$

where $\rho = d\epsilon_{\mathrm{II}}^{p}/d\epsilon_{\mathrm{I}}^{p}$ denotes the plastic strain ratio. Eq. (2.5) can also be rewritten as

$$d\epsilon_{e}^{p} = \sqrt{\frac{2}{3}\left[\hat{A}_{\mathrm{I,I}} + \rho^{2}\hat{A}_{\mathrm{II,II}} + \left(1+\rho\right)^{2}\hat{A}_{\mathrm{III,III}}\right]}d\epsilon_{\mathrm{I}}^{p}. \tag{2.15}$$

σ_{e} can then be expressed as

$$\sigma_{e} = \frac{\sigma_{Y}}{d\epsilon_{e}^{p}}\sqrt{\frac{2}{3}\left[\hat{A}_{\mathrm{I,I}} + \rho^{2}\hat{A}_{\mathrm{II,II}} + \left(1+\rho\right)^{2}\hat{A}_{\mathrm{III,III}}\right]}d\epsilon_{\mathrm{I}}^{p}. \tag{2.16}$$

2.1.2 Overview

As mentioned above, there are debates on whether the initial geometric defects are the main cause of localized necking. In this paper, an attempt is made to demonstrate whether localized geometric softening at a certain stage of deformation is the main cause of localized necking. For this reason, assume the sheet to have no initial geometric defects. Accordingly, let the neck formation be subject to a Considère-type criterion, that is, as the load acting on a certain sheet cross section attains its maximum, the sheet will not be able to support a higher load along this cross section, and a neck hereby forms. It can be seen that the necks can only form after the sheet has been deformed over certain strain.

Here the neck is idealized as a groove thinner than its neighboring region. Fig. 2.1 schematically illustrates different neck configurations. Further investigation will show that the critical major strain for a set of necks to form is a function of the strain ratio and the neck orientation. Here the term “a set of necks” means that all the necks of the same set are of the same neck orientation. Specifically, as the sheet is deformed along a proportional strain path, the first set of necks forms perpendicular to the major strain direction at $\epsilon_1 = n$ (see Fig. 2.1 (c) and (d)), while the other sets of necks form along other orientations at $\epsilon_1 > n$ (see Fig. 2.1 (a) and (b)). As a result, as the sheet is deformed at $\epsilon_1 > n$, multiple sets of necks coexist. To reduce the complexity of derivations, the necks are assumed to be so far apart from each other that the interaction

between any two necks is negligible. This assumption ensures that different necks can form and evolve independently.

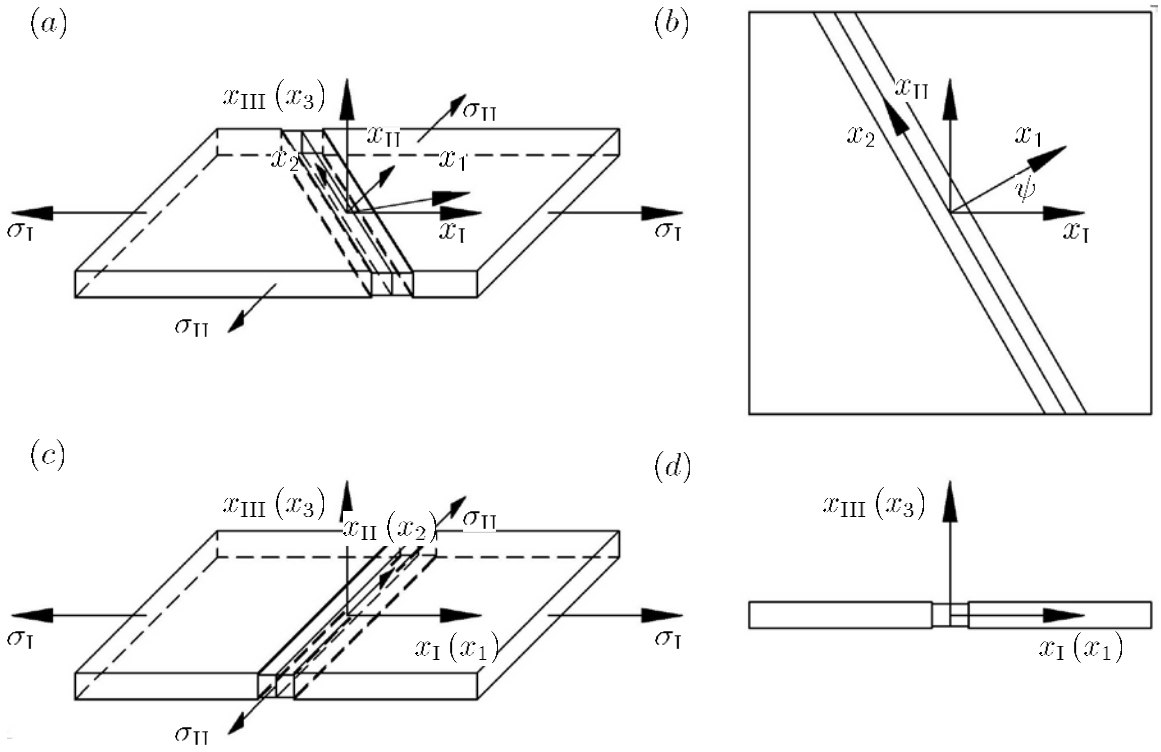

Fig. 2.1. Sketch of different neck configurations: (a) $\psi \neq 0$; (b) the top view of (a); (c) $\psi = 0$; (d) the side view of (c).

As the sheet becomes more and more deformed, different sets of necks form successively along different orientations and evolve simultaneously. This process lasts till the failure criterion is first met in one set of necks. After failure occurs, the sheet becomes unstressed, while all the neck evolution ceases. The limiting strains and the real neck orientations can be subsequently obtained. During this process, different sets of necks can be treated as competing with each other to cause material failure.

It now turns out that the deformation process can be divided into two stages: at the first stage, the sheet is uniformly deformed till the first set of necks forms perpendicular to the major strain direction at $\epsilon_{\mathrm{I}} = n$; at the second stage, different sets of necks form successively along different orientations, evolve simultaneously, and compete with each other to cause material failure during continued deformation till the failure criterion is first met in one set of them. It is worth notice that this process enables the competition among different set of necks. The numerical results will show that this competition causes the resulting forming limit curves to exhibit some special characteristics.

2.1.3 Critical Strains for a Neck to Form

Consider the neck configurations as shown in Fig. 2.1. Introduce a Cartesian coordinate system x_I ($I = \mathrm{I}, \mathrm{II}, \mathrm{III}$) with the x_{I} and x_{II} axes parallel to the major and minor strain directions, respectively, and also introduce a Cartesian coordinate system x_i ($i = 1, 2, 3$) with the x_1 and x_2 axes parallel to the neck width and length directions, respectively.

Let n_I denote the unit normal vector to the sheet cross section, and let P_n, A_n and σ_n denote the corresponding load, cross sectional area and traction, respectively. According to the aforementioned Considère-type neck formation criterion, as the load acting on a sheet cross section attains its maximum, a neck forms, that is,

$$dP_n = \sigma_n dA_n + A_n d\sigma_n = 0. \tag{2.17}$$

For large deformation, the elastic strain increments are negligible. The total strain increments can hereby be approximated as the plastic strain increments, i.e., $d\epsilon_{ij} \approx d\epsilon_{ij}^{p}$. Eq. (2.17) can then be rewritten as

$$\frac{d\sigma_n}{\sigma_n} = -\frac{dA_n}{A_n} = -d\epsilon_{22} - d\epsilon_{33}. \tag{2.18}$$

Let ψ_1 denote the angle between the x_1 and x_1 axes at the neck formation. $d\epsilon_{22}$ and $d\epsilon_{33}$ can then be expressed in terms of ψ_1 and $d\epsilon_e$ as

$$d\epsilon_{22} = \frac{\sin^2\psi_1 + \rho\cos^2\psi_1}{\sqrt{\frac{2}{3}\left[\hat{A}_{\mathrm{I,I}} + \rho^2\hat{A}_{\mathrm{II,II}} + (1+\rho)^2\hat{A}_{\mathrm{III,III}}\right]}} d\epsilon_e, \tag{2.19}$$

$$d\epsilon_{33} = d\epsilon_{\mathrm{III}} = -\frac{1+\rho}{\sqrt{\frac{2}{3}\left[\hat{A}_{\mathrm{I,I}} + \rho^2\hat{A}_{\mathrm{II,II}} + (1+\rho)^2\hat{A}_{\mathrm{III,III}}\right]}} d\epsilon_e. \tag{2.20}$$

Let the sheet be deformed along a proportional strain path. This implies that σ_n is proportional to σ_e. Substituting Eqs. (2.19) and (2.20) into Eq. (2.18) and replacing σ_n by σ_e give

$$\frac{d\sigma_e}{\sigma_e} = \frac{\cos^2\psi_1 + \rho\sin^2\psi_1}{\sqrt{\frac{2}{3}\left[\hat{A}_{\mathrm{I,I}} + \rho^2\hat{A}_{\mathrm{II,II}} + (1+\rho)^2\hat{A}_{\mathrm{III,III}}\right]}} d\epsilon_e. \tag{2.21}$$

Also assume that the strain rate is constant. Therefore,

$$d\sigma_Y = nK\dot{\epsilon}_e^m\epsilon_e^{n-1}d\epsilon_e. \tag{2.22}$$

Substituting Eqs. (2.2) and (2.22) into Eq. (2.21) gives the critical effective strain for the neck to form as

$$(\epsilon_e)_1 = \frac{\sqrt{\frac{2}{3}\left[\hat{A}_{\mathrm{I,I}} + \rho^2\hat{A}_{\mathrm{II,II}} + (1+\rho)^2\hat{A}_{\mathrm{III,III}}\right]}}{\cos^2\psi_1 + \rho\sin^2\psi_1} n, \tag{2.23}$$

where the subscript 1 denotes the quantity value at the neck formation. The corresponding critical major strain can then be obtained as

$$(\epsilon_{\mathrm{I}})_1 = \frac{n}{\cos^2\psi_1 + \rho\sin^2\psi_1}. \tag{2.24}$$

If the neck is required to only form along the zero extension direction, that is,

$$\tan\psi_1 = \sqrt{-\rho} \quad (\rho \leq 0)\,, \tag{2.25}$$

Eq. (2.24) reduces to

$$(\epsilon_{\mathrm{I}})_1 = \frac{n}{1+\rho}, \tag{2.26}$$

which is the expression for the limiting major strain predicted by Hill [94].

It is worth notice that the Considère-type neck formation criterion is more general than the Considère criterion. According to the Considère criterion, the neck orientation is known a priori, and $(\epsilon_{\mathrm{I}})_1$ is hereby independent of ψ_1. In fact, Tadros and Mellor [27] have already suggested that the necks lie perpendicular to the major strain direction and start to evolve at the onset of diffuse necking However, here the neck orientation is set to be variable, and $(\epsilon_{\mathrm{I}})_1$ is hereby a function of ψ_1. This enables different sets of necks to form successively along different orientations, to evolve simultaneously, and to compete with each other to cause material failure during continued deformation. Further investigation will also show that the necks causing material failure do not always lie perpendicular to the major strain direction.

2.1.4 Defect Ratio at the Neck Formation

As mentioned above, the neck is idealized as a groove thinner than its neighboring region. The defect ratio, f, characterizes the neck size. It is defined as the ratio of the thickness within the neck to that out of the neck. The defect ratio at the neck formation, f_1, hereby characterizes the neck size at its formation. According to Hosford and Caddell [95], the necks are generally undetectable at their formation and can be detected only after they have evolved over certain strain. This implies that f_1 should be close to 1. The initial defect ratio, f_0, in the M-K model is similar to f_1 here. It is generally assumed to be constant to represent a constant initial defect size. However, since here the neck forms at a certain stage of the deformation process, f_1 should no longer be constant. In this section, an energy-based hypothesis will be proposed to quantify f_1. Based on this hypothesis, f_1 will be found to be a function of ρ, ψ_1, and the initial sheet thickness, t_0.

Let the superscripts a and b denote the quantities out of and within the neck, respectively. Assume that, as ϵ_e^a increases from $(\epsilon_e)_1$ to $(\epsilon_e)_1 + d\epsilon_e^a$, ϵ_e^b increases from $(\epsilon_e)_1$ to $(\epsilon_e)_1 + d\epsilon_e^b$ due to localized geometric softening, where $d\epsilon_e^a$ is a small positive constant. Also let $\Delta(\cdot)$ denote the difference between the quantities within and out of the neck, e.g., $\Delta d\epsilon_e = d\epsilon_e^b - d\epsilon_e^a$. f_1 can hereby be expressed as

$$f_1 = 1 - \Delta d\epsilon_{33}. \tag{2.27}$$

Note that $\Delta d\epsilon_{33} < 0$. f_1 is hereby less than 1. Let the superscript $*$ denote the quantity value at $\rho = 0$, and also let the superscript $**$ denote the quantity value at $\rho = 0$ and $\psi_1 = 0$. Eq. (2.27) can then be rewritten as

$$f_1 = 1 - \chi_1 \left(1 - f_1^{**}\right), \tag{2.28}$$

where $\chi_1 = \Delta d\epsilon_{33}/\Delta d\epsilon_{33}^{**}$ and f_1^{**} is the value of f_1 at $\rho = 0$ and $\psi_1 = 0$. Here f_1^{**} is employed as a referential parameter. Further investigation will show that it can be obtained from the experimental results. As mentioned above, f_1 be a function of ρ, ψ_1, and t_0. As both ρ and ψ_1 are fixed, f_1^{**} seems to be only a function of t_0. This, however, is not true. In fact, f_1^{**} also integrates the effects of the internal defects and the crystal structure on the neck size at its formation. For example, real limiting major strains under plane strain conditions have been observed to be much greater than n as predicted by Hill [94] for steel sheets but to be close to n for aluminum sheets [7, 9]. This indicates that the values of f_1^{**} for steel sheets should be closer to 1 compared with those for aluminum sheets. However, it is beyond the scope of this part to quantify the effect of the internal defects and the crystal structure on f_1^{**}.

Here $\Delta d\epsilon_{33}$ and hereby χ_1 will be obtained using an energy-based hypothesis. Consider the region of $-l/2 \leq x_1, x_2 \leq l/2$ in the $x_1x_2x_3$ coordinates in the neighborhood of the neck (see Fig. 2.1). Let t denote the current sheet thickness, and assume that $l \gg t$. According to Hosford and Caddell [95], the neck width generally roughly equals the neck thickness. For simplicity, assume the neck width to be the same as the neck thickness.

Since here the neck is idealized as a groove, the hardening law and the equilibrium equations cannot be satisfied simultaneously. This implies that the strain increments can only be approximated. Note that $\Delta d\epsilon_{33}$ is very small. This leads one to approximate ϵ_e^a as increasing first by $d\epsilon_e^a$ and then by $\Delta d\epsilon_e$ and also to assume $\sigma_e^b \approx \sigma_e^a \approx (\sigma_e)_1$ and $l \approx l_1$. The total strain energy in the region takes the form of

$$U = \int_V \left(\int_0^{\epsilon_e} \sigma_e d\epsilon_e \right) dV = \int_V W dV, \tag{2.29}$$

where W denotes the strain energy density and V denotes the volume occupied by the region. For uniform stress and strain fields, Eq. (2.29) can be rewritten as

$$U = WV. \tag{2.30}$$

The total strain energy increment can be expressed as

$$dU = WdV + VdW = VdW. \tag{2.31}$$

Note that, in Eq. (2.31), $dV = 0$ for plastic deformation. The strain energy increments within and out of the neck can be obtained as

$$dU^a = V^a dW^a = \left(l^2 t_1 - l t_1^2 \right) (\sigma_e)_1 \, d\epsilon_e^a, \tag{2.32}$$

$$\begin{aligned} dU^b = V^b dW^b = l t_1^2 (\sigma_e)_1 (d\epsilon_e^a + \Delta d\epsilon_e) = l t_1^2 (\sigma_e)_1 d\epsilon_e^a + l t_1^2 (\sigma_e)_1 \Delta d\epsilon_e \\ = V^b dW^a + V^b \Delta dW. \end{aligned} \tag{2.33}$$

The total strain energy increment hereby becomes

$$dU = dU^a + dU^b = V^a dW^a + V^b dW^a + V^b \Delta dW = VdW^a + V^b \Delta dW. \tag{2.34}$$

Eq. (2.34) indicates that the total strain energy increment, dU, consists of two parts, i.e., the unperturbed strain energy increment, VdW^a, and the perturbing strain energy increment, $V^b \Delta dW$. As a perturbation, $V^b \Delta dW$ should be much smaller than VdW^a.

Therefore, $V dW^a$ can be approximated as dU, while $V^b \Delta dW$ can be treated as its perturbation, ΔdU.

Note that $d\epsilon_e^a$ and $\Delta d\epsilon_e$ are proportional to dW^a and ΔdW, respectively. $\Delta d\epsilon_e$ can hereby be expressed as

$$\Delta d\epsilon_e = \frac{\Delta dW}{dW^a} d\epsilon_e^a = \frac{V}{V^b} \frac{V^b \Delta dW}{V dW^a} d\epsilon_e^a = \frac{l}{t_1} \frac{\Delta dU}{dU} d\epsilon_e^a. \tag{2.35}$$

where

$$t_1 = t_0 e^{(\epsilon_{\mathrm{III}})_1} = t_0 \exp\left(-\frac{1+\rho}{\cos^2 \psi_1 + \rho \sin^2 \psi_1} n \right). \tag{2.36}$$

Eq. (2.35) leads one to assume $\Delta dU / dU$ is a small positive constant. This physically means that a small fixed fraction of the total strain energy increment is consumed to form the neck.

The compatibility condition through the neck can be expressed as

$$d\epsilon_{22}^a = d\epsilon_{22}^b. \tag{2.37}$$

Rearranging Eq. (2.37) gives

$$\Delta d\epsilon_{22} = \Delta d\epsilon_{\mathrm{I}} \sin^2 \psi_1 + \Delta d\epsilon_{\mathrm{II}} \cos^2 \psi_1 = 0. \tag{2.38}$$

Eq. (2.38) implies that $\Delta d\epsilon_{33} = -\Delta d\epsilon_{11}$. $\Delta d\epsilon_{33}$ can hereby be related to $\Delta d\epsilon_e$ by

$$\Delta d\epsilon_{33} = -\frac{1 - \tan^2 \psi_1}{\sqrt{\frac{2}{3}\left[\hat{A}_{\mathrm{I,I}} + \tan^4 \psi_1 \hat{A}_{\mathrm{II,II}} + \left(1 - \tan^2 \psi_1\right)^2 \hat{A}_{\mathrm{III,III}} \right]}} \Delta d\epsilon_e. \tag{2.39}$$

Combining Eqs. (2.35), (2.36), and (2.39) gives

$$\chi_1 = \frac{\left(1-\tan^2\psi_1\right)\sqrt{\hat{A}_{\mathrm{I,I}}+\hat{A}_{\mathrm{III,III}}}}{\sqrt{\hat{A}_{\mathrm{I,I}}+\tan^4\psi_1\hat{A}_{\mathrm{II,II}}+\left(1-\tan^2\psi_1\right)^2\hat{A}_{\mathrm{III,III}}}} \times \exp\left(\frac{\sin^2\psi_1+\rho\cos^2\psi_1}{\cos^2\psi_1+\rho\sin^2\psi_1}n\right). \tag{2.40}$$

As can be seen, χ_1 consists of two factors: the former one is the ratio of $\Delta d\epsilon_{33}/\Delta d\epsilon_e$ to $\Delta d\epsilon_{33}^{**}/\Delta d\epsilon_e^{**}$, which is a function of ψ_1 (see Eqs. (2.37) – (2.39)); the latter one equals $\Delta d\epsilon_e/\Delta d\epsilon_e^{**}$ and hereby t_1^{**}/t_1 (see Eqs. (2.35) and (2.36)). As can be seen, for given material properties, χ_1 is a function of ρ and ψ_1. This implies that χ_1 actually characterizes the effects of ρ and ψ_1 on the neck formation.

The relationship between f_1^{**} and t_0 can also be obtained from Eqs. (2.35), (2.36), and (2.39). Eq. (2.35) implies that $\Delta d\epsilon_e^{**}$ is proportional to t_1^{**-1}, while Eq. (2.39) implies that $\Delta d\epsilon_{33}^{**}$ is proportional to $\Delta d\epsilon_e^{**}$. Therefore, $\Delta d\epsilon_{33}^{**}$ is proportional to t_1^{**-1}. Eq. (2.36) also implies that $t_1^{**} = t_0\exp(-n)$. Therefore, $\Delta d\epsilon_{33}^{**}$ is also proportional to t_0^{-1}. Note that $f_1^{**} = 1-\Delta d\epsilon_{33}^{**}$. It then turns out that $(1-f_1^{**})^{-1}$ is proportional to t_0. The sheet thickness effect can hereby be incorporated as follows: first, select a experimental forming limit curve of sheet metals of a given sheet thickness; second, find the value of f_1^{**} which fits the predicted forming limit curve well to the experimental one; last, adjust the value of f_1^{**} to obtain the forming limit curves for different sheet thicknesses.

2.1.5 Neck Evolution

The neck evolution can be modeled first by imposing an major strain increment, $d\epsilon_{\mathrm{I}}^{b}$, on the material within the neck and then by finding the corresponding major strain increment out of the neck, $d\epsilon_{\mathrm{I}}^{a}$, iteratively till the failure criterion is met [95].

As ϵ_e increases to $\epsilon_e + d\epsilon_e$, the current effective stress becomes

$$\sigma_Y = K\dot{\epsilon}_e^m \left(\epsilon_e + d\epsilon_e\right)^n, \tag{2.41}$$

where $d\epsilon_e$ can be expressed in terms of $d\epsilon_{\mathrm{I}}$ as shown in Eq. (2.15). The neck generally rotates during continued deformation. The change in the neck orientation can be represented by

$$\tan\left(\psi + d\psi\right) = \frac{1 + d\epsilon_{\mathrm{I}}^{a}}{1 + d\epsilon_{\mathrm{II}}^{a}} \tan\psi. \tag{2.42}$$

The current defect ratio, f, can be obtained as

$$f = \frac{t^b}{t^a} = \frac{t_1^b e^{\epsilon_{33}^b - (\epsilon_{33})_1}}{t_1^a e^{\epsilon_{33}^a - (\epsilon_{33})_1}} = e^{\epsilon_{33}^b - \epsilon_{33}^a}. \tag{2.43}$$

The compatibility condition through the neck can be expressed as

$$d\epsilon_{22}^a = d\epsilon_{22}^b, \tag{2.44}$$

where $d\epsilon_{22}$ can be expressed as

$$d\epsilon_{22} = d\epsilon_{\mathrm{I}} \sin^2\psi + d\epsilon_{\mathrm{II}} \cos^2\psi. \tag{2.45}$$

Eq. (2.45) can be rewritten as

$$d\epsilon_{22} = \left(\sin^2\psi + \rho\cos^2\psi\right) d\epsilon_{\mathrm{I}}. \tag{2.46}$$

It can be seen that, once $d\epsilon_{\mathrm{I}}^b$, $d\epsilon_{\mathrm{I}}^a$ and ρ^a are specified, $d\epsilon_{\mathrm{II}}^b$ and ρ^b can be obtained from Eqs. (2.44) and (2.46).

For the special case of $\tan\psi = \sqrt{-\rho^a}$ ($\rho^a \leq 0$), $d\epsilon_{22} = 0$ within and out of the neck. This implies that

$$\rho^b = \rho^a = -\tan^2\psi. \tag{2.47}$$

The stress resultants acting on the neck cross section are

$$N_{11} = \sigma_{11}t, \quad N_{12} = \sigma_{12}t. \tag{2.48}$$

These stress resultants have to be continuous through the neck, that is,

$$N^a_{11} = N^b_{11}, \quad N^a_{12} = N^a_{12}. \tag{2.49}$$

Substituting Eq. (2.48) into Eq. (2.49) and noting that $f = t^b/t^a$ give

$$\sigma^a_{11} = f\sigma^b_{11}, \quad \sigma^a_{12} = f\sigma^b_{12}. \tag{2.50}$$

Note that σ_{11} and σ_{12} can be related to σ_{I} and σ_{II} by

$$\sigma_{11} = \sigma_{\mathrm{I}}\cos^2\psi + \sigma_{\mathrm{II}}\sin^2\psi, \tag{2.51}$$

$$\sigma_{12} = -\left(\sigma_{\mathrm{I}} - \sigma_{\mathrm{II}}\right)\cos\psi\sin\psi, \tag{2.52}$$

where σ_{I} and σ_{II} can be expressed in terms of $d\epsilon_{\mathrm{I}}$ as shown in Eqs. (2.13) and (2.14), respectively. Also note that σ_n can by related to σ_{11} and σ_{12} by

$$\sigma_n^2 = \sigma_{11}^2 + \sigma_{12}^2. \tag{2.53}$$

Combining Eqs. (2.50) and (2.53) gives

$$\sigma^a_n = f\sigma^b_n. \tag{2.54}$$

Let $\mu = \sigma_n/\sigma_e$. σ_n can then be expressed in terms of σ_e as

$$\sigma_n = \mu\sigma_e, \tag{2.55}$$

where σ_e can be expressed in terms of $d\epsilon_{\mathrm{I}}$ as shown in Eq. (2.16). Substituting Eq. (2.41) into Eq. (2.55) gives

$$\sigma_n = \mu K\dot{\epsilon}_e^m\left(\epsilon_e + d\epsilon_e\right)^n. \tag{2.56}$$

Substituting Eq. (2.56) into Eq. (2.54) gives

$$\mu^a K\left(\dot{\epsilon}_e^a\right)^m\left(\epsilon_e^a+d\epsilon_e^a\right)^n=f\mu^b K\left(\dot{\epsilon}_e^b\right)^m\left(\epsilon_e^b+d\epsilon_e^b\right)^n. \tag{2.57}$$

Note that $d\epsilon_e^a/d\epsilon_e^b=\dot{\epsilon}_e^a/\dot{\epsilon}_e^b$. Eq. (2.57) can then be rewritten as

$$\mu^a\left(d\epsilon_e^a\right)^m\left(\epsilon_e^a+d\epsilon_e^a\right)^n=f\mu^b\left(d\epsilon_e^b\right)^m\left(\epsilon_e^b+d\epsilon_e^b\right)^n. \tag{2.58}$$

For the other cases, $d\epsilon_{22}\neq 0$ and $\rho^a\neq\rho^b$. Let $\beta=d\epsilon_e/d\epsilon_{22}=\dot{\epsilon}_e/\dot{\epsilon}_{22}$. Eq. (2.56) can then be rewritten as

$$\sigma_n=\mu K\left(\beta\dot{\epsilon}_{22}\right)^m\left(\epsilon_e+d\epsilon_e\right)^n. \tag{2.59}$$

Accordingly, Eq. (2.57) can be rewritten as

$$\mu^a K\left(\beta\dot{\epsilon}_{22}^a\right)^m\left(\epsilon_e^a+d\epsilon_e^a\right)^n=f\mu^b K\left(\beta\dot{\epsilon}_{22}^b\right)^m\left(\epsilon_e^b+d\epsilon_e^b\right)^n. \tag{2.60}$$

Note that $\dot{\epsilon}_{22}^a=\dot{\epsilon}_{22}^b$. Eq. (2.60) can then be rewritten as

$$\mu^a\left(\beta^a\right)^m\left(\epsilon_e^a+d\epsilon_e^a\right)^n=f\mu^b\left(\beta^b\right)^m\left(\epsilon_e^b+d\epsilon_e^b\right)^n. \tag{2.61}$$

In this part, the Van Wijngaarden-Dekker-Brent method [96] is employed to solve for $d\epsilon_{\mathrm{I}}^a$ satisfying either Eq. (2.58) or Eq. (2.61). In fact, Eq. (2.61) provides higher accuracy compared with Eq. (2.58). This is because Eq. (2.58) includes the factors $\left(dc_e^a\right)^m$ and $\left(d\epsilon_e^b\right)^m$ and hereby causes its solution to depend on the step size. However, if the step size is properly selected, Eqs. (2.58) and (2.61) can both provide sufficient accuracy. The numerical results show that a step size of 0.001 is appropriate. The calculation process lasts till the failure criterion is met. After this, the final values of ϵ_{II}^a and ϵ_{I}^a can be taken as the limiting strains for given values of ρ^a and ψ_1.

Note that, for a given value of ρ^a, ϵ_{II}^a and ϵ_{I}^a are only functions of ψ_1. Since the neck causing material failure should provide the lowest value of ϵ_{I}^a, the real limiting

strains can be determined by finding the value of ψ_1 minimizing ϵ_{I}^a. In this part, the Brent method [96] is employed to find ψ_1. Once ψ_1 is obtained, its corresponding values of ϵ_{II}^a and ϵ_{I}^a determines the point for the given value of ρ^a on the forming limit curve.

2.1.6 Failure Criterion

In the previous works, a number of failure criterions have been proposed in different ways. According to Marciniak and Kuczynski [16], failure occurs when f drops below a critical value, while according to several other authors [20, 30, 35, 97, 98], failure occurs when a certain strain increment the ratio (e.g., $d\epsilon_e^b/d\epsilon_e^a$, $d\epsilon_{\mathrm{I}}^b/d\epsilon_{\mathrm{I}}^a$, or either $d\epsilon_{11}^b/d\epsilon_{11}^a$ or $d\epsilon_{12}^b/d\epsilon_{12}^a$) attains a critical value (e.g., 10). A failure criterion should be able to characterize the strain increment discontinuity through the neck cross section and also to avoid the problem of division by zero. In this part, a novel failure criterion being able to meet these two requirements, is proposed.

Consider two neighboring material points on the interface between the neck and its neighboring region with one on the neck side of the interface and the other on the other side. Let $d\boldsymbol{\epsilon}_n$ denote a strain increment related vector on the interface, which is given by

$$d\boldsymbol{\epsilon}_n = d\boldsymbol{\epsilon} \cdot \mathbf{n}, \tag{2.62}$$

or

$$d\boldsymbol{\epsilon}_n = d\epsilon_{11}\mathbf{i} + d\epsilon_{12}\mathbf{j}. \tag{2.63}$$

The difference in $d\boldsymbol{\epsilon}_n$ between these two material points can be obtained as

$$\Delta d\boldsymbol{\epsilon}_n = d\boldsymbol{\epsilon}_n^b - d\boldsymbol{\epsilon}_n^a = \left(d\epsilon_{11}^b - d\epsilon_{11}^a\right)\mathbf{i} + \left(d\epsilon_{12}^b - d\epsilon_{12}^a\right)\mathbf{j}. \tag{2.64}$$

$\Delta d\boldsymbol{\epsilon}_n$ characterizes the magnitude and direction of the relative motion of the material point on the neck side with respect to that on the other side. The failure criterion can hereby be proposed as follows: failure occurs when

$$\frac{|\Delta d\boldsymbol{\epsilon}_n|}{|d\boldsymbol{\epsilon}_n^a|} = \sqrt{\frac{\left(d\epsilon_{11}^b - d\epsilon_{11}^a\right)^2 + \left(d\epsilon_{12}^b - d\epsilon_{12}^a\right)^2}{d\epsilon_{11}^{a2} + d\epsilon_{12}^{a2}}} \geq \kappa, \tag{2.65}$$

where κ is a sufficiently great critical value and is set to be 9 in this part in correspondence to the critical value of 10 in the previous works. Eq. (2.65) physically means that failure occurs when the material point on the neck side moves away from that on the other side at a high relative rate. By using Eq. (2.65), the problem of division by zero is avoided.

2.2 Results and Discussion

2.2.1 Competition between Two Types of Necks

As mentioned above, the deformation process can be divided into two stages: at the first stage, the sheet is uniformly deformed till the first set of necks forms perpendicular to the major strain direction at $\epsilon_1 = n$; at the second stage, different sets of necks form successively along different orientations, evolve simultaneously, and compete with each other to cause material failure during continued deformation till the failure criterion is first met in one set of them. The numerical results show that two types

of necks are most likely to cause material failure: the first type of necks forms perpendicular to the major strain direction at $\epsilon_1 = n$; the second type of necks form along certain favorable orientations at $\epsilon_1 > n$. Here the term “favorable” means that the orientations enable the necks to evolve very quickly. As can be seen, the first type of necks forms earliest but may not evolve very quickly, while the second type of necks does not form earliest but evolves very quickly. As a result, in thin sheets, since all the necks just need to evolve over limited strain to cause material failure, the first type of necks can cause material failure earliest; in thick sheets, since most of the necks have to evolve over considerable strain to cause material failure, the second type of necks can cause material failure just after their formation. Therefore, the first type of necks is more competitive in thin sheets, while the second type of necks is more competitive in thick sheets.

The above discussion implies that a forming limit curve can be determined by only focusing on the competition between the aforementioned two types of necks, or specifically, first by plotting the forming limit curves due to the two types of necks, respectively, and then by connecting the lower points on the two curves along each proportional path. Fig. 2.2 (a) and (b) illustrate how this is accomplished: in Fig. 2.2 (a), the parabolic curves are due to the first type of necks, while the decline curves are due to the second type of necks; in Fig. 2.2 (b), the forming limit curves are obtained by connecting the lower points on the corresponding parabolic and decline curves along each proportional path.

(a)

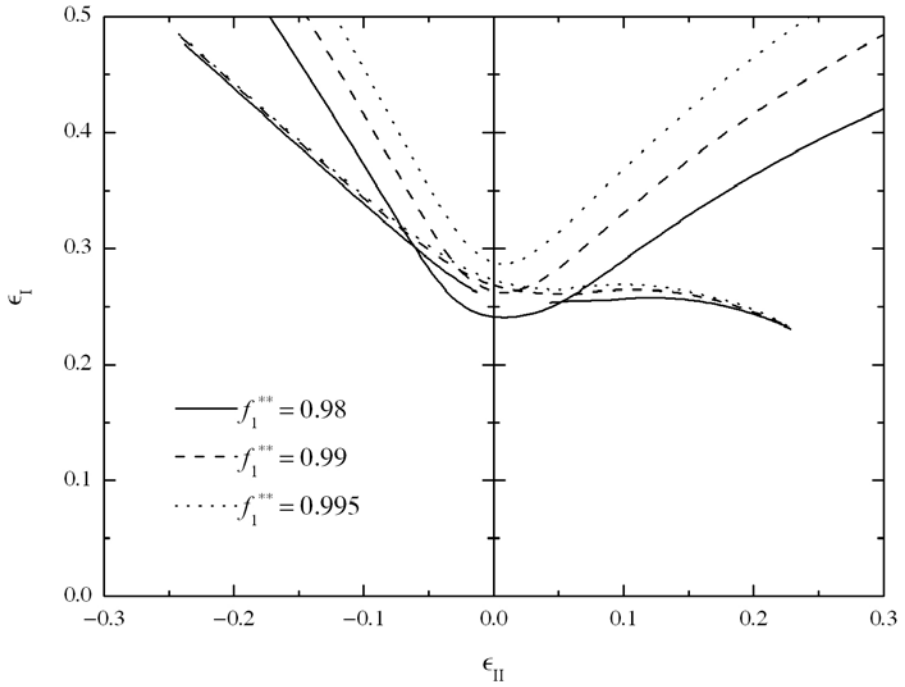

Fig. 2.2. (a) Competition between two types of necks for different sheet thicknesses ($(1 - f_1^{**})^{-1}$ is proportional to t_0), (b) the resulting forming limit diagrams, and (c) the corresponding ψ_1 versus ρ curves.

(b)

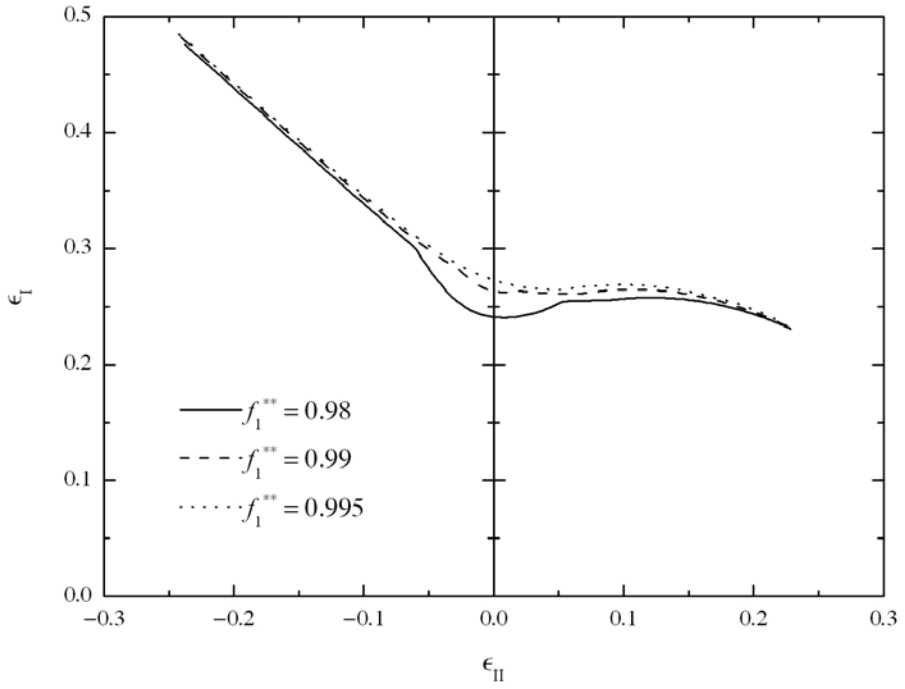

Fig. 2.2 Continued.

(c)

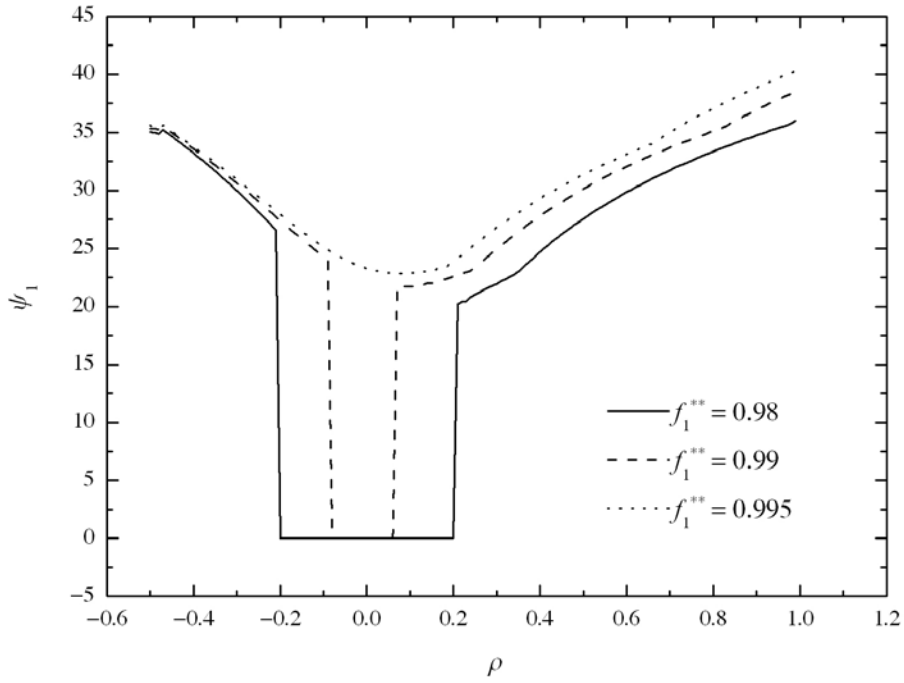

Fig. 2.2 Continued.

Fig. 2.2 (b) and (c) show the forming limit curves for different values of f_1^{**} and the corresponding ψ_1 versus ρ curves, respectively. Here the other parameters are set to be $n = 0.23$, $m = 0.012$, and $R = 1$. As mentioned above, $(1 - f_1^{**})^{-1}$ is proportional to t_0. Therefore, Fig. 2.2 (b) actually shows the forming limit curves for different sheet thickness. As a result of the competition between the two types of necks, the forming limit curves may exhibit different characteristics in different regions. For example, in Fig. 2.2 (b), the curve for $f_1^{**} = 0.98$ exhibits different characteristics in three different regions: for $-0.2 < \rho < 0.2$, the first type of necks causes material failure, and this curve section is parabolic; for $\rho < -0.2$ and $\rho > 0.2$, the second type of necks causes

material failure, and the curve sections are decline. These three regions can classified into two types, i.e., the neck formation dominate region for $-0.2 < \rho < 0.2$ and the neck evolution dominate region for $\rho < -0.2$ and $\rho > 0.2$. As can be seen in Fig. 2.2 (b), as f_1^{**} increases, the forming limit curve shifts upwards, while the neck formation dominant region first shrinks and finally disappears. This can be understood by noting that, as the sheet thickness increases, the second type of necks becomes more competitive. As can be seen in Fig. 2.2 (c), for each curve, $\psi_1 = 0°$ in the neck formation dominate region, while $\psi_1 \neq 0°$ in the neck evolution dominate region. Therefore, the value of ψ_1 can indicate the region type.

2.2.2 Comparison between the Predicted Forming Limit Curve and the Experimental Results

Fig. 2.3 compares the predicted forming limit curve of 2036-T4 aluminum with the experimental results [5]. Here the tensile properties are $n = 0.245$, $m = -0.006$, and $R = 0.78$ [5, 6], while the parameters are set to be $m = 0.001$ and $f_1^{**} = 0.99$ to obtain the numerical results. Since the calculation converges only if $m > 0$, here m is set to be 0.001 instead of -0.006 according to Hecker [6]. The failure criterion implies that the predicted curve should lie among the necked and fractured data points. As can be seen, the predicted curve fits the experimental results well except that some of the necked data points lie below the predicted curve or even below $\epsilon_1 = n$. This is because, in real

materials, the internal defects weaken the material and hereby lower the sheet metal formability.

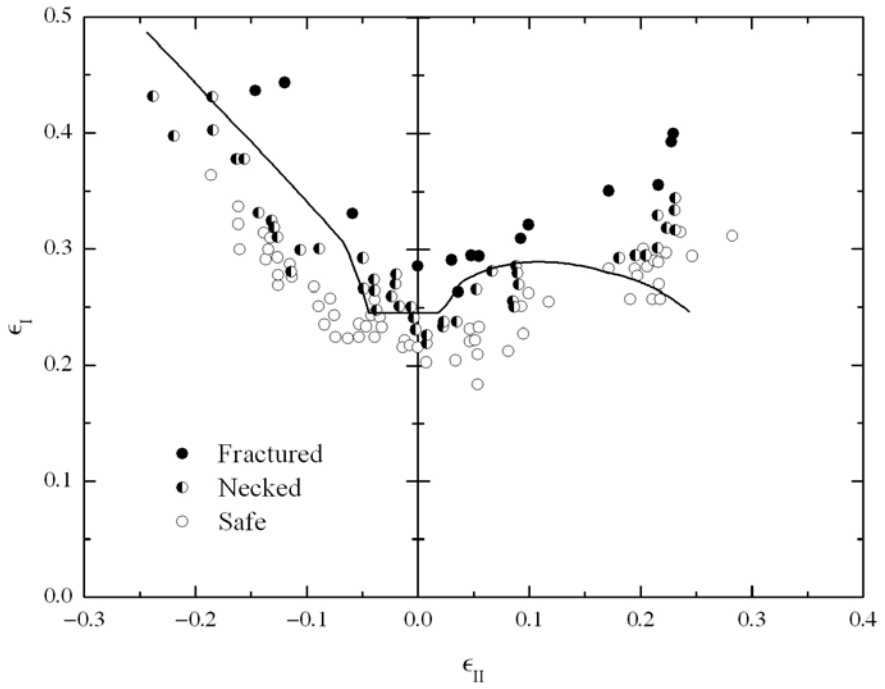

Fig. 2.3. Comparison between the predicted forming limit curve and the experimental results for 2036-T4 aluminum (after Hecker [5]).

2.2.3 Effects of the Strain Hardening Behavior, the Sheet Thickness, and Plastic Anisotropy

Fig. 2.4 (a) and (b) show the forming limit curves for different values of n and the corresponding ψ_1 versus ρ curves, respectively. Here the other parameters are set to be $m = 0.012$, $R = 1$, and $f_1^{**} = 0.99$. In Fig. 2.4 (a), as n increases, the forming limit curve approximately scales up. This can be understood by noting that, in Eq. (2.24), $(\epsilon_I)_1$

is proportional to n. However, this does not imply that these curves are similar. Fig. 2.4 (b) shows that the extent of the neck formation dominant region also varies with n. This indicates that n affects not only the neck formation but also the neck evolution. In fact, Eqs. (2.58) and (2.61) have already indicated that n affects the equilibrium equations for the neck evolution.

(a)

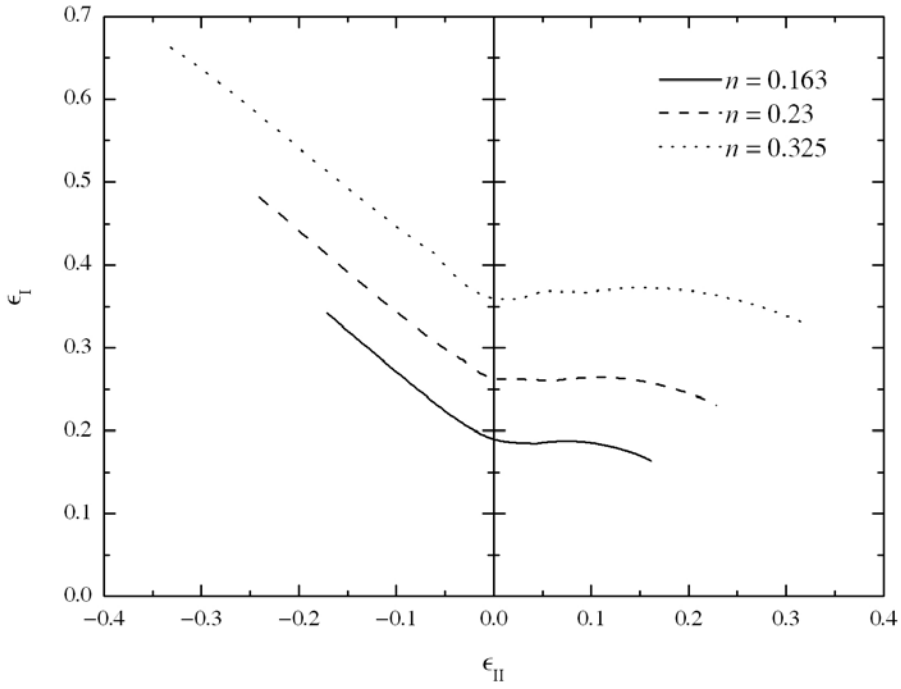

Fig. 2.4. (a) Forming limit diagrams for different values of n and (b) the corresponding ψ_1 versus ρ curves.

(b)

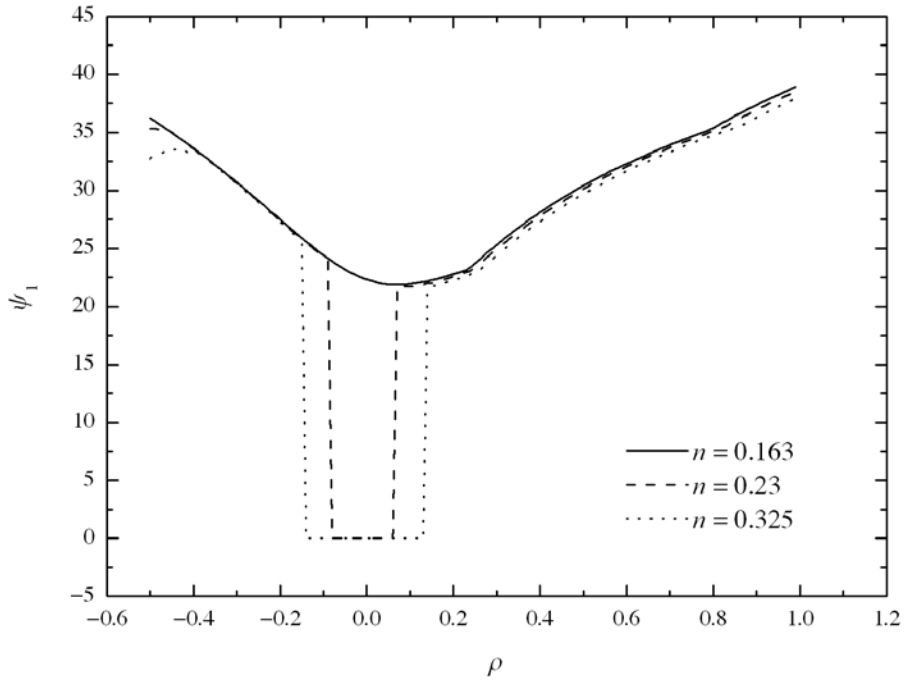

Fig. 2.4 Continued.

Fig. 2.5 shows the forming limit curves for different values of m, where $m = 0.001$ is used to approximate a rate-independent strain hardening behavior. Here the other parameters are set to be $n = 0.23$, $R = 1$, and $f_1^{**} = 0.99$. In Fig. 2.5, as m increases, the forming limit curve shifts upwards, while the neck formation dominant region shrinks and finally disappears. This indicates that, as m increases, the second type of necks also becomes more competitive. It is also worth notice that one segment of the curve for $m = 0.001$ is horizontal. This indicates that, for low values of m (in fact, and also f_1^{**}), the first type of necks forms earliest and causes material failure only after the first strain increment.

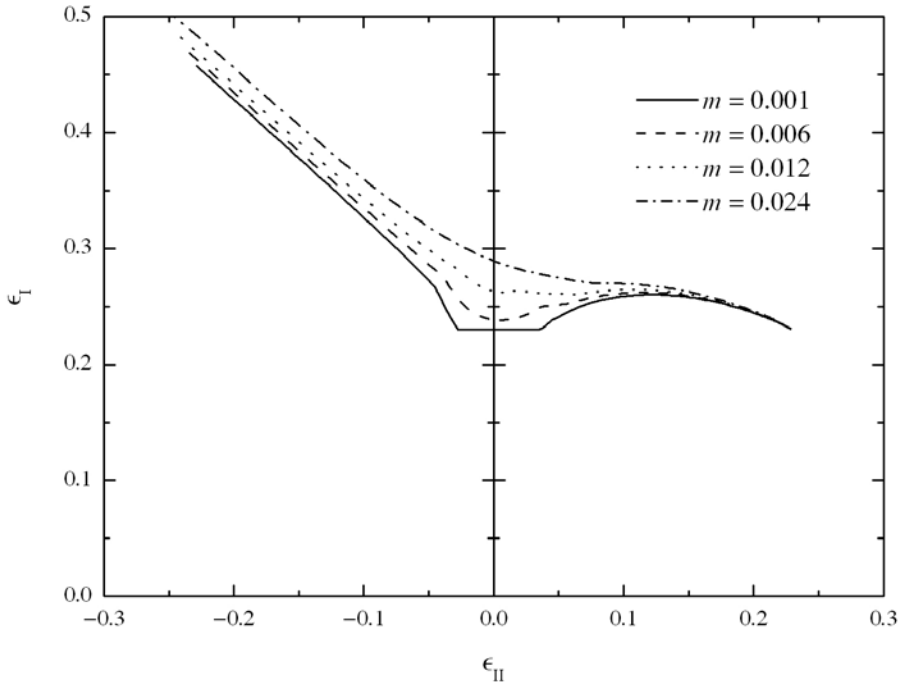

Fig. 2.5. Forming limit curves for different values of m.

Fig. 2.6 plots ϵ_1^*/n versus $\log_2 (1 - f_1^{**})^{-1}$ for different values of m. Here the other parameters are set to be $n = 0.23$ and $R = 1$. As mentioned above, $(1 - f_1^{**})^{-1}$ is proportional to t_0, or in other words, the difference between $\log_2 (1 - f_1^{**})^{-1}$ and $\log_2 t_0$ is merely a constant. Therefore, Fig. 2.6 actually shows the relationship between the sheet metal formability under plane strain conditions and the sheet thickness. The curve for $m = 0.012$ exhibits a typical trend and can be taken as an example. It can be divided into three segments, i.e., a horizontal one, a rapidly increasing one, and a slowly increasing one. Here each segment actually corresponds to a certain deformation process: in the first segment, since the sheet is very thin, the first type of necks forms

earliest and causes material failure only after the first strain increment; in the second segment, as the sheet becomes thicker, the first type of necks causes material failure after they have evolved over certain strain; in the third segment, as the sheet becomes sufficiently thick, the second type of necks turns out to cause material failure. It is worth notice that the slope of the second segment is greater than that of the third one. This is in agreement with Fig. 2.2 (a). In Fig. 2.2 (a), as f_1^{**} increases, the parabolic curve shifts upwards more rapidly than the decline curve. In addition, in Fig. 2.6, for $m = 0.001$, the sheet metal formability under plane strain conditions is insensitive to the sheet thickness; as m increases, it becomes more and more sensitive to the sheet thickness. This indicates that m affects the sheet thickness effect.

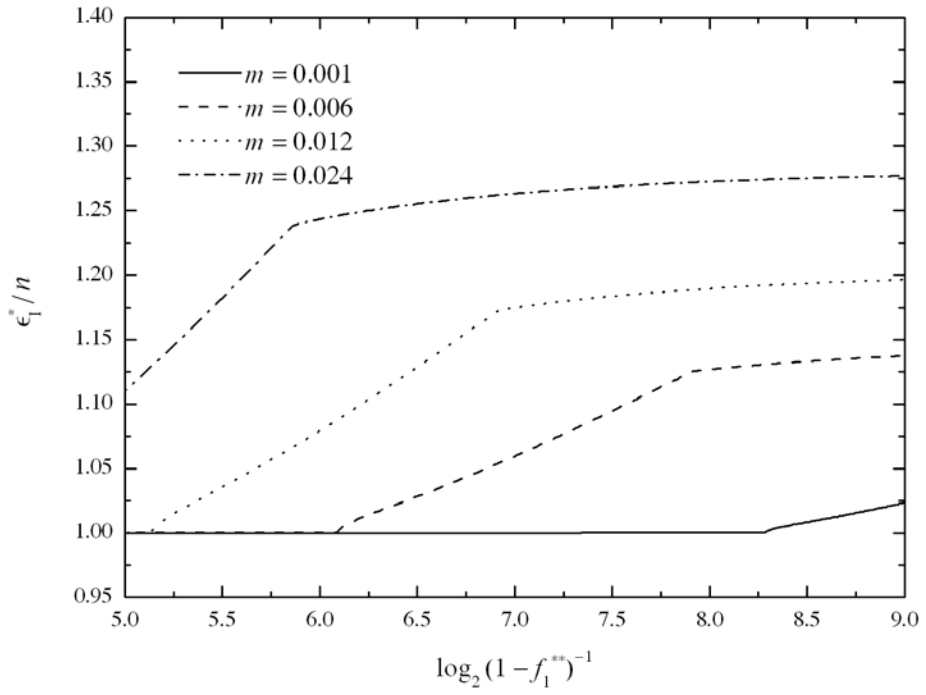

Fig. 2.6. ϵ_1^*/n versus $\log_2 (1 - f_1^{**})^{-1}$ for different values of m.

Fig. 2.7 (a) and (b) show the forming limit curves for different values of R and the corresponding ψ_1 versus ρ curves, respectively. Here the other parameters are set to be $n = 0.23$, $m = 0.012$, and $f_1^{**} = 0.99$. It has been widely accepted that the lowest point on the forming limit curve is always located at $\rho = 0$. However, in Fig. 2.7 (a), as R increases, the neck formation dominant region shifts rightwards, while the lowest point shifts from the left-hand side of the FLD to the right-hand of the FLD. This indicates that the R-value can generally affect the neck evolution and hereby the location of the lowest point. However, it is also worth notice that, in Fig. 2.7 (a), all the curves intersect at the same point at $\rho = 0$. This indicates that the R-value cannot affect the neck evolution under plane strain conditions. In addition, the numerical results also show that, for $R < 1$, the sheet metal formability may exhibit a dramatic increase in the neighborhood of $\rho = 1$. This implies that, in this case, the neck evolution is inhibited by the prevailing stress and strain states. This, however, is unrealistic. This is because here the sheet is assumed to have no internal defects. In fact, in real materials, the defect growth limits the sheet metal formability. As a result, the real sheets may fail before it is deformed to a great extent. Therefore, in Fig. 2.7 (a), the curves including unrealistic predictions are marked with a dot. Here the dot denotes that, to the right of it, the predictions become unrealistic.

(a)

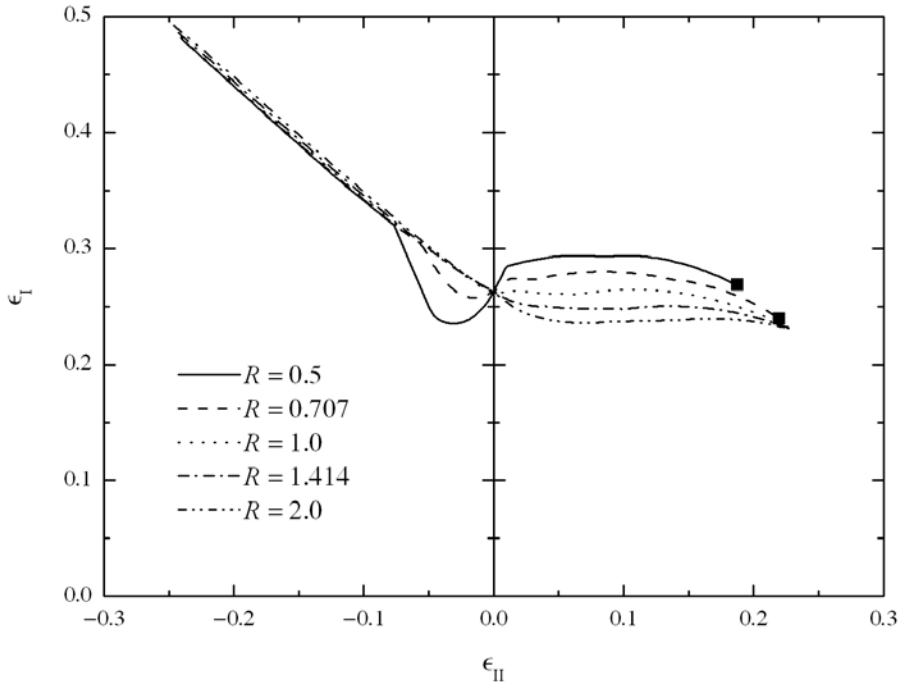

Fig. 2.7. Forming limit curves for different values of R and the corresponding ψ_1 versus ρ curves.

(b)

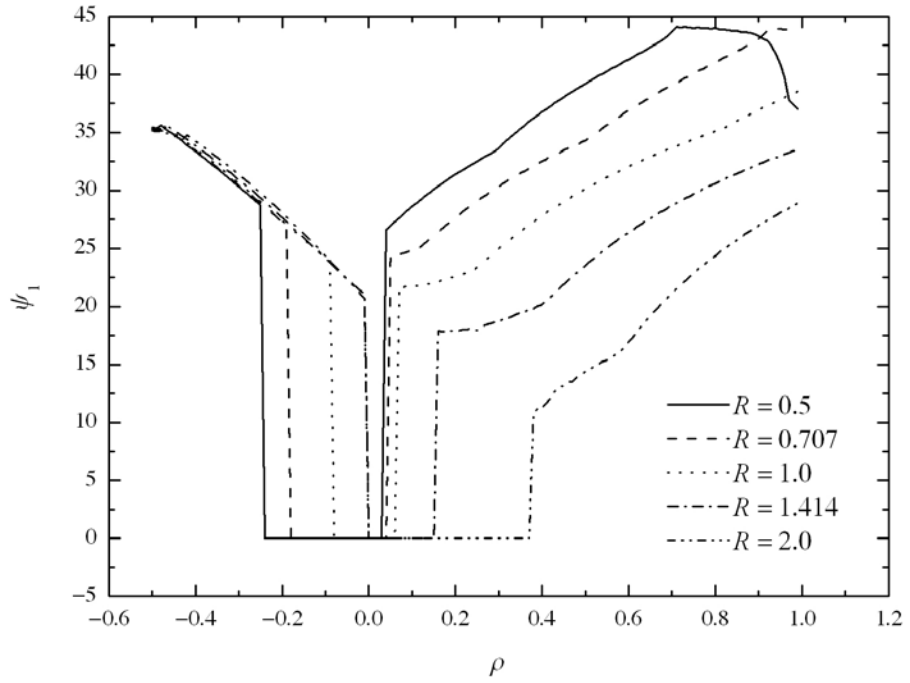

Fig. 2.7 Continued.

2.2.4 Comparison among Different Models

Fig. 2.8 (a) compares the forming limit curves predicted by the proposed model, the Hill model, the M-K model, and the Swift model, respectively, and Fig. 2.8 (b) also compares the corresponding ψ_1 versus ρ curves. Here the parameters are set to be $n = 0.23$, $m = 0.012$, $R = 1$, and $f_1^{**} = 0.99$, and specially, $f_1^{**} = 0.9904$ is used for the M-K model to let $\epsilon_1^* = n$. In Fig. 2.8 (a) and (b), for $\rho \leq 0$, the curves predicted by the proposed model and the Hill model are quite close to each other. However, there exists a major distinction between these two models. According to Hill [94], localized necking

occurs as a velocity discontinuity. This implies that the necks in the Hill model do not evolve. However, here the necks are assumed to form at a certain stage of the deformation process and also to be able to evolve after that. It should be noted that, only if the neck evolution is considered, the parameters such as the sheet thickness and plastic anisotropy can be incorporated into the model. Therefore, the Hill model is valid for rigid plastic materials, while the proposed model is appropriate for strain hardening materials.

(a)

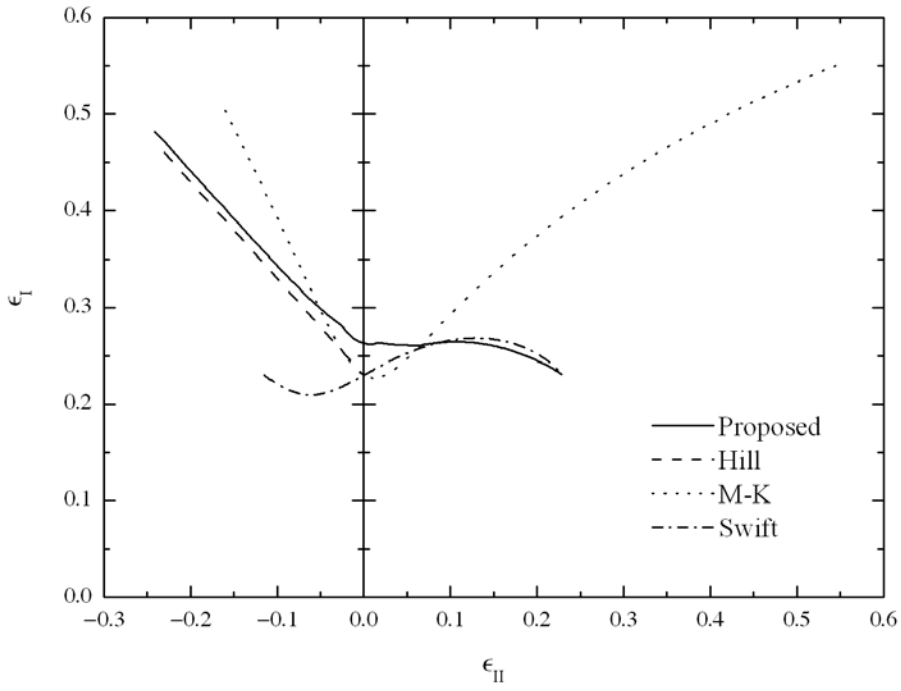

Fig. 2.8. (a) Forming limit curves predicted by the proposed model, the Hill model, the M-K model, and the Swift model and (b) the corresponding ψ_1 versus ρ curves.

(*b*)

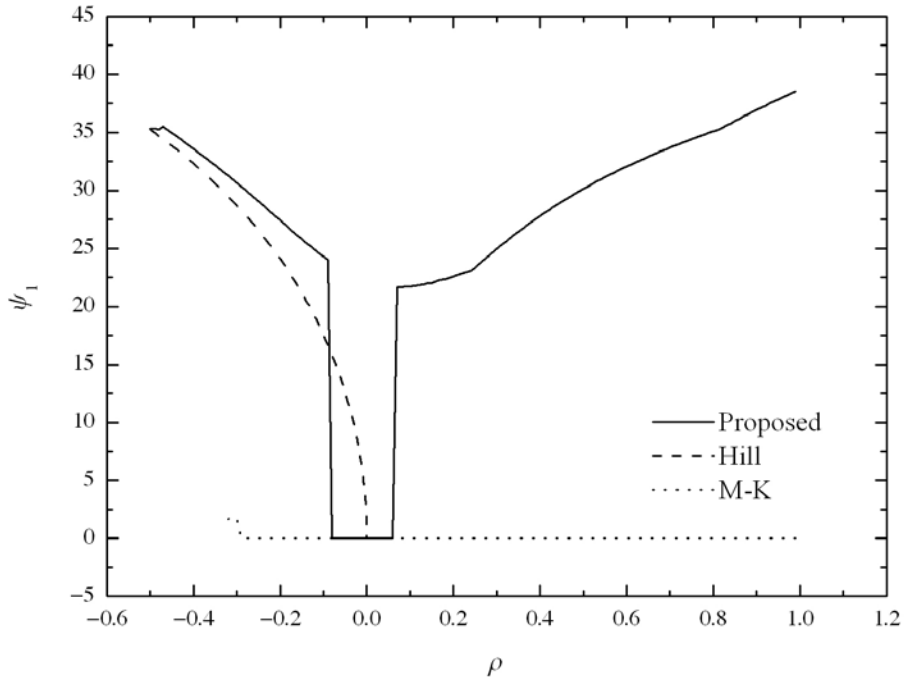

Fig. 2.8 Continued.

For comparison purposes, the quadratic Hill yield criterion is also implemented into the M-K model. As expected, in Fig. 2.8 (a), for $\rho \geq 0$, the M-K model tends to overestimate the sheet metal formability. As mentioned above, this is also why nowadays the nonquadratic yield criterions are preferable to the quadratic ones. However, as can be seen in Fig. 2.3, although the quadratic Hill yield criterion is also implemented into the proposed model, the predicted forming limit curve still fits the experimental results well. In fact, as mentioned above, Marciniak and Kuczynski [16] suggested that it is the initial geometric defects that cause material failure. However, several authors [24-26] found that the defect sizes measured in the test specimens are

much smaller than those selected to fit the predictions to the experimental results, while Tadros and Mellor [26] observed that, in sheet metals with premade aligned grooves, the necks tend to form perpendicular to the major strain direction no matter the groove orientation. All these findings indicate that the initial geometric defects may not be the main cause of localized necking. Therefore, it is necessary to consider the neck formation at a certain stage of the deformation process.

2.3 Conclusions

This part provides an alternative approach to modeling the localized necking in anisotropic sheet metals. Localized geometric softening at a certain stage of deformation rather than the initial defects is found to be the main cause of localized necking. The deformation process is found to consist of two stages: at the first stage, the sheet is uniformly deformed till the first set of necks forms perpendicular to the major strain direction at $\epsilon_1 = n$; at the second stage, different sets of necks form successively along different orientations, evolve simultaneously, and compete with each other to cause material failure during continued deformation till the failure criterion is first met in one set of them. The Considère-type criterion is found to be able to determine the critical strains for a neck to form. The energy-based hypothesis is found to be able to quantify the defect ratio at the neck formation. Two types of necks are found to be most competitive to cause material failure: the first type of necks forms earliest but evolves not so quickly; the second type of necks forms not so early but evolves very quickly. As

a result, the forming limit curves are found to exhibit different characteristics in different regions that can be classified as the neck formation dominant region and the neck evolution dominant region. ψ_1 is found to be able to indicate the region type. The predicted forming limit curve of 2036-T4 aluminum is also found to mostly fit the experimental results well. The strain hardening exponent is found to affect both the neck formation and the neck evolution, while the sheet thickness and the strain rate hardening exponent are found to only affect the neck evolution. The R-value is also found to affect the neck evolution and hereby the location of the lowest point on the forming limit curve. The proposed model is also found to work well with the quadratic Hill yield criterion.

The following conclusions can be drawn from the above findings:

- In sheet metals subject to nonuniform deformation, localized necking tends to occur in the regions of strain ratios resulting in lower limiting major strains;
- In sheet metals of nonuniform thicknesses in different regions, localized necking tends to occur in the regions of smaller thicknesses;
- If the sheet thickness is below a certain value, increasing the sheet thickness can enhance the sheet metal formability; if the sheet thickness is above this value, increasing the sheet thickness will hardly enhance the sheet metal formability the sheet metal formability;
- More realistic yield criterions and strain hardening behaviors can be implemented into the proposed model.

3. MODELING THE BIFURCATION IN ANISOTROPIC BULK AND SHEET METALS

3.1 Theoretical Considerations

3.1.1 Conditions for the Onsets of Bifurcation in Bulk Metals

Let a bulk metal be subject to uniform quasistatic deformation. Accroding to Rudnicki and Rice, once the uniform rates of deformation enables a singularity to exist, bifurcation occurs. Fig. 3.1 schematically illustrates the bulk metal geometry at the onset of bifurcation. Introduce a Cartesian coordinate system x_i ($i = 1, 2, 3$) with the x_2 axis perpendicular to the planar bifurcation band. Note that the rate of deformation has to vary through the band. The difference between the velocity gradient within and out of the band can be expressed as

$$\Delta \left(\partial v_i / \partial x_j \right) = \partial v_i / \partial x_j - \left(\partial v_i / \partial x_j \right)^0 , \tag{3.1}$$

where v_i denotes the velocity vector; Δ denotes the difference between the quantities within and out of the band; the superscript 0 denotes the quantities out of the band. Also note that the velocity is still continuous at the onset of bifurcation. This implies that the compatibility condition

$$\Delta \left(\partial v_i / \partial x_j \right) = g_i n_j \tag{3.2}$$

has to be satisfied, where n_i denotes the unit normal to the bifurcation band and g_i is a function of the distance through the band ($n_i x_i$) and vanishes out of the band.

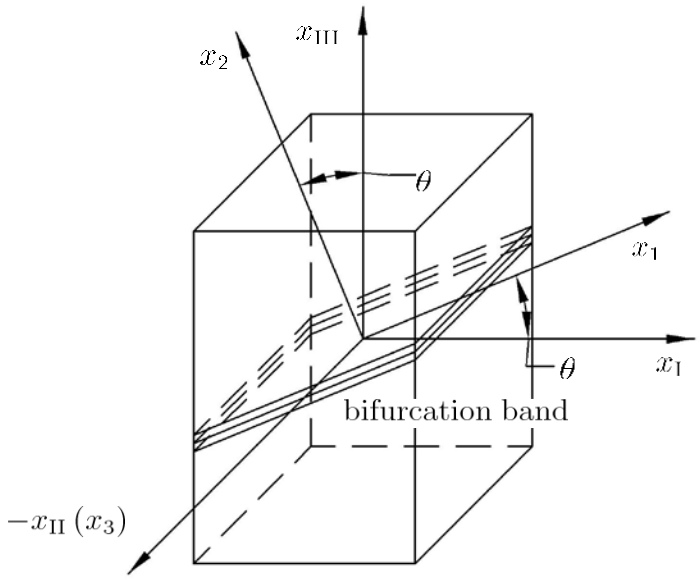

Fig. 3.1. Bulk metal geometry at the onset of bifurcation.

The equilibrium equations can be expressed as

$$\frac{\partial}{\partial x_i}\left(\dot{\sigma}_{ij} - v_k \frac{\partial \sigma_{ij}}{\partial x_k}\right) = 0, \tag{3.3}$$

where the superposed dot denotes the material time rate. Note that the stresses are uniform here. Eq. (3.3) hereby reduces to

$$\frac{\partial \dot{\sigma}_{ij}}{\partial x_i} = 0. \tag{3.4}$$

In addition, Eq. (3.2) also implies that, at the onset of bifurcation, the stresses remain uniform out of the band and are functions of $n_i x_i$ within the band. This, together with Eq. (3.4), implies that

$$n_i \Delta \dot{\sigma}_{ij} = 0. \tag{3.5}$$

Let the constitutive relations take a general form of

$$\overset{\triangledown}{\sigma}_{ij} = \mathscr{L}_{ijkl} D_{kl}, \tag{3.6}$$

where D_{ij} denotes the rate of deformation tensor and is the symmetric part of $\partial v_i / \partial x_j$, and $\mathscr{L}_{ijkl}$ is symmetric with respect to the interchange of i and j and also that of k and l. The Jaumann co-rotational rate of stress tensor takes the form of

$$\overset{\triangledown}{\sigma}_{ij} = \dot{\sigma}_{ij} + \sigma_{ik}\Omega_{kj} - \Omega_{ik}\sigma_{kj}, \tag{3.7}$$

where Ω_{ij} denotes the rotation tensor and is the anti-symmetric part of $\partial v_i / \partial x_j$. Note that $\dot{\sigma}_{ij}$ is not invariant to under rigid rotations [57, 58]. For this reason, $\overset{\triangledown}{\sigma}_{ij}$ rather than $\dot{\sigma}_{ij}$ is employed here.

Let the constitutive relations take the forms of

$$\overset{\triangledown}{\sigma}_{ij} = \mathscr{L}_{ijkl} D_{kl} \tag{3.8}$$

within the band and

$$\overset{\triangledown}{\sigma}{}^0_{ij} = \mathscr{L}^0_{ijkl} D^0_{kl} \tag{3.9}$$

out of the band. Eq. (3.2) can be rewritten in terms of D_{ij} as

$$\Delta D_{ij} = D_{ij} - D^0_{ij} = \frac{1}{2}\left(g_i n_j + n_i g_j\right). \tag{3.10}$$

Combining Eqs. (3.5), (3.8), (3.9) and (3.10) gives

$$\left(n_i \mathscr{L}_{ijkl} n_l + R_{jk}\right) g_k = n_i \left(\mathscr{L}^0_{ijkl} - \mathscr{L}_{ijkl}\right) D^0_{kl}, \tag{3.11}$$

where

$$2R_{ij} = -n_i \left(n_k \sigma_{kj}\right) + \left(n_k \sigma_{kl} n_l\right) \delta_{ij} + \left(n_k \sigma_{ki}\right) n_j - \sigma_{ij} \tag{3.12}$$

is the difference between $\overset{\nabla}{\sigma}_{ij}$ and $\dot{\sigma}_{ij}$. Eq. (3.11) represents the conditions for the onsets of bifurcation in bulk metals. If $\mathscr{L}_{ijkl}$ remains continuous within and out of the band at the onset of bifurcation (i.e., $\mathscr{L}_{ijkl} = \mathscr{L}^0_{ijkl}$), the term on the right hand side of Eq. (3.11) vanishes. In this case, only if g_i is nonzero, or to say, only if

$$\det\left(n_i \mathscr{L}_{ijkl} n_l + R_{jk}\right) = 0, \tag{3.13}$$

bifurcation can occur.

3.1.2 Constitutive Relations

Following Rudnicki and Rice [58], let the strain increments be related by the stress increments by

$$d\epsilon^p_e = \frac{1}{H}\left(d\sigma_e + \mu d\sigma_m\right), \tag{3.14}$$

$$d\epsilon^p_m = \beta d\epsilon^p_e, \tag{3.15}$$

where σ_e and σ_m denote the effective and mean stresses, respectively; ϵ^p_e and ϵ^p_m denote the effective and mean plastic strains, respectively; μ and β denote the internal friction coefficient and the dilatancy factor, respectively [57, 58]; H denotes the tangent modulus. σ_e can be expressed in terms of the anisotropy tensor, $\mathscr{A}_{ijkl}$, as shown in Section 2.1.1. As mentioned above, also assume the material to exhibit a non-linear

strain hardening/softening behavior whose form will be presented in the section of results and discussion.

Divide D_{ij} into its elastic and plastic parts, D^e_{ij} and D^p_{ij}. D^p_{ij} can be obtained from Eqs. (3.14) and (3.15) as

$$D^p_{ij} = \frac{1}{H}\left(\frac{3}{2}\frac{\hat{\sigma}_{ij}}{\sigma_Y} + \beta\delta_{ij}\right)\left(\frac{3}{2}\frac{\hat{\sigma}_{kl}}{\sigma_Y} + \mu\delta_{kl}\right)\overset{\triangledown}{\sigma}_{kl}. \tag{3.16}$$

where $\hat{\sigma}_{ij} = (\mathscr{A}_{ijkl}\sigma'_{kl})'$ is named as the nominal stress tensor in this part and is a purely deviatoric second order tensor. Eq. (3.16) can be inversed to give $\overset{\triangledown}{\sigma}_{ij}$ by solving

$$\overset{\triangledown}{\sigma}_{ij} = \mathscr{C}_{ijkl}\left(D_{kl} - D^p_{kl}\right), \tag{3.17}$$

where $\mathscr{C}_{ijkl}$ denotes the fourth order elastic modulus tensor. Let the material be elastically isotropic. $\mathscr{C}_{ijkl}$ hereby takes the form of [57]

$$\mathscr{C}_{ijkl} = G\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right) + \left(K - \frac{2}{3}G\right)\delta_{ij}\delta_{kl}. \tag{3.18}$$

Substituting Eqs. (3.16) and (3.18) into Eq. (3.17) gives

$$\begin{aligned}\overset{\triangledown}{\sigma}_{ij} &= 2GD_{ij} + \left(K - \frac{2}{3}G\right)\delta_{ij}D_{kk} \\ &- \frac{9\left(G\frac{\hat{\sigma}_{ij}}{\sigma_Y} + K\beta\delta_{ij}\right)\left(G\frac{\hat{\sigma}_{kl}}{\sigma_Y} + K\mu\delta_{kl}\right)}{H + 3G\omega^2 + 9K\mu\beta}D_{kl},\end{aligned} \tag{3.19}$$

where

$$\omega = \left(\frac{3}{2}\frac{\hat{\sigma}_{ij}\hat{\sigma}_{ij}}{\sigma_Y^2}\right)^{1/2}. \tag{3.20}$$

Eq. (3.19) represents the constitutive relations for the material under consideration.

3.1.3 Critical Tangent Moduli at the Onsets of Bifurcation in Bulk Metals

To reduce the complexity of derivations, here assume the term R_{ij} in Eq. (3.13) to be negligible. Note this term is the difference between $\overset{\triangledown}{\sigma}_{ij}$ and $\dot{\sigma}_{ij}$. This assumption hereby implies that $\overset{\triangledown}{\sigma}_{ij} = \dot{\sigma}_{ij}$. Further investigation will show that making this assumption is equivalent to neglecting the terms of magnitude σ_Y/G in the expression for H/G. Eq. (3.13) then reduces to

$$\det \mathscr{L}_{2ij2} = 0, \tag{3.21}$$

where

$$\begin{aligned}\mathscr{L}_{2ij2} &= G\left(\delta_{2i}\delta_{j2} + \delta_{ij}\right) + \left(K - \frac{2}{3}G\right)\delta_{2i}\delta_{j2} \\ &- \frac{9\left(G\frac{\hat{\sigma}_{2i}}{\sigma_Y} + K\beta\delta_{2i}\right)\left(G\frac{\hat{\sigma}_{j2}}{\sigma_Y} + K\mu\delta_{j2}\right)}{H + 3G\omega^2 + 9K\mu\beta}.\end{aligned} \tag{3.22}$$

Solving for the tangent modulus, H, from Eq. (3.21) gives

$$\begin{aligned}H &= \frac{9}{K + \frac{4}{3}G}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y} + K\beta\right)\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y} + K\mu\right) \\ &+ 9G\left[\left(\frac{\hat{\sigma}_{12}}{\sigma_Y}\right)^2 + \left(\frac{\hat{\sigma}_{23}}{\sigma_Y}\right)^2\right] - 3G\omega^2 - 9K\mu\beta.\end{aligned} \tag{3.23}$$

Eq. (3.23) indicates that H is a function of G, K, μ, β, and the nominal stresses. For most strain hardening materials and also the materials under consideration, H is a decreasing function of ϵ_e^p. This implies that, as ϵ_e^p increases, bifurcation is more and more likely to occur. In addition, for certain values of ϵ_e^p and ϵ_m^p, H should be a function only

of the band orientation. This implies that the critical tangent modulus at the onset of bifurcation, H_c, can be obtained by finding the band orientation maximizing H.

Let $\hat{\sigma}_K$ ($K = \mathrm{I}, \mathrm{II}, \mathrm{III}$) denote the principal nominal stresses with $\hat{\sigma}_\mathrm{I} \geq \hat{\sigma}_\mathrm{II} \geq \hat{\sigma}_\mathrm{III}$. Introduce a Cartesian coordinate system x_K with the x_K axis parallel to the $\hat{\sigma}_K$ direction (see Fig. 3.1). The $\hat{\sigma}_{ij}$'s in Eq. (3.23) can be related to $\hat{\sigma}_K$ by

$$\hat{\sigma}_{22} = n_\mathrm{I}^2 \hat{\sigma}_\mathrm{I} + n_\mathrm{II}^2 \hat{\sigma}_\mathrm{II} + n_\mathrm{III}^2 \hat{\sigma}_\mathrm{III}, \tag{3.24}$$

$$\hat{\sigma}_{12}^2 + \hat{\sigma}_{23}^2 = n_\mathrm{I}^2 \hat{\sigma}_\mathrm{I}^2 + n_\mathrm{II}^2 \hat{\sigma}_\mathrm{II}^2 + n_\mathrm{III}^2 \hat{\sigma}_\mathrm{III}^2 - \hat{\sigma}_{22}^2, \tag{3.25}$$

where n_K denotes the unit normal to the bifurcation band.

Rudnicki and Rice [58] have shown that, for plastically isotropic material, if the principal deviatoric stresses are distinct, the normal to the bifurcation band lies perpendicular to the x_II direction only if the inequality

$$2\sigma'_\mathrm{I} - \sigma'_\mathrm{II} - \sigma'_\mathrm{III} > \frac{2}{3}\sigma_Y (\beta + \mu) \tag{3.26}$$

is satisfied, otherwise it lies perpendicular to the x_III direction. For plastically anisotropic materials under consideration here, a similar statement can be made, that is, if the principal nominal stresses are distinct, the normal to the bifurcation band lies perpendicular to the $\hat{\sigma}_\mathrm{II}$ direction only if the inequality

$$2\hat{\sigma}_\mathrm{I} - \hat{\sigma}_\mathrm{II} - \hat{\sigma}_\mathrm{III} > \frac{2}{3}\sigma_Y (\beta + \mu) \tag{3.27}$$

is satisfied, otherwise it lies perpendicular to the x_III direction. Needleman and Rice [99] suggested that, for metals, μ and β should both be very small. This implies that the term on the right hand side of Eq. (3.27) is close to zero. In addition, it can also be deduced

that, if two of the principal nominal stresses are equal, the normal to the bifurcation band is nonunique.

To reduce to complexity of derivations, let the principal nominal stresses be distinct. As mentioned above, the normal to the bifurcation band always lies perpendicular to one principal nominal stresses direction. Here let the normal lies perpendicular to the x_{II} direction. If the normal lies perpendicular to the x_{III} direction, one just needs to exchange the subscripts II and III in the corresponding equations. Without loss of generality, let the x_3 direction is set to coincide with the $-x_{\mathrm{II}}$ direction. This leads g_3 to vanish. Eq. (3.21) hereby reduces to

$$\det \mathscr{L}_{2\alpha\beta 2} = 0, \tag{3.28}$$

where the indices α and β vary from 1 to 2. Accordingly, Eq. (3.23) reduces to

$$H = \frac{9}{K + \frac{4}{3}G}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y} + K\beta\right)\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y} + K\mu\right) + 9G\left(\frac{\hat{\sigma}_{12}}{\sigma_Y}\right)^2 - 3G\omega^2 - 9K\mu\beta. \tag{3.29}$$

Let θ denote the angle between the normal and the x_{III} direction. n_K can then be related to θ by

$$n_{\mathrm{I}} = \sin\theta, \quad n_{\mathrm{II}} = 0, \quad n_{\mathrm{III}} = \cos\theta. \tag{3.30}$$

Now the problem becomes finding the value of θ maximizing H, say θ_c. Rudnicki and Rice [58] have shown that θ_c can be solved using the method of Lagrange multipliers. Following Rudnicki and Rice [58], θ_c can be finally obtained as

$$\cos 2\theta_c = \frac{(1-2\nu)\,\hat{\sigma}_{\mathrm{II}} - \frac{2}{3}\sigma_Y\,(1+\nu)\,(\beta+\mu)}{\hat{\sigma}_{\mathrm{I}} - \hat{\sigma}_{\mathrm{III}}}. \tag{3.31}$$

Eq. (3.31) indicates that, even if all the three principal nominal stresses are distinct, there may exist two distinct possible values of θ_c, say $(\theta_c)_1$ and $(\theta_c)_2$, where

$$(\theta_c)_2 = \pi - (\theta_c)_1. \tag{3.32}$$

This physically means that two bifurcation bands may develop simultaneously at the onset of bifurcation. In fact, this phenomenon is in agreement with the experimental observations. After this, H_c can be obtained as

$$H_c = G\frac{1+\nu}{1-\nu}(\beta-\mu)^2 - \frac{9}{2}G(1+\nu)\left(\frac{\hat{\sigma}_{\mathrm{II}}}{\sigma_Y} + \frac{\beta+\mu}{3}\right)^2. \tag{3.33}$$

Eq. (3.33), together with the hardening law, determines the strains for the onset of bifurcation along a certain proportional strain path. This will be discussed in detail in the section of results and discussion.

3.1.4 Correction due to the Co-Rotational Stress Rates

In Eq. (3.13), the term R_{ij} is assumed to be negligible. Here incorporate the correction due to R_{ij} into the critical tangent modulus. Still let the x_3 direction coincide with the $-x_{\mathrm{II}}$ direction. Eq. (3.13) hereby becomes

$$\det\left(\mathscr{L}_{2\alpha\beta 2} + R_{\alpha\beta}\right) = 0. \tag{3.34}$$

Solve for H from Eq. (3.34) gives

$$H = \frac{9}{K+\frac{4}{3}G}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y} + K\beta\right)\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y} + K\mu\right) + \frac{9G}{1+\dfrac{\sigma_{22}-\sigma_{11}}{2G}}\left(\frac{\hat{\sigma}_{12}}{\sigma_Y}\right)^2 - 3G\omega^2 - 9K\mu\beta. \tag{3.35}$$

It can be deduced that the difference between the expression for H/G obtained from Eq. (3.35) and that obtained from Eq. (3.29) is of magnitude σ_Y/G. For metals, this difference is quite small. For this reason, the value of θ maximizing H in Eq. (3.35), or to say, the corrected θ_c, should be close to that in Eq. (3.31). For the same reason, the corrected H_c should also be close that in Eq. (3.33). In this part, the Brent method [96] is employed to find the corrected θ_c and H_c with knowing their approximate values.

3.1.5 Conditions for the Onsets of Bifurcation in Sheet Metals

The conditions for the onsets of bifurcation in sheet metals are quite similar with those in bulk metals. However, there exists one significant distinction between these two sets of conditions, that is, the stress resultant equilibrium is required here rather than the stress equilibrium. Fig. 3.2 schematically illustrates the sheet metal geometry at the onset of bifurcation. Let the sheet metal subject to in-plane biaxial extension (i.e., σ_{III} vanishes). Let σ_{I} and σ_{II} denote the principal stresses in the sheet plane, and also let $\hat{\sigma}_{\mathrm{I}}$ and $\hat{\sigma}_{\mathrm{II}}$ denote the corresponding principal nominal stresses with $\hat{\sigma}_{\mathrm{I}} \geq \hat{\sigma}_{\mathrm{II}}$. Introduce a Cartesian coordinate system Introduce a Cartesian coordinate system x_K ($K = \mathrm{I}, \mathrm{II}, \mathrm{III}$) with the x_{I} and x_{II} axes parallel to the σ_{I} and σ_{II} directions, respectively. Also Introduce a Cartesian coordinate system x_i ($i = 1, 2, 3$) with the x_2 axis perpendicular to the bifurcation band. Unlike Stören and Rice [13], let the normal to the bifurcation band be able lie out of the sheet plane. Further investigation will show that whether the normal lies in the sheet plane depends on the nominal stresses.

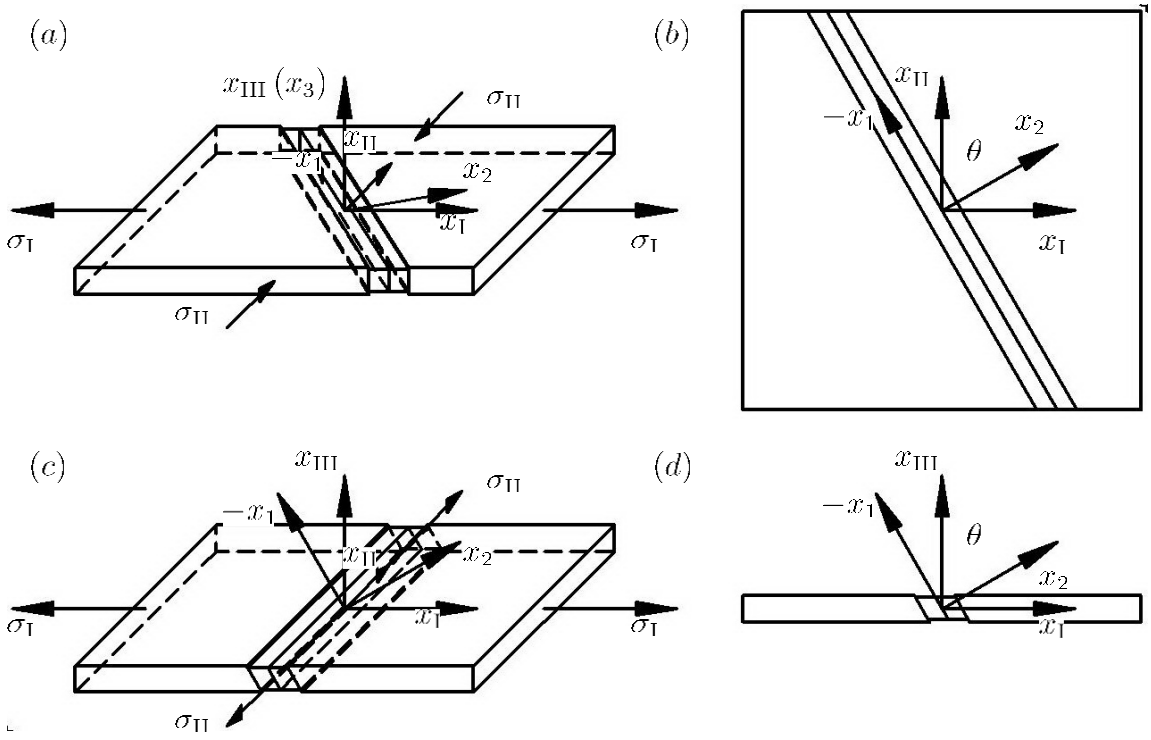

Fig. 3.2. Sheet metal geometry at the onset of bifurcation: (a) the normal to the bifurcation band lies in the sheet plane; (b) the top view of (a); (c) the normal to the bifurcation band lies out of the sheet plane; (d) the side view of (c).

The in-plane stress resultants are

$$N_{\mathrm{I}} = \sigma_{\mathrm{I}} t, \quad N_{\mathrm{II}} = \sigma_{\mathrm{II}} t, \tag{3.36}$$

where t is the initial sheet thickness. As mentioned above, the stress resultant equilibrium is required here rather than the stress equilibrium. The equilibrium equations hereby become

$$\frac{\partial}{\partial x_{\mathrm{I}}}\left(\dot{N}_{\mathrm{I}} - v_k \frac{\partial N_{\mathrm{I}}}{\partial x_k}\right) = 0, \quad \frac{\partial}{\partial x_{\mathrm{II}}}\left(\dot{N}_{\mathrm{II}} - v_k \frac{\partial N_{\mathrm{II}}}{\partial x_k}\right) = 0. \tag{3.37}$$

Similarly to Section 3.1.1, Eq. (3.37) implies that

$$\frac{\partial}{\partial x_{\mathrm{I}}}\left(\Delta\dot{\sigma}_{\mathrm{I}}+\sigma_{\mathrm{I}}\Delta D_{\mathrm{III}}\right)=0,\quad \frac{\partial}{\partial x_{\mathrm{II}}}\left(\Delta\dot{\sigma}_{\mathrm{II}}+\sigma_{\mathrm{II}}\Delta D_{\mathrm{III}}\right)=0, \tag{3.38}$$

where $\Delta D_{\mathrm{III}}=\Delta\dot{t}/t$ is the difference between the D_{III}'s within and out of the band. Substituting Eq. (3.7) into Eq. (3.38) gives

$$\frac{\partial}{\partial x_i}\left(\Delta\overset{\triangledown}{\sigma}_{ij}-\sigma_{ik}\Omega_{kj}+\Omega_{ik}\sigma_{kj}+\sigma_{ij}\Delta D_{\mathrm{III}}\right)=0, \tag{3.39}$$

and hereby

$$n_i\Delta\overset{\triangledown}{\sigma}_{ij}-n_i\sigma_{ik}\Omega_{kj}+n_i\Omega_{ik}\sigma_{kj}+n_i\sigma_{ij}\Delta D_{\mathrm{III}}=0. \tag{3.40}$$

It is convenient to write

$$D_{\mathrm{III}}=M_{kl}D_{kl}. \tag{3.41}$$

Note that the constitutive relations in Section 3.1.2 remain valid here. Combining Eqs. (3.16), (3.17) and (3.18) gives

$$D_{\mathrm{III}}=\frac{3\left(\frac{3}{2}\frac{\hat{\sigma}_{\mathrm{III}}}{\sigma_{\mathrm{Y}}}+\beta\right)\left(G\frac{\hat{\sigma}_{kl}}{\sigma_{\mathrm{Y}}}+K\mu\delta_{kl}\right)}{H+3G\omega^2+9K\mu\beta}D_{kl}. \tag{3.42}$$

M_{kl} can hereby be obtained as

$$M_{kl}=\frac{3\left(\frac{3}{2}\frac{\hat{\sigma}_{\mathrm{III}}}{\sigma_{\mathrm{Y}}}+\beta\right)\left(G\frac{\hat{\sigma}_{kl}}{\sigma_{\mathrm{Y}}}+K\mu\delta_{kl}\right)}{H+3G\omega^2+9K\mu\beta}. \tag{3.43}$$

Combining Eqs. (3.8) – (3.10), (3.40) and (3.41) gives

$$\left(n_i\mathscr{L}_{ijkl}n_l+R_{jk}+n_i\sigma_{ij}M_{kl}n_l\right)g_k=n_i\left(\mathscr{L}^0_{ijkl}-\mathscr{L}_{ijkl}\right)D^0_{kl}. \tag{3.44}$$

Eq. (3.11) represents the conditions for the onsets of bifurcation in bulk metals. Note that Eq. (3.44) is quite similar to Eq. (3.11) except that Eq. (3.44) also includes an term related to M_{kl}. This term results primarily from the restriction imposed by the stress

resultant equilibrium. Its effect will be evaluated in the section of results and discussion. Again, if $\mathscr{L}_{ijkl}$ remains continuous within and out of the band at the onset of bifurcation, the term on the right hand side of Eq. (3.44) vanishes. In this case, only if g_i is nonzero, or to say, only if

$$\det\left(n_i\mathscr{L}_{ijkl}n_l + R_{jk} + n_i\sigma_{ij}M_{kl}n_l\right) = 0, \tag{3.45}$$

bifurcation can occur.

3.1.6 Critical Tangent Moduli at the Onsets of Bifurcation in Sheet Metals

To reduce the complexity of derivations, still assume that R_{ij} is negligible, and still let the x_3 direction coincide with the x_{III} or $-x_{\mathrm{II}}$ direction, i.e., g_3 vanishes. Eq. (3.45) then becomes

$$\det\left(\mathscr{L}_{2\alpha\beta2} + \sigma_{2\alpha}M_{\beta2}\right) = 0. \tag{3.46}$$

Substituting Eqs. (3.22) and (3.43) into Eq. (3.46) and solving for H give

$$\begin{aligned} H = {} & \frac{9}{K+\frac{4}{3}G}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y}+K\beta\right)\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y}+K\mu\right)+9G\left(\frac{\hat{\sigma}_{12}}{\sigma_Y}\right)^2 \\ & -3\left(\frac{3}{2}\frac{\hat{\sigma}_{\mathrm{III}}}{\sigma_Y}+\beta\right)\left[\frac{\sigma_{22}}{K+\frac{4}{3}G}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y}+K\mu\right)+\sigma_{12}\frac{\hat{\sigma}_{12}}{\sigma_Y}\right] \\ & -3G\omega^2-9K\mu\beta. \end{aligned} \tag{3.47}$$

It can be deduced that the difference between the expression for H/G obtained from Eq. (3.47) and that obtained from Eq. (3.29) is also of magnitude σ_Y/G. This implies that, to accurately predict the onsets of bifurcation in sheet metals, one needs to incorporate R_{ij} and $\sigma_{2i}M_{j2}$ simultaneously.

Rewrite Eq. (3.45) as

$$\det\left(\mathscr{L}_{2\alpha\beta 2}+\sigma_{2\alpha}M_{\beta 2}+R_{\alpha\beta}\right)=0. \tag{3.48}$$

Solve for H from Eq. (3.48) gives

$$\begin{aligned}H=&\frac{9}{K+\frac{4}{3}G}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y}+K\beta\right)\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y}+K\mu\right)+\frac{9G}{1+\dfrac{\sigma_{22}-\sigma_{11}}{2G}}\left(\frac{\hat{\sigma}_{12}}{\sigma_Y}\right)^2\\&-3\left(\frac{3}{2}\frac{\hat{\sigma}_{\mathrm{III}}}{\sigma_Y}+\beta\right)\left[\frac{\sigma_{22}}{K+\frac{4}{3}G}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y}+K\mu\right)+\frac{\sigma_{12}}{1+\dfrac{\sigma_{22}-\sigma_{11}}{2G}}\frac{\hat{\sigma}_{12}}{\sigma_Y}\right]\\&-3\left(\frac{3}{2}\frac{\hat{\sigma}_{\mathrm{III}}}{\sigma_Y}+\beta\right)\frac{\sigma_{12}^2}{G\left(K+\frac{4}{3}G\right)\left(1+\dfrac{\sigma_{22}-\sigma_{11}}{2G}\right)}\left(G\frac{\hat{\sigma}_{22}}{\sigma_Y}+K\mu\right)\\&-3G\omega^2-9K\mu\beta.\end{aligned} \tag{3.49}$$

As expected, the difference between the expression for H/G obtained from Eq. (3.49) and that obtained from Eq. (3.29) is still of magnitude σ_Y/G. In addition, it is also worth notice that this difference even includes a term of magnitude $(\sigma_Y/G)^2$ (see the third line of Eq. (3.49)). In fact, this term is due to the interaction between R_{ij} and $\sigma_{2i}M_{j2}$.

Note that here $\hat{\sigma}_{\mathrm{I}}$ and $\hat{\sigma}_{\mathrm{II}}$ are the in-plane principal nominal stresses and that $\hat{\sigma}_{\mathrm{I}}\geq\hat{\sigma}_{\mathrm{II}}$. This implies that, although $\hat{\sigma}_{\mathrm{III}}$ is still smaller than $\hat{\sigma}_{\mathrm{I}}$, it is not always smaller than $\hat{\sigma}_{\mathrm{II}}$. As a result, the direction of the normal to the bifurcation band is more difficult to determine. Here classify the bifurcation into two modes: for the mode I bifurcation, the normal lies in the sheet plane, or to say, lies perpendicular to the x_{III} direction; for the mode II bifurcation, the normal lies out of the sheet plane, or to say, lies perpendicular to the x_{II} direction. Similarly to Section 3.1.3, for $\hat{\sigma}_{\mathrm{II}}\geq\hat{\sigma}_{\mathrm{III}}$, if Eq. (3.27) is unsatisfied, the mode I bifurcation dominates, otherwise the mode II bifurcation

dominates; for $\hat{\sigma}_{\mathrm{II}} \leq \hat{\sigma}_{\mathrm{III}}$, if Eq. (3.27) is satisfied, the mode I bifurcation dominates, otherwise the mode II bifurcation dominates. For the mode I bifurcation, the x_3 direction coincide with the x_{III} direction, and n_K can then be expressed in terms of θ as

$$n_{\mathrm{I}} = \sin\theta, \quad n_{\mathrm{II}} = \cos\theta, \quad n_{\mathrm{III}} = 0; \tag{3.50}$$

for the mode I bifurcation, the x_3 direction coincide with the $-x_{\mathrm{II}}$ direction, and n_K can then be expressed in terms of θ as

$$n_{\mathrm{I}} = \sin\theta, \quad n_{\mathrm{II}} = 0, \quad n_{\mathrm{III}} = \cos\theta. \tag{3.51}$$

Similarly to Section 3.1.4, here the corrected θ_c should be close to that in Eq. (3.31), while the corrected H_c should be close that in Eq. (3.33). The Brent method (Press, Teukolsky et al. 1992) can hereby still be employed to find the corrected θ_c and H_c with knowing their approximate values.

3.2 Results and Discussion

Extensive studies have shown that bifurcation is more likely to occur in materials exhibiting a strain softening behavior. An example of such kind of materials is the equal-channel angular extrusion (ECAE) processed Zn-40Al alloy. Fig. 3.3 shows its typical true stress-strain curve. As can be seen, it exhibits a strain hardening behavior and also a stain softening behavior. Here let the materials under consideration exhibit the same stress-strain relationship as shown Fig. 3.3, and also let them be of the same values of

$$E = 55 \text{ GPa}, \quad \nu = 0.3, \quad \beta = 0 \tag{3.52}$$

but of different values of μ and R. In addition, let the bulk or sheet metals made of these materials be subject to in-plane biaxial extension (i.e., σ_{III} vanishes), and let them be deformed along different proportional strain paths (i.e., $\rho = D_{\mathrm{II}}/D_{\mathrm{I}}$ is constant along each strain path, and here ρ is also the strain ratio). For each value of ρ, one can obtain a corresponding H_c and the major and minor strains at the onset of bifurcation, ϵ_{I} and ϵ_{II}. If one plots ϵ_{I} versus ϵ_{II} for all values of ρ, he can obtain a curve similar to the forming limit curve. Let this curve be named as the bifurcation curve. It physically represents to which extent a metal can be deformed without causing the onset of bifurcation. In this section, its dependences on the form of H_c, μ, and R will be evaluated.

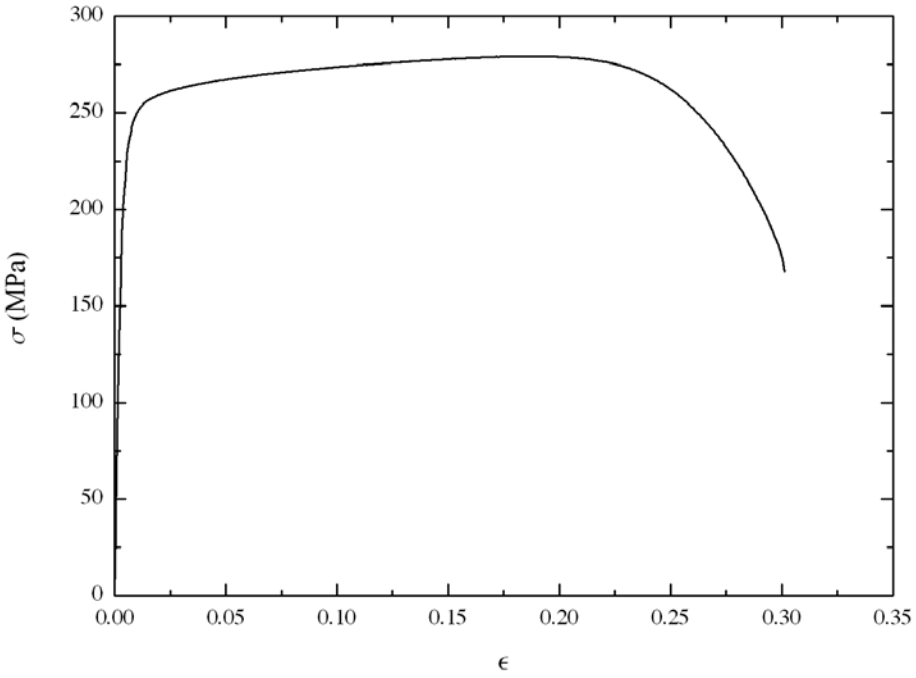

Fig. 3.3. True stress-strain curve of the ECAE processed Zn-40Al alloy (provided by Dr. Karman, Texas A&M University).

Fig. 3.4 (a) plots H_c versus ρ obtained from different equations. Here the other parameters are set to be $\mu = 0.02$ and $R = 1$, and cases 1, 2, and 3 denote that the corresponding H_c's are obtained from Eqs. (48), (50), and (64), respectively. Note that the minimum hardening modulus obtained from Fig. 3.3 is -2500 MPa. For case 1, H_c is greater than this value in two regions. Let region I denote the one in the neighborhood of $\rho = -1$, and let region II denote the one in the neighborhood of $\rho = 0$. According to Section 3.1.6, one can tell that: in region I, the mode I bifurcation dominates; in region II, the mode II bifurcation dominates. Eq. (48) indicates that H_c varies with $\hat{\sigma}_{\mathrm{II}}/\sigma_Y$ hyperbolically. For this reason, in Fig. 3.4 (a), H_c varies with ρ approximately hyperbolically in both regions. In addition, Fig. 3.4 (a) shows that the curves for different cases lie quite close to each other in region I. This indicates that the terms introduced by R_{ij} and $\sigma_{2i}M_{j2}$ are quite small there. However, Fig. 3.4 (a) also shows that the curve for case 3 prominently deviates from the other two curves in region II. This indicates that, if the mean stress is great, the terms introduced by $\sigma_{2i}M_{j2}$ can also significantly affect H_c.

(a)

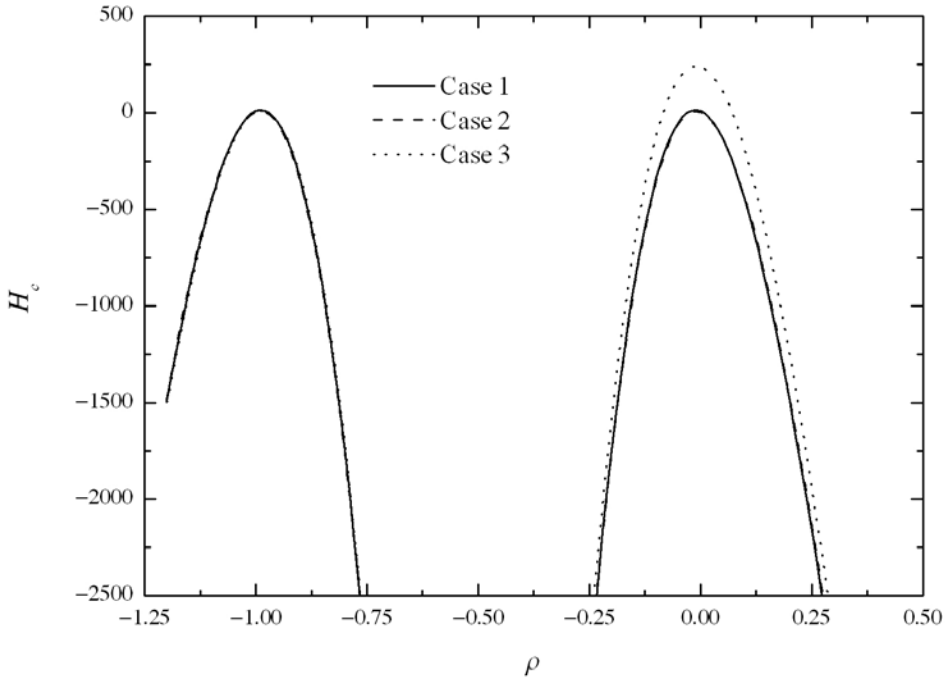

Fig. 3.4. (a) H_c versus ρ curves obtained from different equations and (b) bifurcation curves for different cases.

(b)

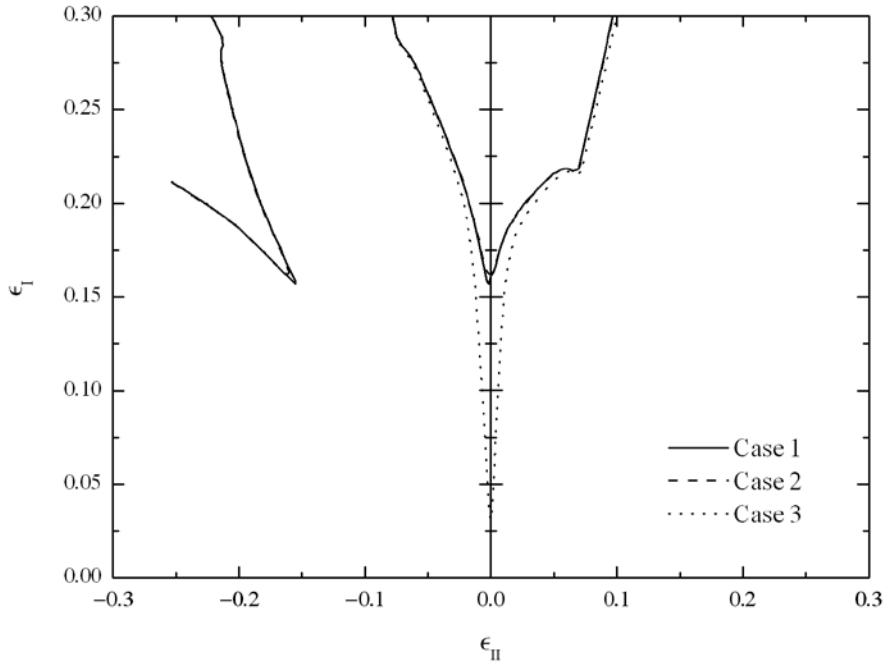

Fig. 3.4 Continued.

Fig. 3.4 (b) shows the bifurcation curves for different cases. As can be seen, each bifurcation curve consists of two segments: one segment is located in region I where the mode I bifurcation dominates; the other segment is located in region II where the mode II bifurcation dominates. In addition, each segment exhibits a V-shape with the lowest point located at either $\rho = -1$ or $\rho = 0$. This is in agreement with Fig. 3.4 (a) in which H_c attains its local maximums at $\rho = -1$ and $\rho = 0$. This also indicates that, unlike the localized necking in sheet metals, bifurcation is most likely to occur under pure shear. This can be treated as a characteristic of bifurcation. Note that, in Fig. 3.4 (a), the H_c versus ρ curve for case 3 prominently deviates from the other two curves in region II.

Accordingly, in Fig. 3.4 (b), the bifurcation curve for case 3 prominently deviates from the other two curves in region II. Especially, this curve provides an quite low local minimum at $\rho = 0$. This again indicates that, if the mean stress is great, the terms introduced by $\sigma_{2i}M_{j2}$ can also significantly affect H_c and hereby the strains at the onset of bifurcation. This physically means that, if the mean stress is great, the onsets of bifurcation in sheet metals can be significantly affected by the restrictions imposed by the stress resultant equilibrium.

Fig. 3.5 plots the bifurcation curves for case 3 as shown in Fig 4 (b) for different values of μ. Here R is set to be 1. According to Rudnicki and Rice [58], for rocks, the onset of bifurcation is quite sensitive to μ and β. This is because, for rocks, the values of μ and β can both range from 0 to 0.6 or even to 0.9. However, according to Needleman and Rice [99], for metals, μ also takes much smaller values compared to that for rocks (e.g., 0.02), while β generally equals zero. This implies that, for metals, the effects of β and μ on the onset of bifurcation may not be quite significant. Fig. 3.5 shows that, as μ increases, the bifurcation curve shift rightward and downward in region I. This physically means that high values of μ can promote the onset of bifurcation there. However, Fig. 3.5 also shows that, as μ increases, the bifurcation curve hardly shifts in region II. This can be understood by noting that, in Eq. (3.49), μ can hardly affect the magnitudes of the terms introduced by $\sigma_{2i}M_{j2}$. For this reason, if the terms introduced by $\sigma_{2i}M_{j2}$ are great enough, the effect of μ on the onset of bifurcation can be neglected.

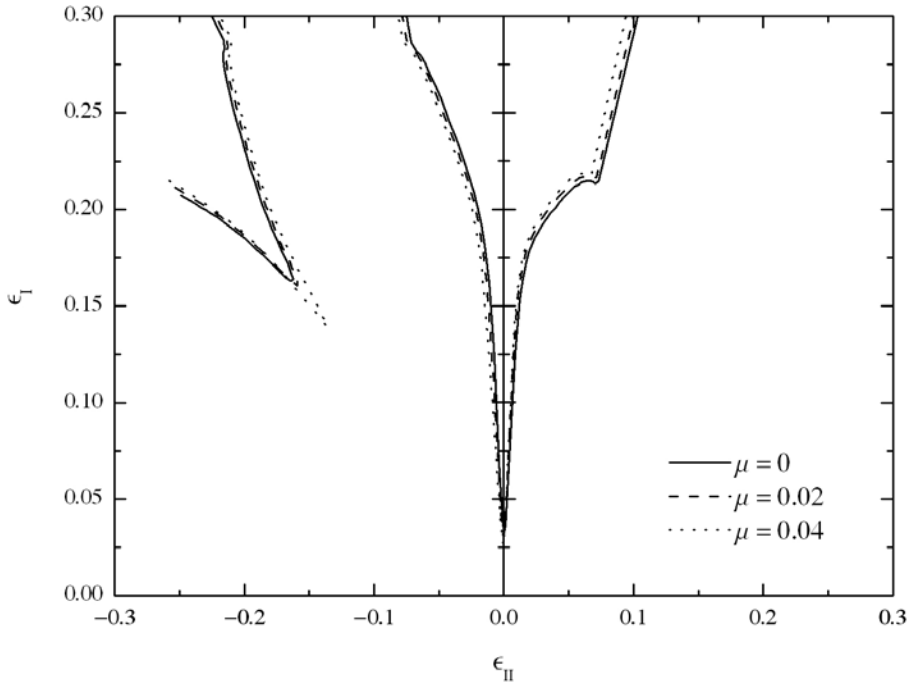

Fig. 3.5. Bifurcation curves for different values of μ.

Fig. 3.6 plots the bifurcation curves for case 3 as shown in Fig 3.4 (b) for different values of R. Here μ is set to be 0.02. Fig. 3.6 shows that, as R increases, the bifurcation curve leftward and upward and also has a decrease in the opening of its V-shape in region I. This indicates that high values of R can differently inhibit the onsets of bifurcation along different strain paths. However, Fig. 3.6 also shows that, as R increases, the bifurcation curve hardly shifts in region II. Similarly to Fig. 3.5, this can be understood by noting that, in Eq. (3.49), $\mathscr{A}_{ijkl}$ and hereby R can hardly affect the magnitudes of the terms introduced by $\sigma_{2i}M_{j2}$. For this reason, if the terms introduced by $\sigma_{2i}M_{j2}$ are great enough, the effect of R on the onset of bifurcation can be neglected.

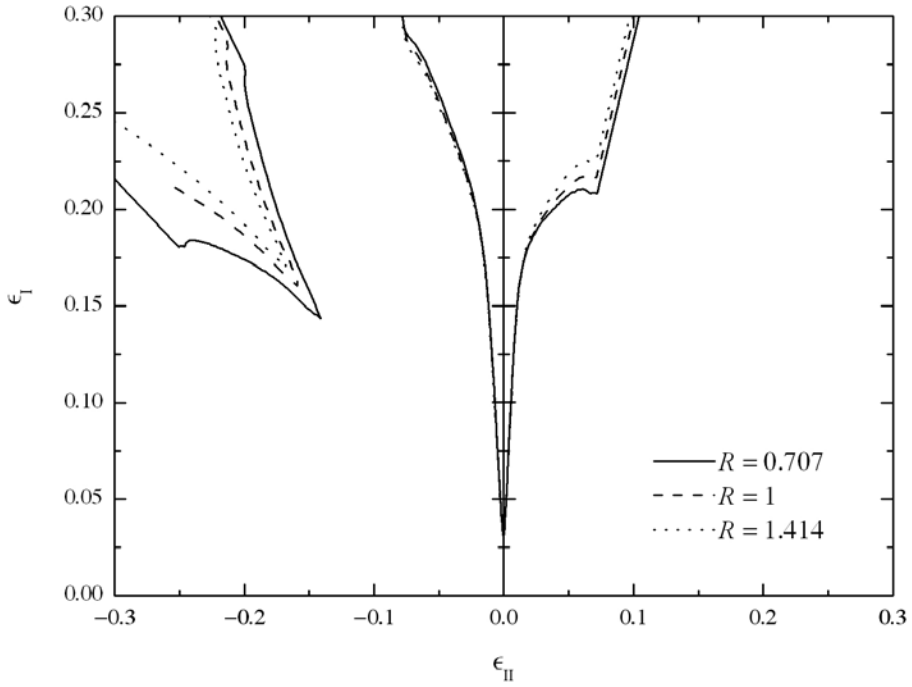

Fig. 3.6. Bifurcation curves for different values of R.

3.3 Conclusions

This part provides an applicable approach to modeling the bifurcation in anisotropic bulk and sheet metals. Plastic anisotropy and the strain hardening/softening behavior are found to be able to be coupled. Two bifurcation modes are found to exist in sheet metals. The critical tangent modulus at the onset of bifurcation and the corresponding bifurcation band orientation are found to be functions of the elasticity coefficients, the internal friction coefficient, the dilatancy factor, and the nominal stresses integrating the effects of the stresses and plastic anisotropy. The so-called

bifurcation curves are generated for the ECAE processed Zn-40Al alloy. It is found to be able to represent to which extent a metal can be deformed without causing the onset of bifurcation. The terms introduced by the co-rotational stress rates are found to be negligible, while those introduced by the stress resultant equilibrium are found to increase with the mean stress and to be able to significantly affect the onset of the mode II bifurcation. The internal friction coefficient is found to affect the onset of the mode I bifurcation but to hardly affect the onset of the mode II bifurcation. The R-value is found to differently affect the onsets of the mode I bifurcation along different strain paths but to hardly affect the onset of the mode II bifurcation.

The following conclusions can be drawn from the above findings:

- The micromechanisms resulting in macroscopic strain softening (e.g., progressive damage and the void nucleation, growth and coalescence) can promote the onset of bifurcation;
- In bulk and sheet metals subject to nonuniform deformation, bifurcation tends to occur in the regions under pure shear;
- In sheet metals subject to nonuniform deformation, bifurcation tends to occur in the regions subject to higher mean stresses;
- More realistic material properties (e.g., the yield criterion, the flow rule, and the strain hardening/softening behavior) can be implemented into the proposed model.

4. DERIVING THE CONSTITUTIVE RELATIONS FOR POROUS METALS USING GENERALIZED GREEN'S FUNCTIONS

4.1 Theoretical Considerations

4.1.1 Fundamentals for Porous Metals

Here the fundamentals for porous metals include the kinematic approach of the Hill-Mandel homogenization theory and the limit analysis theory [100]. Consider a typical RVE for porous metals consisting of a matrix and several voids (see Fig. 4.1). Let Ω and ω denote the total domain and the domain occupied by the voids, respectively. For notational convenience, also let Ω and ω denote their respective volumes. The porosity, f, hereby equals ω/Ω. In this part, the RVE is set to be subject to kinematic boundary conditions on the external surface of the domain, $\partial\Omega$, that is,

$$\mathbf{v} = \mathbf{D} \cdot \mathbf{x} \quad \text{on } \partial\Omega, \tag{4.1}$$

where $\mathbf{v}$ denotes the microscopic velocity field and $\mathbf{D}$ denotes a specified second order symmetric tensor. Therefore, the kinematic approach of the Hill-Mandel homogenization theory is used. The macroscopic stress, $\boldsymbol{\Sigma}$, is defined as the volume average of the microscopic stress, $\boldsymbol{\sigma}$, that is,

$$\boldsymbol{\Sigma} = \langle \boldsymbol{\sigma} \rangle_{\Omega}, \tag{4.2}$$

where $\langle \cdot \rangle_\Omega$ denotes the volume average of the quantity. $\mathbf{D}$ can be proven to equal the volume average of the microscopic rate of deformation, $\mathbf{d}$, using Green's theorem, that is,

$$\mathbf{D} = \langle \mathbf{d} \rangle_\Omega. \tag{4.3}$$

As a result of Eq. (4.3), $\mathbf{D}$ is named as the macroscopic rate of deformation.

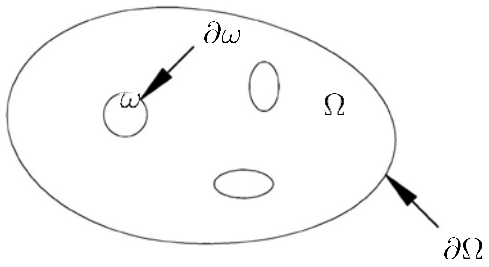

Fig. 4.1. Sketch of a typical RVE for porous metals.

Now let $\mathbf{v}$ be a kinematically admissible velocity field, that is, $\mathbf{v}$ satisfies Eq. (4.1). Also let $\boldsymbol{\sigma}$ be a statically admissible stress field, that is, $\boldsymbol{\sigma}$ is self equilibrating and satisfies the traction free boundary conditions on the void surface, $\partial\omega$. The Hill-Mandel lemma [76, 77] can hereby be stated as

$$\langle \boldsymbol{\sigma} : \mathbf{d} \rangle_\Omega = \boldsymbol{\Sigma} : \mathbf{D}. \tag{4.4}$$

It is worth notice that, in this lemma, $\boldsymbol{\sigma}$ and $\mathbf{d}$ do not need to be related through a constitutive relation.

Assume the matrix to be rigid-perfectly plastic and also to obey a certain yield criterion along with an associated flow rule. For a given matrix deformation field $\mathbf{d}$, the microscopic plastic dissipation is defined as

$$\pi(\mathbf{d}) = \sup_{\boldsymbol{\sigma}^* \in \mathcal{C}} \boldsymbol{\sigma}^* : \mathbf{d}, \tag{4.5}$$

where the supremum is taken over all the microscopic stresses which fall within the microscopic convex domain of elasticity $\mathcal{C}$. The Hill-Mandel lemma, together with Eq. (4.5), implies that

$$\boldsymbol{\Sigma} : \mathbf{D} \leq \Pi(\mathbf{D}) = \inf_{\mathbf{d} \in \mathcal{K}(\mathbf{D})} \langle \pi(\mathbf{d}) \rangle_{\Omega}, \tag{4.6}$$

where $\mathcal{K}(\mathbf{D})$ denotes the set of kinematically admissible microscopic deformation and takes the form of

$$\mathcal{K}(\mathbf{D}) = \left\{ \mathbf{d} \,\middle|\, \exists \mathbf{v},\ \forall \mathbf{x} \in \Omega,\ \mathbf{d} = \frac{1}{2}(\mathbf{v}\nabla + \nabla\mathbf{v}) \text{ and } \forall \mathbf{x} \in \partial\Omega,\ \mathbf{v} = \mathbf{D} \cdot \mathbf{x} \right\}. \tag{4.7}$$

Following Suquet [78], $\Pi(\mathbf{D})$ is defined as the macroscopic plastic dissipation. The classical limit analysis theory states that, as a result of Eq. (4.6), the macroscopic yield surface is defined by

$$\boldsymbol{\Sigma} = \frac{\partial \Pi}{\partial \mathbf{D}}(\mathbf{D}). \tag{4.8}$$

So far, the fundamentals for porous metals have been briefly reviewed. Eq. (4.5) – (4.8) implies that, to determine the macroscopic yield surface, several items need to be specified, i.e., the RVE geometry, the trial microscopic velocity fields defining $\mathcal{K}(\mathbf{D})$, and the microscopic plastic model defining $\pi(\mathbf{d})$. In the following sections, these items will be specified so as to determine the macroscopic yield surfaces of porous metals with cylindrical and spherical voids.

4.1.2 Basic Ideas

Consider an isolated cylindrical void in an infinite medium. Assume the matrix to be rigid-perfectly plastic and also to obey the von Mises criterion along with the associated flow rule. Also assume the matrix to be under generalized plane strain conditions. Tracey [73] proposed a family of microscopic velocity fields consisting of two parts: the first part accounts for purely deviatoric uniform extension; the second part accounts for dilatational expansion. Introduce a cylindrical coordinate system $Or\theta z$ with the origin, O, located at the axis of the cylindrical void and the z axis parallel to the axis of the cylindrical void. The family of microscopic velocity field takes the form of

$$\mathbf{v} = \mathbf{v}^A + \mathbf{v}^B, \tag{4.9}$$

where

$$\mathbf{v}^A = A\boldsymbol{\alpha} \cdot \mathbf{r}, \quad \mathbf{v}^B = \frac{B}{2\pi r}\mathbf{e}_r, \tag{4.10}$$

where $\mathbf{r}$ denotes the radius vectors of an arbitrary point. Also introduce a Cartesian coordinate system $Oxyz$ with the origin, O, and the z axis coincide with those of the coordinate system $Or\theta z$. Let $\boldsymbol{\alpha}$ take the form of

$$\boldsymbol{\alpha} = \mathbf{e}_x \otimes \mathbf{e}_x + \rho\mathbf{e}_y \otimes \mathbf{e}_y - (1+\rho)\,\mathbf{e}_z \otimes \mathbf{e}_z. \tag{4.11}$$

Here $\boldsymbol{\alpha}$ is traceless second order tensor with its principal directions parallel to the x, y, and z axes, and $\rho = D^A_{yy}/D^A_{xx}$, where $\mathbf{D}^A$ is the macroscopic rate of deformation due to $\mathbf{v}^A$. In Eq. (4.10), $\mathbf{v}^B$ accounts for in-plane isotropic radial expansion and equals

$(B/2\pi r)\,\mathbf{e}_r$ rather than $(B/r)\,\mathbf{e}_r$ proposed by Tracey [73]. Further investigation will show that, by letting $\mathbf{v}^B$ equal $(B/2\pi r)\,\mathbf{e}_r$, B can have a clear physical meaning.

Eq. (4.10) indicates that $\mathbf{v}^B$ is irrotational and can be represented as the gradient of a velocity potential. Let Φ^B denote the velocity potential for $\mathbf{v}^B$. Φ^B can be related to $\mathbf{v}^B$ by

$$\mathbf{v}^B = \boldsymbol{\nabla}\Phi^B = \frac{\partial \Phi^B}{\partial r}\mathbf{e}_r. \tag{4.12}$$

Substituting Eq. (4.10) into Eq. (4.12) and solving for Φ^B give

$$\Phi^B = \frac{B}{2\pi}\ln r. \tag{4.13}$$

Eq. (4.13) reminds one of the Green's function on an unbounded two-dimensional domain with the point source located at the origin, that is,

$$G\left(\mathbf{r};\mathbf{0}\right) = \frac{1}{2\pi}\ln r. \tag{4.14}$$

The appendix gives the derivations for the Green's and generalized Green's functions on different domains. It can be seen that Φ^B can be related to $G\left(\mathbf{r};\mathbf{0}\right)$ by

$$\Phi^B = BG\left(\mathbf{r};\mathbf{0}\right). \tag{4.15}$$

Note that, even if the matrix is incompressible, the porous medium is not. This is because the void is compressible. In fact, the void acts as the source of dilatation. This leads one to treat the void as a perturbation to the perfect medium. Accordingly, one can also treat $\mathbf{v}^A$ as the unperturbed velocity field in the perfect medium and $\mathbf{v}^B$ as the perturbing velocity field due to the void. In addition, one can further idealize the void as a point source located at the origin. According to the definition of the point source, now

$\nabla \cdot \mathbf{v}^B$ should be singular at the origin but vanish elsewhere. Therefore, Φ^B should satisfy the two-dimensional Poisson's equation

$$\nabla \cdot \nabla \Phi^B = B\delta(\mathbf{r} - \mathbf{0}) \tag{4.16}$$

on the unbounded domain. By comparing Eq. (4.16) with Eq. (C1.2) (see Appendix C1), one can relate Φ^B to $G(\mathbf{r}; \mathbf{0})$ as described in Eq. (4.15). In addition, Eq. (4.16) also indicates that B is mathematically the point source intensity. Note that the void acts as the source of dilatation. This implies that B is physically a measure of the void dilatation.

So far, it has been proven valid to use the Green's function to find the perturbing velocity potential so as to provide the microscopic velocity field in an infinite medium with an isolated void. This leads one to use generalized Green's functions to also find the perturbing velocity potentials so as to provide the microscopic velocity fields in porous metals with different void shapes and distributions.

4.1.3 RVE Geometry and Microscopic Velocity Fields for Porous Metals with Cylindrical Voids

Note that, by definition, the RVEs should be able to form into a continuum. Therefore, it is better for a RVE to be a tetrahedron, a parallelepiped, a hexagonal prism, or etc. Consider a hollow cuboid RVE of width b, length c, and thickness d (not shown) and with a cylindrical void of radius a located at its center (see Fig. 4.2). Here b can be set not to equal c so as to incorporate the void distribution anisotropy. Still assume the

matrix to be rigid-perfectly plastic and also to obey the von Mises criterion along with the associated flow rule. Also assume the matrix to be under generalized plane strain conditions. Introduce a Cartesian coordinate system $Oxyz$ with the origin, O, located at the center of the void and the x, y, and z axes parallel to the length, width and height directions of the RVE, respectively.

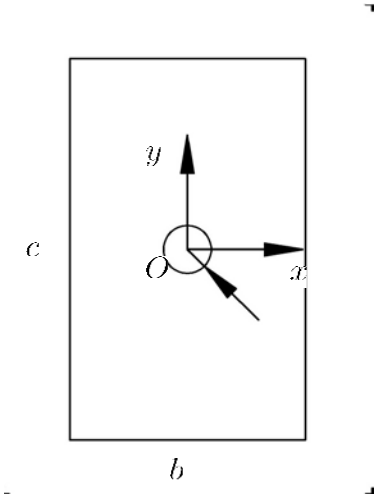

Fig. 4.2. Sketch of the RVE for porous metals with cylindrical voids.

Let the family of microscopic velocity fields take the form of

$$\mathbf{v} = \mathbf{v}^A + \mathbf{v}^B, \tag{4.17}$$

where

$$\mathbf{v}^A = A\boldsymbol{\alpha} \cdot \mathbf{r}, \quad \mathbf{v}^B = \nabla \Phi^B. \tag{4.18}$$

Let $\boldsymbol{\alpha}$ still take the form of

$$\boldsymbol{\alpha} = \mathbf{e}_x \otimes \mathbf{e}_x + \rho \mathbf{e}_y \otimes \mathbf{e}_y - (1+\rho)\, \mathbf{e}_z \otimes \mathbf{e}_z. \tag{4.19}$$

Note that the matrix is under generalized plane strain conditions. This implies that $\mathbf{v}^B$ and Φ^B are still independent of z. Still idealize the void as a point source located at the origin. Φ^B should hereby satisfy the two-dimensional Poisson's equation

$$\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \Phi^B = B\delta\left(x\right)\delta\left(y\right) \tag{4.20}$$

within the RVE.

Here let $\mathbf{v}^B$ account for in-plane equal biaxial expansion. This implies that $\mathbf{v}^B \cdot \mathbf{n}$ has to be constant across each external RVE edge, where $\mathbf{n}$ denotes the unit normal to the edge. In addition, due to symmetry,

$$v_x^B\left(\frac{b}{2}, y\right) = -v_x^B\left(-\frac{b}{2}, y\right), \quad v_y^B\left(x, \frac{c}{2}\right) = -v_y^B\left(x, -\frac{c}{2}\right), \tag{4.21}$$

while due to in-plane equal biaxial expansion,

$$v_x^B\left(\frac{b}{2}, y\right)\Big/\frac{b}{2} = v_y^B\left(x, \frac{c}{2}\right)\Big/\frac{c}{2} = C_1, \tag{4.22}$$

where C_1 is a constant to be determined. Note that here

$$v_x^B = \frac{\partial \Phi^B}{\partial x}, \quad v_y^B = \frac{\partial \Phi^B}{\partial y}. \tag{4.23}$$

Combining Eqs.(4.21), (4.22), and (4.23) gives

$$\begin{aligned} -\left.\frac{\partial \Phi^B}{\partial x}\right|_{x=-\frac{b}{2}} &= \left.\frac{\partial \Phi^B}{\partial x}\right|_{x=\frac{b}{2}} = \frac{C_1 b}{2}, \\ -\left.\frac{\partial \Phi^B}{\partial y}\right|_{y=-\frac{c}{2}} &= \left.\frac{\partial \Phi^B}{\partial y}\right|_{y=\frac{c}{2}} = \frac{C_1 c}{2}. \end{aligned} \tag{4.24}$$

which are the boundary conditions associated with Eq. (4.20). However, in Eq. (4.24), C_1 is still undetermined.

To determine C_1, again compare Φ^B with its corresponding generalized Green's function, where the term "generalized" indicates that nonhomogeneous boundary conditions are applied. This generalized Green's function should satisfy the two-dimensional Poisson's equation

$$\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} G\left(x, y; 0, 0\right) = \delta\left(x\right)\delta\left(y\right) \tag{4.25}$$

within the RVE. For notational convenience, let G denote $G\left(x, y; 0, 0\right)$ in this and the next sections. By comparing Eq. (4.20) with Eq. (4.25), one can relate Φ^B to G by

$$\Phi^B = BG \tag{4.26}$$

and also obtain the boundary conditions associated with Eq. (4.26) as

$$\begin{aligned} -\left.\frac{\partial G}{\partial x}\right|_{x=-\frac{b}{2}} &= \left.\frac{\partial G}{\partial x}\right|_{x=\frac{b}{2}} = \frac{C_1 b}{2B}, \\ -\left.\frac{\partial G}{\partial y}\right|_{y=-\frac{c}{2}} &= \left.\frac{\partial G}{\partial y}\right|_{y=\frac{c}{2}} = \frac{C_1 c}{2B}. \end{aligned} \tag{4.27}$$

It can be proven that G can exist only if $\mathbf{n} \cdot \nabla G$ satisfies

$$\int_{\Omega} \delta\left(x\right)\delta\left(y\right) d\Omega = \int_{\partial\Omega} \mathbf{n} \cdot \boldsymbol{\nabla} G dS \tag{4.28}$$

[101]. Substituting Eqs. (4.25) and (4.27) into Eq. (4.28) gives

$$C_1 = \frac{B}{2bc}. \tag{4.29}$$

Now the boundary conditions are fully specified, while G is uniquely determined. Appendix C1 gives the derivation for G, while Eq. (C1.29) gives the expression for G. Substituting Eq. (C1.29) into Eq. (4.26) gives the expression for Φ^B. The family of microscopic velocity fields for porous metals with cylindrical voids can be subsequently obtained.

Eqs. (4.26) and (4.28) imply that $\mathbf{n}\cdot\boldsymbol{\nabla}\Phi^B$ should satisfy

$$\int_{\Omega} B\delta\left(x\right)\delta\left(y\right)d\Omega=\int_{\partial\Omega}\mathbf{n}\cdot\boldsymbol{\nabla}\Phi^B dS. \tag{4.30}$$

Eq. (4.30) is quite similar to Eq. (4.28) and has its clear physical meaning. In fact, in Eq. (4.30), the integral to the left of the equal sign represents the void dilatation, while that to the right represents the RVE dilatation. Eq. (4.30) requires these two integrals to be equal and hereby physically means that the matrix is incompressible. Similarly, Eq. (4.28) also physically means that the matrix is incompressible.

4.1.4 Constitutive Relations for Porous Metals with Cylindrical Voids

Note that

$$\mathbf{d}=\frac{1}{2}\left(\mathbf{v}\boldsymbol{\nabla}+\boldsymbol{\nabla}\mathbf{v}\right) \tag{4.31}$$

by definition. Substituting Eq. (4.31) into Eq. (4.3) and applying the Gauss theorem to the equation give

$$\mathbf{D}=\frac{1}{\Omega}\int_{\Omega}\mathbf{d}d\Omega=\frac{1}{2\Omega}\int_{\Omega}\left(\mathbf{v}\boldsymbol{\nabla}+\boldsymbol{\nabla}\mathbf{v}\right)d\Omega=\frac{1}{2\Omega}\int_{\partial\Omega}\left(\mathbf{v}\cdot\mathbf{n}+\mathbf{n}\cdot\mathbf{v}\right)dS. \tag{4.32}$$

Substituting Eqs. (4.24) and (4.29) into Eq. (4.32) gives

$$D_{xx}=A+\frac{B}{2bc},\quad D_{yy}=A\rho+\frac{B}{2bc},\quad D_{zz}=-A\left(1+\rho\right),$$
$$D_{xy}=D_{yz}=D_{zx}=0. \tag{4.33}$$

Separate $\mathbf{D}$ into its deviatoric and mean parts, i.e., $\mathbf{D}'$ and $\mathbf{D}_m$. $\mathbf{D}'$ and $\mathbf{D}_m$ can hereby be expressed as

$$\mathbf{D}' = A\boldsymbol{\alpha} + \frac{B}{6bc}\boldsymbol{\beta}, \quad \mathbf{D}_m = \frac{B}{3bc}\mathbf{I}, \tag{4.34}$$

where $\boldsymbol{\alpha}$ is given by Eq. (4.19);

$$\boldsymbol{\beta} = \mathbf{e}_x \otimes \mathbf{e}_x + \mathbf{e}_y \otimes \mathbf{e}_y - 2\mathbf{e}_z \otimes \mathbf{e}_z \tag{4.35}$$

here; $\mathbf{I}$ denotes the unit second order tensor. Accordingly, the macroscopic effective and mean rates of deformation, i.e., D_e and D_m, can be obtained as

$$D_e = \sqrt{\frac{2}{3}\mathbf{D}' : \mathbf{D}'}, \quad D_m = \frac{B}{3bc}. \tag{4.36}$$

Also separate $\boldsymbol{\Sigma}$ into its deviatoric and mean parts, i.e., $\boldsymbol{\Sigma}'$ and $\boldsymbol{\Sigma}_m$. Note that

$$\Pi = \boldsymbol{\Sigma} : \mathbf{D} = \boldsymbol{\Sigma}' : \mathbf{D}' + \boldsymbol{\Sigma}_m : \mathbf{D}_m. \tag{4.37}$$

$\boldsymbol{\Sigma}'$ and $\boldsymbol{\Sigma}_m$ can hereby be expressed as

$$\boldsymbol{\Sigma}' = \frac{\partial \Pi}{\partial \mathbf{D}'}, \quad \boldsymbol{\Sigma}_m = \frac{\partial \Pi}{\partial \mathbf{D}_m}. \tag{4.38}$$

For a matrix obeying the von Mises criterion along with the associated flow rule,

$$\frac{\partial \Pi}{\partial \mathbf{D}'} = \lambda_1 \mathbf{D}', \tag{4.39}$$

$$\frac{\partial \Pi}{\partial \mathbf{D}_m} = \lambda_2 \mathbf{D}_m = \lambda_2 \frac{B}{3bc}\mathbf{I}, \tag{4.40}$$

where λ_1 and λ_2 are positive scalar-valued functions. Following Keralavarma and Benzerga [92], use the change of variables $(\mathbf{D}', \mathbf{D}_m) \rightarrow (A, B)$ to determine λ_1 and λ_2 and hereby $\boldsymbol{\Sigma}'$ and $\boldsymbol{\Sigma}_m$. Express $\partial\Pi/\partial A$ and $\partial\Pi/\partial B$ in terms of $\partial\Pi/\partial\mathbf{D}'$ and $\partial\Pi/\partial\mathbf{D}_m$ using the chain rule as

$$\frac{\partial \Pi}{\partial A} = \frac{\partial \Pi}{\partial \mathbf{D}'} : \frac{\partial \mathbf{D}'}{\partial A} + \frac{\partial \Pi}{\partial \mathbf{D}_m} : \frac{\partial \mathbf{D}_m}{\partial A}, \tag{4.41}$$

$$\frac{\partial \Pi}{\partial B} = \frac{\partial \Pi}{\partial \mathbf{D}'} : \frac{\partial \mathbf{D}'}{\partial B} + \frac{\partial \Pi}{\partial \mathbf{D}_m} : \frac{\partial \mathbf{D}_m}{\partial B}. \tag{4.42}$$

Substituting Eq. (4.34) into Eqs. (4.41) and (4.42) gives

$$\frac{\partial \Pi}{\partial A} = \frac{\partial \Pi}{\partial \mathbf{D}'} : \boldsymbol{\alpha}, \tag{4.43}$$

$$\frac{\partial \Pi}{\partial B} = \frac{1}{6bc}\frac{\partial \Pi}{\partial \mathbf{D}'} : \boldsymbol{\beta} + \frac{1}{3bc}\frac{\partial \Pi}{\partial \mathbf{D}_m} : \mathbf{I}. \tag{4.44}$$

Substituting Eq. (4.39) into Eq. (4.43) gives

$$\frac{\partial \Pi}{\partial A} = \lambda_1 \mathbf{D}' : \boldsymbol{\alpha}. \tag{4.45}$$

Therefore,

$$\lambda_1 = \frac{\dfrac{\partial \Pi}{\partial A}}{\mathbf{D}' : \boldsymbol{\alpha}}, \tag{4.46}$$

and

$$\frac{\partial \Pi}{\partial \mathbf{D}'} = \frac{\dfrac{\partial \Pi}{\partial A}}{\mathbf{D}' : \boldsymbol{\alpha}}\mathbf{D}'. \tag{4.47}$$

Substituting Eq. (4.40) and (4.47) into Eq. (4.44) gives

$$\frac{\partial \Pi}{\partial B} = \frac{1}{6bc}\frac{\dfrac{\partial \Pi}{\partial A}}{\mathbf{D}' : \boldsymbol{\alpha}}\mathbf{D}' : \boldsymbol{\beta} + \frac{1}{3bc}\lambda_2 \frac{B}{3bc}\mathbf{I} : \mathbf{I}. \tag{4.48}$$

Therefore,

$$\lambda_2 = \frac{bc\dfrac{\partial \Pi}{\partial B} - \dfrac{1}{6}\dfrac{\mathbf{D}' : \boldsymbol{\beta}}{\mathbf{D}' : \boldsymbol{\alpha}}\dfrac{\partial \Pi}{\partial A}}{\dfrac{B}{3bc}}, \tag{4.49}$$

and

$$\frac{\partial \Pi}{\partial \mathbf{D}_m} = \frac{bc\dfrac{\partial \Pi}{\partial B} - \dfrac{1}{6}\dfrac{\mathbf{D}' : \boldsymbol{\beta}}{\mathbf{D}' : \boldsymbol{\alpha}}\dfrac{\partial \Pi}{\partial A}}{\dfrac{B}{3bc}}\frac{B}{3bc}\mathbf{I} = \left(bc\frac{\partial \Pi}{\partial B} - \frac{1}{6}\frac{\mathbf{D}' : \boldsymbol{\beta}}{\mathbf{D}' : \boldsymbol{\alpha}}\frac{\partial \Pi}{\partial A}\right)\mathbf{I}. \tag{4.50}$$

Substituting Eqs. (4.47) and (4.50) into Eq. (4.38) gives

$$\boldsymbol{\Sigma}' = \frac{\dfrac{\partial \Pi}{\partial A}}{\mathbf{D}' : \boldsymbol{\alpha}}\mathbf{D}', \quad \boldsymbol{\Sigma}_m = \left(bc\frac{\partial \Pi}{\partial B} - \frac{1}{6}\frac{\mathbf{D}' : \boldsymbol{\beta}}{\mathbf{D}' : \boldsymbol{\alpha}}\frac{\partial \Pi}{\partial A}\right)\mathbf{I}, \tag{4.51}$$

which are the constitutive relations for porous metals with cylindrical voids. Accordingly, the macroscopic effective and mean stresses, i.e., Σ_e and Σ_m, can be obtained as

$$\Sigma_e = \sqrt{\frac{3}{2}\boldsymbol{\Sigma}' : \boldsymbol{\Sigma}'} = \sqrt{\frac{3}{2}}\frac{\sqrt{\mathbf{D}' : \mathbf{D}'}}{\mathbf{D}' : \boldsymbol{\alpha}}\frac{\partial \Pi}{\partial A}, \quad \Sigma_m = bc\frac{\partial \Pi}{\partial B} - \frac{1}{6}\frac{\mathbf{D}' : \boldsymbol{\beta}}{\mathbf{D}' : \boldsymbol{\alpha}}\frac{\partial \Pi}{\partial A}. \tag{4.52}$$

The values of $\partial\Pi/\partial A$ and $\partial\Pi/\partial B$ can be numerically calculated. Note that $\boldsymbol{\sigma}$ vanishes within the void. Therefore,

$$\Pi = \frac{1}{\Omega}\int_{\Omega-\omega}\sigma_Y d_e d\Omega = \frac{\sigma_Y}{\Omega}\int_{\Omega-\omega} d_e d\Omega, \tag{4.53}$$

and

$$\frac{\partial \Pi}{\partial A} = \frac{\sigma_Y}{\Omega}\int_{\Omega-\omega}\frac{\partial d_e}{\partial A}d\Omega, \quad \frac{\partial \Pi}{\partial B} = \frac{\sigma_Y}{\Omega}\int_{\Omega-\omega}\frac{\partial d_e}{\partial B}d\Omega, \tag{4.54}$$

where $\Omega - \omega$ denotes the domain occupied by the matrix. For notational convenience, let $(\cdot)_{,x}$ denote $\partial(\cdot)/\partial x$. Substituting Eqs. (4.17) – (4.19), and (4.26) into Eq. (4.31) gives

$$\begin{gathered} d_{xx} = A + BG_{,xx}, \quad d_{yy} = A\rho + BG_{,yy}, \quad d_{zz} = -A(1+\rho), \\ d_{xy} = BG_{,xy}, \quad d_{yz} = d_{zx} = 0, \end{gathered} \tag{4.55}$$

where $G_{,xx}$, $G_{,yy}$, and $G_{,xy}$ can be numerically calculated as described in Appendix. Also note that

$$d_e = \sqrt{\frac{2}{3}\left(d_{xx}^2 + d_{yy}^2 + d_{zz}^2 + 2d_{xy}^2 + 2d_{yz}^2 + 2d_{zx}^2\right)} \tag{4.56}$$

by definition. Substituting Eq. (4.55) into Eq. (4.56) and noting that $G_{,yy} = -G_{,xx}$ give

$$d_e = \sqrt{\frac{2}{3}\left[\begin{array}{c} 2A^2\left(1+\rho+\rho^2\right) + 2AB\left(1-\rho\right)G_{,xx} \\ +B^2\left(G_{,xx}^2 + G_{,yy}^2 + 2G_{,xy}^2\right) \end{array}\right]}. \tag{4.57}$$

$\partial d_e/\partial A$ and $\partial d_e/\partial B$ can hereby be obtained as

$$\frac{\partial d_e}{\partial A} = \frac{2}{3d_e}\left[2A\left(1+\rho+\rho^2\right) + B\left(1-\rho\right)G_{,xx}\right], \tag{4.58}$$

$$\frac{\partial d_e}{\partial B} = \frac{2}{3d_e}\left[A\left(1-\rho\right)G_{,xx} + B\left(G_{,xx}^2 + G_{,yy}^2 + 2G_{,xy}^2\right)\right]. \tag{4.59}$$

Substituting Eqs. (4.58) and (4.59) into Eq. (4.54) and numerically calculating the integrals give the values of $\partial\Pi/\partial A$ and $\partial\Pi/\partial B$.

4.1.5 RVE Geometry and Microscopic Velocity Fields for Porous Metals with Spherical Voids

Consider a hollow cuboid RVE of width b, length c, and height d and with a spherical void of radius a located at its center (see Fig. 4.3). Here b, c, and d can be set not to equal each other so as to incorporate the void distribution anisotropy. Still assume the matrix to be rigid-perfectly plastic and also to obey the von Mises criterion along with the associated flow rule. Introduce a Cartesian coordinate system $Oxyz$ with the origin, O, located at the center of the void and the x, y, and z axes parallel to the length, width, and height directions of the RVE, respectively.

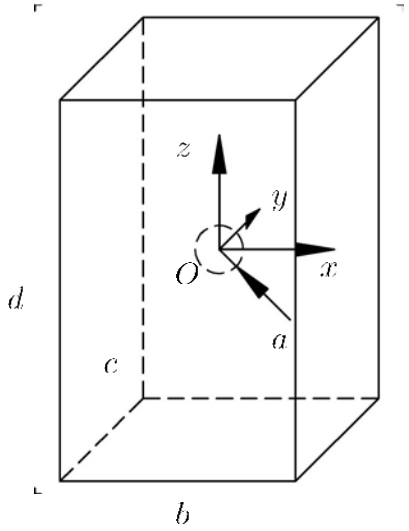

Fig. 4.3. Sketch of the RVE for porous metals with spherical voids.

Let the family of microscopic velocity fields take the form of

$$\mathbf{v} = \mathbf{v}^A + \mathbf{v}^B, \tag{4.60}$$

where

$$\mathbf{v}^A = A\boldsymbol{\alpha} \cdot \mathbf{r}, \quad \mathbf{v}^B = \nabla \Phi^B. \tag{4.61}$$

Let $\boldsymbol{\alpha}$ still take the form of

$$\boldsymbol{\alpha} = \mathbf{e}_x \otimes \mathbf{e}_x + \rho \mathbf{e}_y \otimes \mathbf{e}_y - (1+\rho)\, \mathbf{e}_z \otimes \mathbf{e}_z. \tag{4.62}$$

Note that the matrix is no longer under generalized plane strain conditions. This implies that here $\mathbf{v}^B$ and Φ^B depend on z. Still idealize the void as a point source located at the origin. Φ^B should hereby satisfy the three-dimensional Poisson's equation

$$\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} \Phi^B = B\delta(x)\,\delta(y)\,\delta(z) \tag{4.63}$$

within the RVE.

Here let $\mathbf{v}^B$ account for equal triaxial extension. This implies that $\mathbf{v}^B \cdot \mathbf{n}$ has to be constant across each external RVE surface, where $\mathbf{n}$ denotes the unit normal to the surface. In addition, due to symmetry,

$$v_x^B\left(-\frac{b}{2},y,z\right)=-v_x^B\left(\frac{b}{2},y,z\right),\quad v_y^B\left(x,-\frac{c}{2},z\right)=-v_y^B\left(x,\frac{c}{2},z\right),$$
$$v_z^B\left(x,y,-\frac{d}{2}\right)=-v_z^B\left(x,y,\frac{d}{2}\right), \tag{4.64}$$

while due to equal triaxial extension,

$$v_x^B\left(\frac{b}{2},y,z\right)\Big/\frac{b}{2}=v_y^B\left(x,\frac{c}{2},z\right)\Big/\frac{c}{2}=v_z^B\left(x,y,\frac{d}{2}\right)\Big/\frac{d}{2}=C_2, \tag{4.65}$$

where C_2 is a constant to be determined. Note that here

$$v_x^B=\frac{\partial\Phi^B}{\partial x},\quad v_y^B=\frac{\partial\Phi^B}{\partial y},\quad v_z^B=\frac{\partial\Phi^B}{\partial z}. \tag{4.66}$$

Combining Eqs. (4.64), (4.65), and (4.66) gives

$$\begin{aligned}
-\left.\frac{\partial\Phi^B}{\partial x}\right|_{x=-\frac{b}{2}}&=\left.\frac{\partial\Phi^B}{\partial x}\right|_{x=\frac{b}{2}}=\frac{C_2 b}{2},\\
-\left.\frac{\partial\Phi^B}{\partial y}\right|_{y=-\frac{c}{2}}&=\left.\frac{\partial\Phi^B}{\partial y}\right|_{y=\frac{c}{2}}=\frac{C_2 c}{2},\\
-\left.\frac{\partial\Phi^B}{\partial z}\right|_{z=-\frac{d}{2}}&=\left.\frac{\partial\Phi^B}{\partial z}\right|_{z=\frac{d}{2}}=\frac{C_2 d}{2},
\end{aligned} \tag{4.67}$$

which are the boundary conditions associated with Eq. (4.63).

Similarly to Section 4.1.3, $\mathbf{n}\cdot\boldsymbol{\nabla}\Phi^B$ should satisfy

$$\int_\Omega B\delta(x)\,\delta(y)\,\delta(z)\,d\Omega=\int_{\partial\Omega}\mathbf{n}\cdot\boldsymbol{\nabla}\Phi^B dS. \tag{4.68}$$

Substituting Eqs. (4.63) and (4.67) into Eq. (4.68) gives

$$C_2 = \frac{B}{3bcd}. \tag{4.69}$$

Now the boundary conditions are fully specified, while Φ^B is uniquely determined. Similarly to Section 4.1.3, one can still Φ^B relate to its corresponding generalized Green's function by

$$\Phi^B = BG\left(x, y, z; 0, 0, 0\right). \tag{4.70}$$

For notational convenience, let G denote $G\left(x, y, z; 0, 0, 0\right)$ in this and the next sections. G should satisfy the three-dimensional Poisson's equation

$$\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} G\left(x, y, z; 0, 0, 0\right) = \delta\left(x\right)\delta\left(y\right)\delta\left(z\right) \tag{4.71}$$

subject to the nonhomogeneous boundary conditions. Appendix C1 gives the derivation for G, while Eq. (C1.42) gives the expression for G. Substituting (C1.42) into Eq. (4.70) gives the expression for Φ^B. The family of microscopic velocity fields for porous metals with spherical voids can be subsequently obtained.

4.1.6 Constitutive Relations for Porous Metals with Spherical Voids

Subsitituting Eqs. (4.67) and (4.69) into (4.32) gives

$$D_{xx} = A + \frac{B}{3bcd}, \quad D_{yy} = A\rho + \frac{B}{3bcd}, \quad D_{zz} = -A\left(1+\rho\right) + \frac{B}{3bcd},$$
$$D_{xy} = D_{yz} = D_{zx} = 0. \tag{4.72}$$

Separate $\mathbf{D}$ into $\mathbf{D}'$ and $\mathbf{D}_m$. $\mathbf{D}'$ and $\mathbf{D}_m$ can hereby be expressed as

$$\mathbf{D}' = A\boldsymbol{\alpha}, \quad \mathbf{D}_m = \frac{B}{3bcd}\mathbf{I}. \tag{4.73}$$

Accordingly, D_e and D_m can be obtained as

$$D_e = A\sqrt{\frac{2}{3}\boldsymbol{\alpha} : \boldsymbol{\alpha}}, \quad D_m = \frac{B}{3bcd}. \tag{4.74}$$

Also separate $\boldsymbol{\Sigma}$ into $\boldsymbol{\Sigma}'$ and $\boldsymbol{\Sigma}_m$. Similarly to Section 4.1.4, the constitutive relations for porous metals with spherical voids can be finally obtained as

$$\boldsymbol{\Sigma}' = \frac{\dfrac{\partial \Pi}{\partial A}}{\boldsymbol{\alpha} : \boldsymbol{\alpha}}\boldsymbol{\alpha}, \quad \boldsymbol{\Sigma}_m = bcd\frac{\partial \Pi}{\partial B}\mathbf{I}. \tag{4.75}$$

Accordingly, Σ_e and Σ_m can be obtained as

$$\Sigma_e = \sqrt{\frac{3}{2}}\frac{\dfrac{\partial \Pi}{\partial A}}{\sqrt{\boldsymbol{\alpha} : \boldsymbol{\alpha}}}, \quad \Sigma_m = bcd\frac{\partial \Pi}{\partial B}. \tag{4.76}$$

The values of $\partial\Pi/\partial A$ and $\partial\Pi/\partial B$ can be numerically calculated. Substituting Eqs. (4.60) – (4.62) and (4.70) into Eq. (4.31) gives

$$\begin{gathered} d_{xx} = A + BG_{,xx}, \quad d_{yy} = A\rho + BG_{,yy}, \quad d_{zz} = -A\left(1+\rho\right) + BG_{,zz}, \\ d_{xy} = BG_{,xy}, \quad d_{yz} = BG_{,yz}, \quad d_{zx} = BG_{,zx}, \end{gathered} \tag{4.77}$$

where $G_{,xx}$, $G_{,yy}$, $G_{,zz}$, and etc. can be numerically calculated as described in Section 4.4.2. Substituting Eq. (4.77) into Eq. (4.56) and noting that $G_{,zz} = -G_{,xx} - G_{,yy}$ give

$$d_e = \sqrt{\frac{2}{3}\left\{\begin{array}{c} 2A^2\left(1+\rho+\rho^2\right) + 2AB\left[\left(2+\rho\right)G_{,xx} + \left(1+2\rho\right)G_{,yy}\right] \\ +B^2\left[G_{,xx}^2 + G_{,yy}^2 + G_{,zz}^2 + 2\left(G_{,xy}^2 + G_{,yz}^2 + G_{,zx}^2\right)\right] \end{array}\right\}}. \tag{4.78}$$

$\partial d_e/\partial A$ and $\partial d_e/\partial B$ can hereby be obtained as

$$\frac{\partial d_e}{\partial A} = \frac{2}{3d_e}\left\{2A\left(1+\rho+\rho^2\right) + B\left[\left(2+\rho\right)G_{,xx} + \left(1+2\rho\right)G_{,yy}\right]\right\}, \tag{4.79}$$

$$\begin{gathered} \frac{\partial d_e}{\partial B} = \frac{2}{3d_e}\Big\{A\left[\left(2+\rho\right)G_{,xx} + \left(1+2\rho\right)G_{,yy}\right] \\ +B^2\left[G_{,xx}^2 + G_{,yy}^2 + G_{,zz}^2 + 2\left(G_{,xy}^2 + G_{,yz}^2 + G_{,zx}^2\right)\right]\Big\}. \end{gathered} \tag{4.80}$$

Substituting Eqs. (4.79) and (4.80) into Eq. (4.54) and numerically calculating the integrals give the values of $\partial\Pi/\partial A$ and $\partial\Pi/\partial B$.

4.2 Results and Discussion

4.2.1 Effects of the Macroscopic Mean Stress and the Porosity for Porous Metals with Cylindrical Voids

In this section, let the RVE be subject to in-plane equal biaxial extension, and let its cross section be square. Fig. 4.4 plots Σ_e/σ_Y versus Σ_m/σ_Y for different values of f. Here the other parameters are set to be $b/c = 1$ and $\rho = 1$. As can be seen, as Σ_m/σ_Y increases, Σ_e/σ_Y decreases monotonically, and especially, as Σ_m/σ_Y approaches its upper limit, Σ_e/σ_Y decreases rapidly to zero. In addition, as f increases, the curve shifts inward. This physically means that, as the porosity increases, the material can sustain lower macroscopic effective and mean stresses. This is in agreement with the experimental and numerical observations. The results here are also quite similar to those obtained by Gurson [74]. However, further investigation will show that there still exists some discrepancy between the results here and those obtained by Gurson [74].

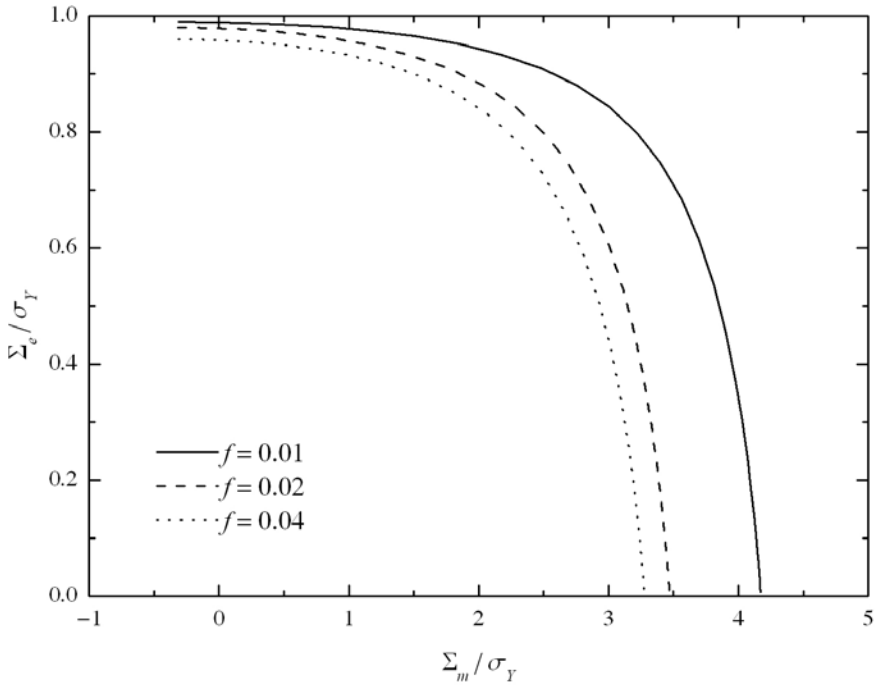

Fig. 4.4. Σ_e/σ_Y versus Σ_m/σ_Y for different values of f.

Fig. 4.5 shows the contour plots of the microscopic effective rate of deformation fields for different values of Σ_m/σ_Y. Here the other parameters are set to be $f = 0.02$, $b/c = 1$, $\rho = 1$, and $D_e = 0.01$. Here the RVE is divided into 100×100 elements and is hereby of 101×101 nodes. For given values of the parameters, the value of d_e at each node can be calculated from Eq. (4.57). A contour plot can hereby be plotted based on the calculated values of d_e. Note that here the porosity is set to be quite low. This causes the node density to be relatively low around the void. As a result, the illustrated void may not be strictly circular. This, however, has no effect on the field within the matrix. In addition, a logarithmic scale is used in each contour plot. As can be seen in Fig. 4.5,

the assumed microscopic effective rate of deformation fields are quite sophisticated: as Σ_m/σ_Y increases, the field exhibits different characteristics: for $\Sigma_m/\sigma_Y = 0$, the field is quite uniform, while d_e slightly concentrates around the void (see Fig. 4.5 (a)); as Σ_m/σ_Y increases, d_e concentrates more and more around the void and also at the ligaments (see Fig. 4.5 (b) and (c)); as Σ_m/σ_Y attains its upper limit, the field becomes highly nonuniform, while d_e heavily concentrates around the void and at the ligaments but diminishes at the corners (see Fig. 4.5 (d)).

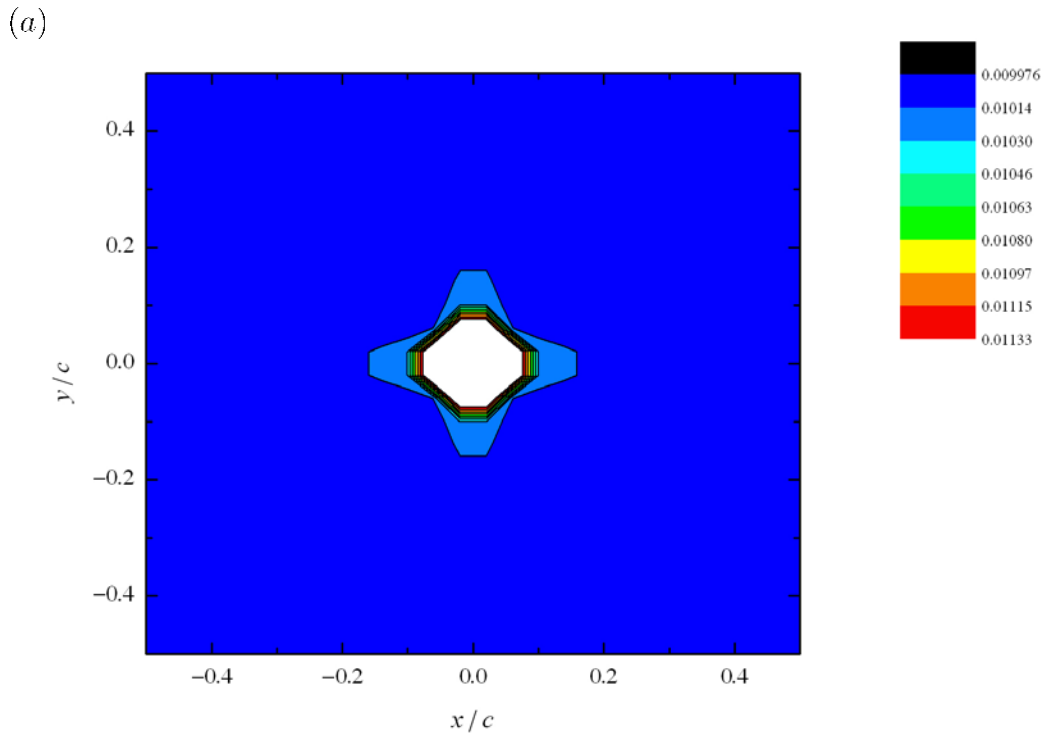

Fig. 4.5. Contour plots of the microscopic effective rate of deformation fields (a) for $\Sigma_m/\sigma_Y = 0$, (b) for $\Sigma_m/\sigma_Y = 1.1574$, (c) for $\Sigma_m/\sigma_Y = 2.3147$, and (d) for $\Sigma_m/\sigma_Y = 3.4721$.

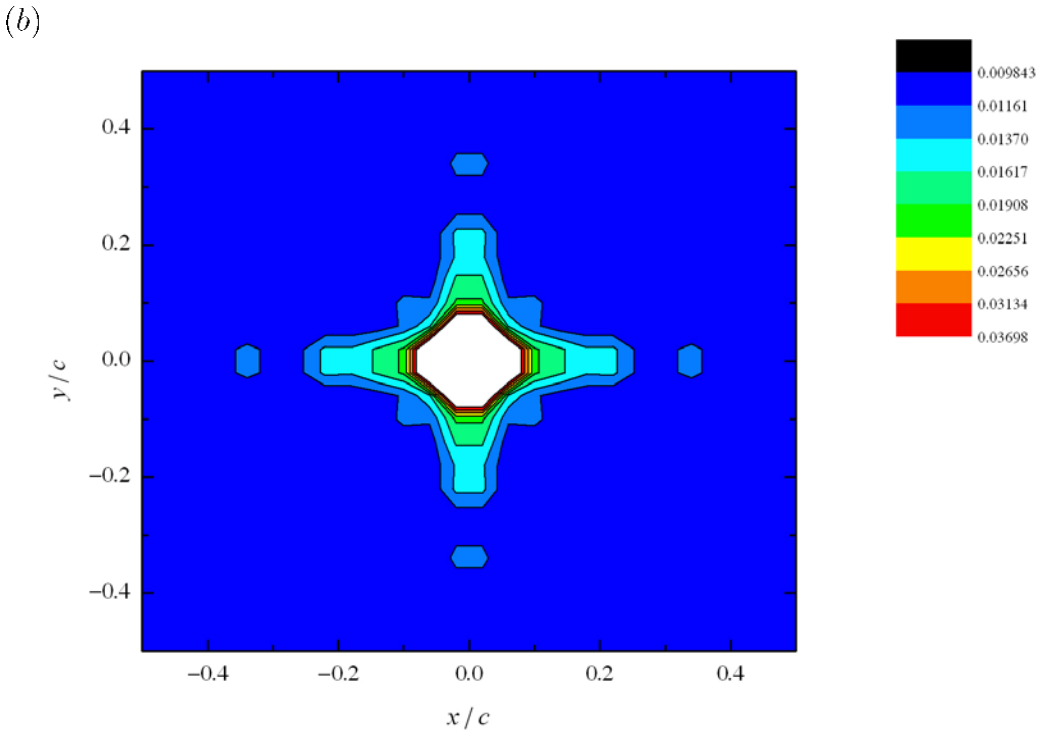

Fig. 4.5 Continued.

(c)

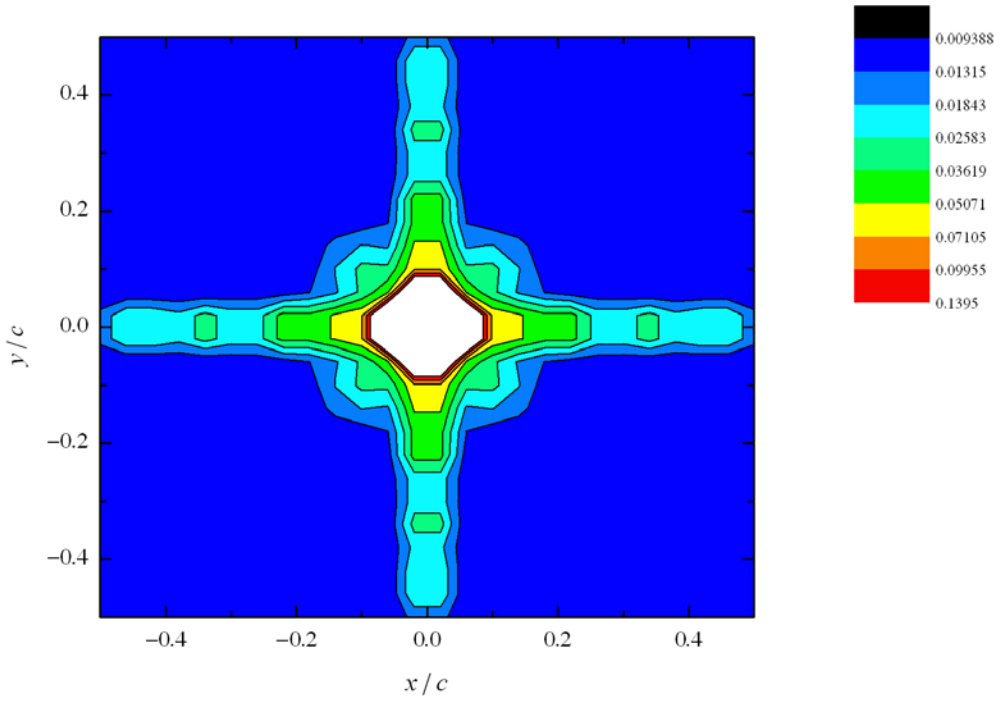

Fig. 4.5 Continued.

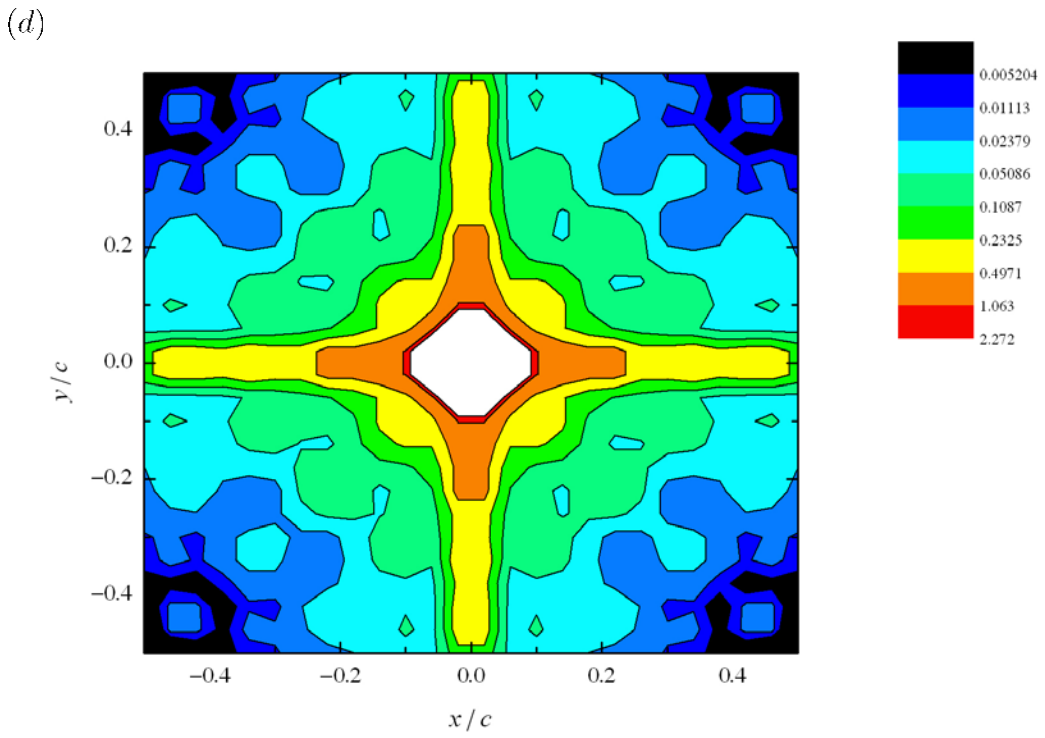

Fig. 4.5 Continued.

Fig. 4.5 (c) and Fig. 4.6 show the contour plots of the microscopic effective rate of deformation fields for different values of f. Here the other parameters are set to be $\Sigma_m/\sigma_Y = 2.3147$, $b/c = 1$, $\rho = 1$, and $D_e = 0.01$. Note that, as the porosity increases, the void becomes greater, while the ligament becomes narrower. As a result, in Fig. 4.5 (c) and Fig. 4.6, as f increases, d_e concentrates more and more around the void and at the ligaments. Fig. 4.5 and Fig. 4.6 indicate that the macroscopic mean stress and the porosity both affect the microscopic effective rate of deformation field.

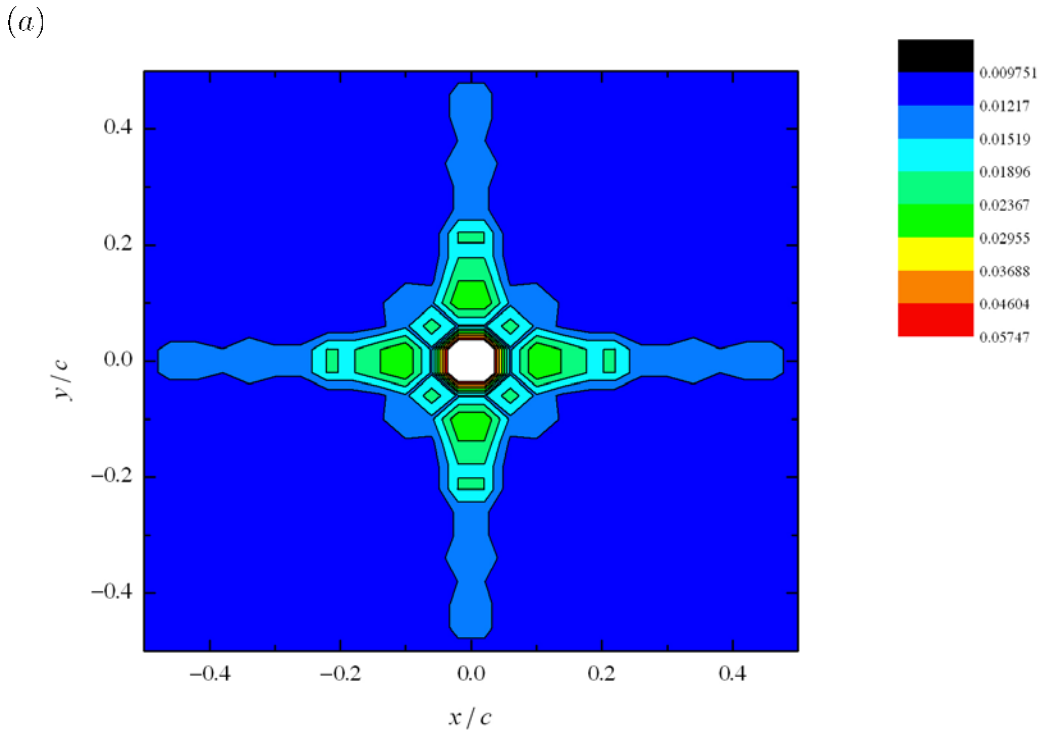

Fig. 4.6. Contour plots of the microscopic effective rate of deformation fields (a) for $f = 0.01$ and (b) for $f = 0.04$.

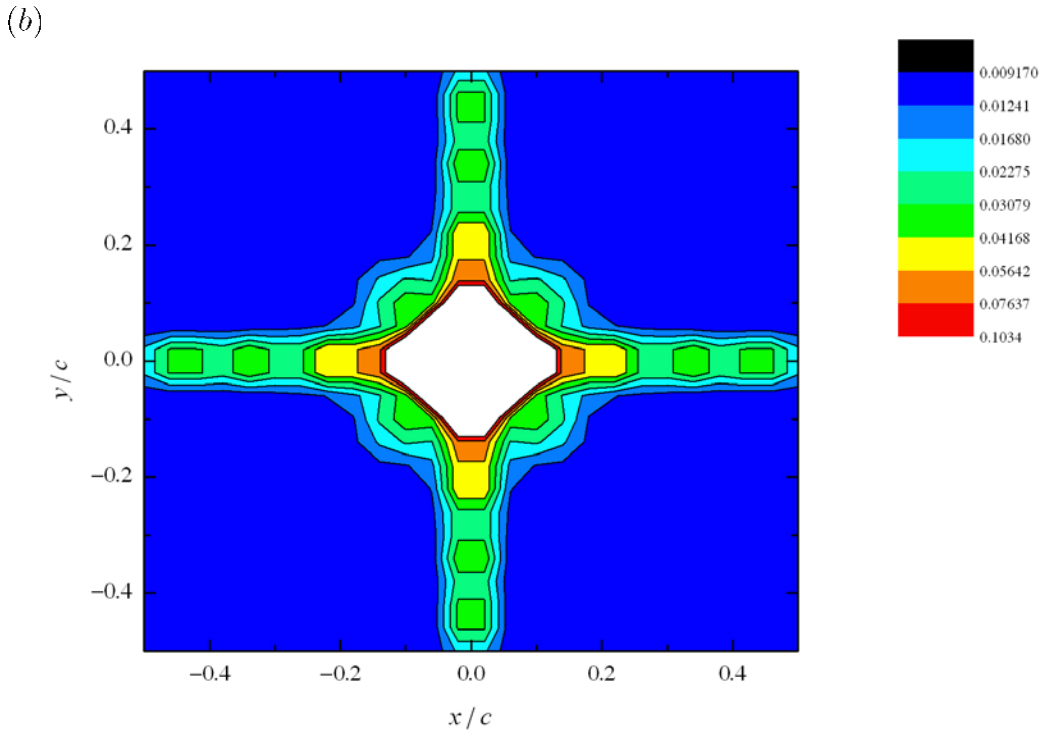

Fig. 4.6 Continued.

4.2.2 Effect of the Unperturbed Velocity Field for Porous Metals with Cylindrical Voids

In this section, let the RVE be subject to in-plane biaxial extension, and let its cross section still be square. Here the other parameters are set to be $f = 0.02$ and $b/c = 1$. As mentioned above, one can treat $\mathbf{v}^A$ as the unperturbed velocity field in the perfect medium and $\mathbf{v}^B$ as the perturbing velocity field due to the void. Also note that $\rho = D^A_{yy}/D^A_{xx}$ by definition. This leads one to name ρ as the unperturbed rate of deformation ratio. Fig. 4.7 plots Σ_e/σ_Y versus Σ_m/σ_Y for different values of ρ. As can

be seen, ρ hardly affects the relationship between Σ_e/σ_Y and Σ_m/σ_Y. This physically means that the unperturbed velocity field does not affect the macroscopic effective stress. This is in agreement with the Gurson model.

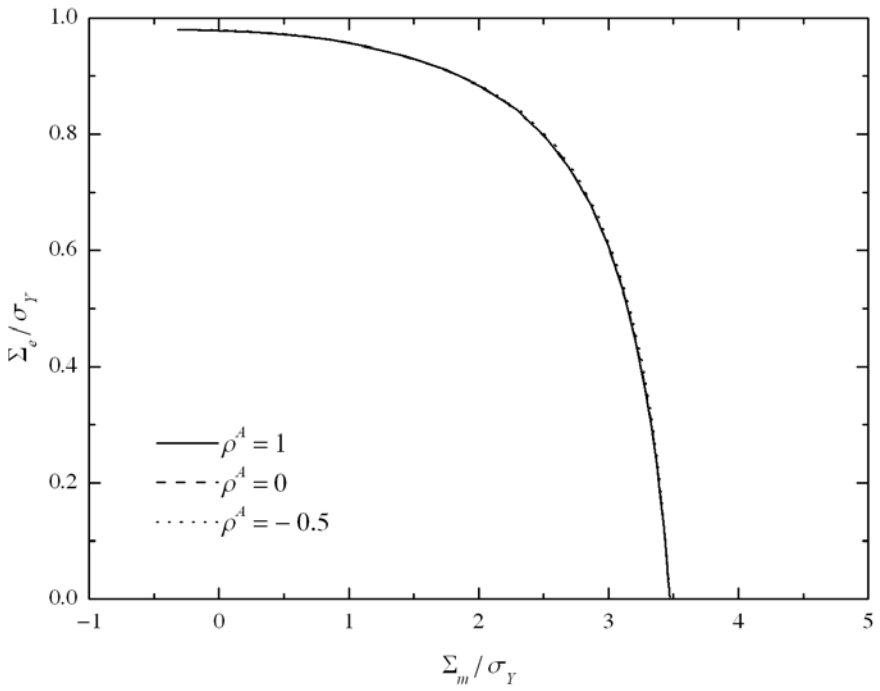

Fig. 4.7. Σ_e/σ_Y versus Σ_m/σ_Y for different values of ρ.

Fig. 4.5 (c) and Fig. 4.8 show the contour plots of the microscopic effective rate of deformation fields for different values of ρ. Here the other parameters are set to be $\Sigma_m/\sigma_Y = 2.3147$, $f = 0.02$, $b/c = 1$, and $D_e = 0.01$. As can be seen, for $\rho = 1$, the field is rotation symmetric of the fourth order (see Fig. 4.5 (c)); as ρ decreases to zero and further below zero, the field becomes merely centrosymmetric, while d_e concentrates more and more at the ligaments perpendicular to the x direction (i.e., the ligaments

subject to higher in-plane extension) but less and less at the ligaments perpendicular to the y direction (i.e., the ligaments subject to lower in-plane extension) (see Fig. 4.8). Fig. 4.5 (c) and Fig. 4.8 indicate that, although the specific unperturbed velocity field does not affect the macroscopic effective stress, it affects the microscopic effective rate of deformation field.

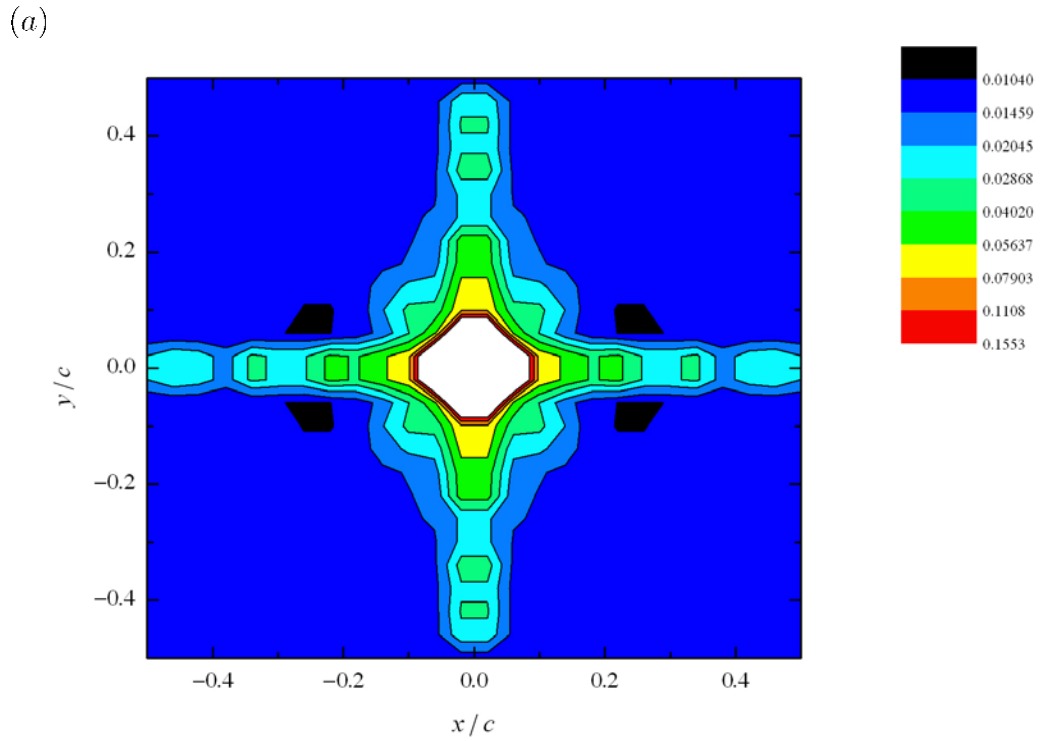

Fig. 4.8. Contour plots of the microscopic effective rate of deformation fields (a) for $\rho = 0$ and (b) for $\rho = -0.5$.

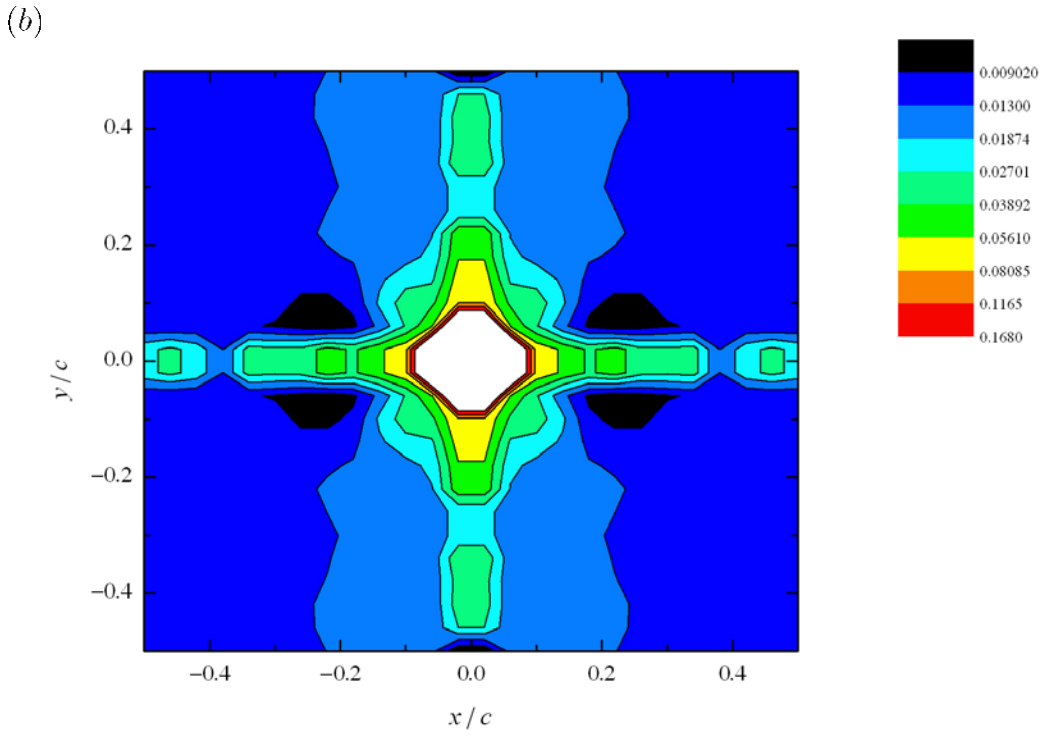

Fig. 4.8 Continued.

4.2.3 Effect of the Void Distribution Anisotropy for Porous Metals with Cylindrical Voids

In this section, let the RVE be subject to in-plane equal biaxial extension, and let its cross section be rectangular so as to incorporate the void distribution anisotropy. Fig. 4.9 plots Σ_e/σ_Y versus Σ_m/σ_Y for different values of b/c. Here the other parameters are set to be $f = 0.02$ and $\rho = 1$.As can be seen, as b/c decreases, the curve slightly shifts outward. This physically means that the void distribution anisotropy slightly affects the macroscopic effective stress.

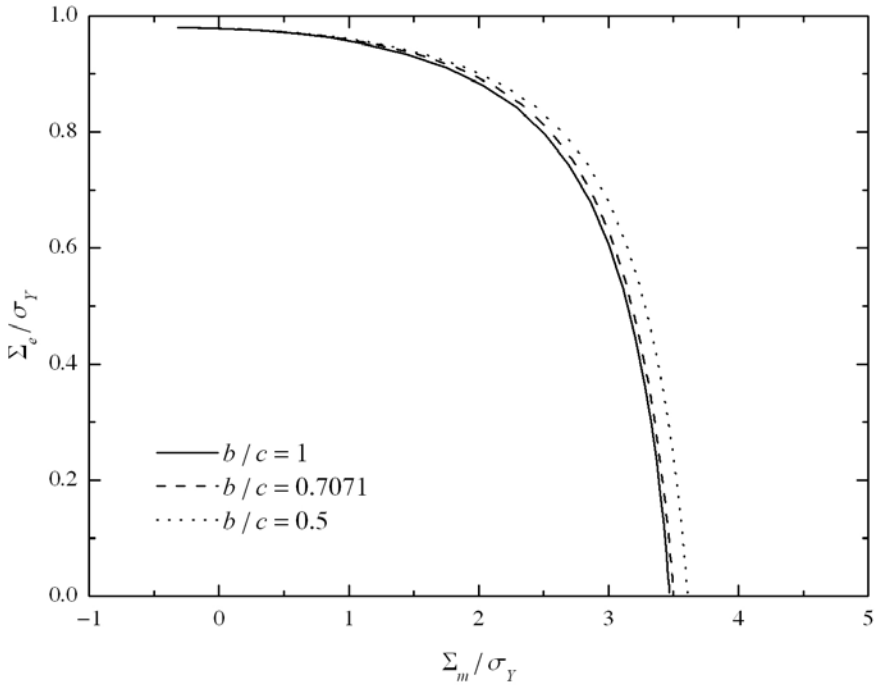

Fig. 4.9. Σ_e/σ_Y versus Σ_m/σ_Y for different values of b/c.

Fig. 4.5 (c) and Fig. 4.10 show the contour plots of the microscopic effective rate of deformation fields for different values of b/c. Here the other parameters are set to be $\Sigma_m/\sigma_Y = 2.3147$, $f = 0.02$, $\rho = 1$, and $D_e = 0.01$. As can be seen, for $b/c = 1$, the field is rotation symmetric of the fourth order (see Fig. 4.5 (c)); as b/c decreases, the field becomes centrosymmetric, while d_e concentrates more and more at the ligaments perpendicular to the x direction (i.e., the wider ligaments) but less and less at the ligaments perpendicular to the y direction (i.e., the narrower ligaments) (see Fig. 4.10). Fig. 4.5 (c) and Fig. 4.10 indicate that the void distribution anisotropy also affects the microscopic effective rate of deformation field.

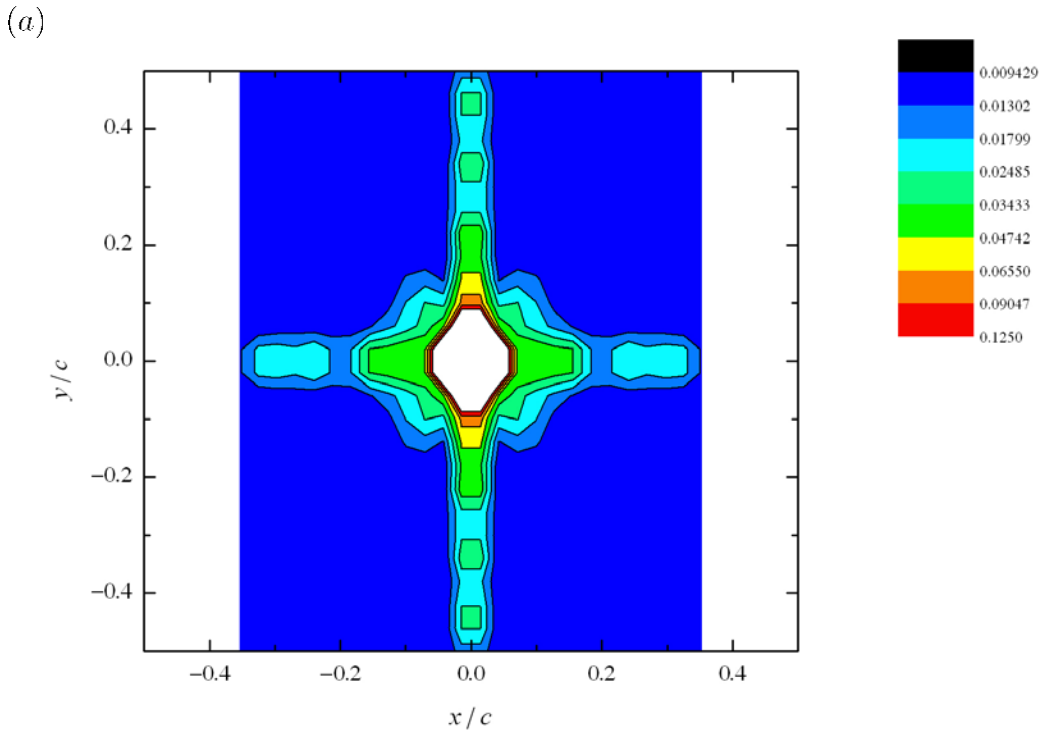

Fig. 4.10. Contour plots of the microscopic effective rate of deformation fields (a) for $b/c = 0.7071$ and (b) for $b/c = 0.5$.

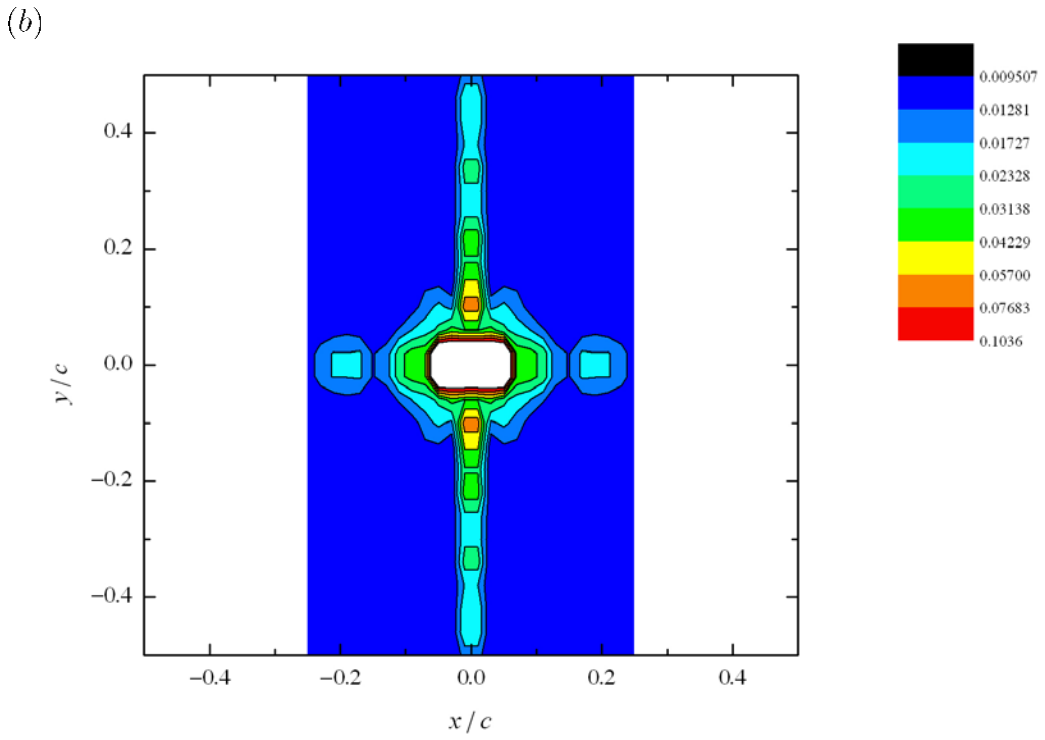

Fig. 4.10 Continued.

4.2.4 Effect of the Void Distribution Anisotropy for Porous Metals with Spherical Voids

The effects of different factors for porous metals with special voids can be evaluated similarly to Sections 4.2.1 – 4.2.3. Here evaluate the effect of the void distribution anisotropy on the microscopic effective rate of deformation field as an example. In this section, let the RVE be subject to equal triaxial extension, and let the RVE width, b, length, c, and height, d, not equal each other so as to incorporate the void distribution anisotropy. Fig. 4.11 shows the contour plots of the microscopic effective

rate of deformation fields for different values of $b/c = b/d$. Here the other parameters are set to be $f = 0.005$ and $D_m = 0.01$. As can be seen, for $b/c = 1$, the field is rotation symmetric of the fourth order (see Fig. 4.11 (a) and (b)); as b/c decreases, the field becomes centrosymmetric, while while d_e concentrates more and more at the ligaments perpendicular to the x direction (i.e., the wider ligaments) but less and less at the ligaments perpendicular to the y direction (i.e., the narrower ligaments) (see Fig. 4.11 (c) and (d)). Fig. 4.11 indicates that, for porous metals with spherical voids, the void distribution anisotropy still affects the microscopic effective rate of deformation field.

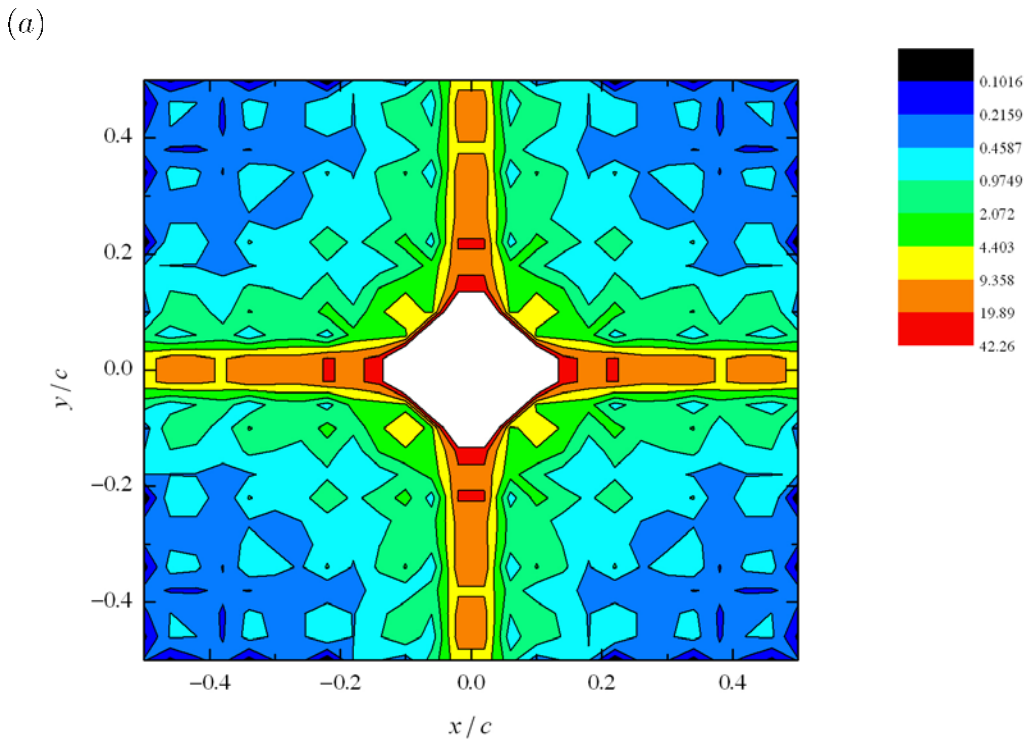

Fig. 4.11. Contour plots of the microscopic effective rate of deformation fields (a) for $b/c = b/d = 1$ at $z = 0$ and (b) at $z = d/2$ and (c) for $b/c = b/d = 0.7071$ at $z = 0$ and (d) at $z = d/2$. The RVE is set to be subject to equal triaxial extension.

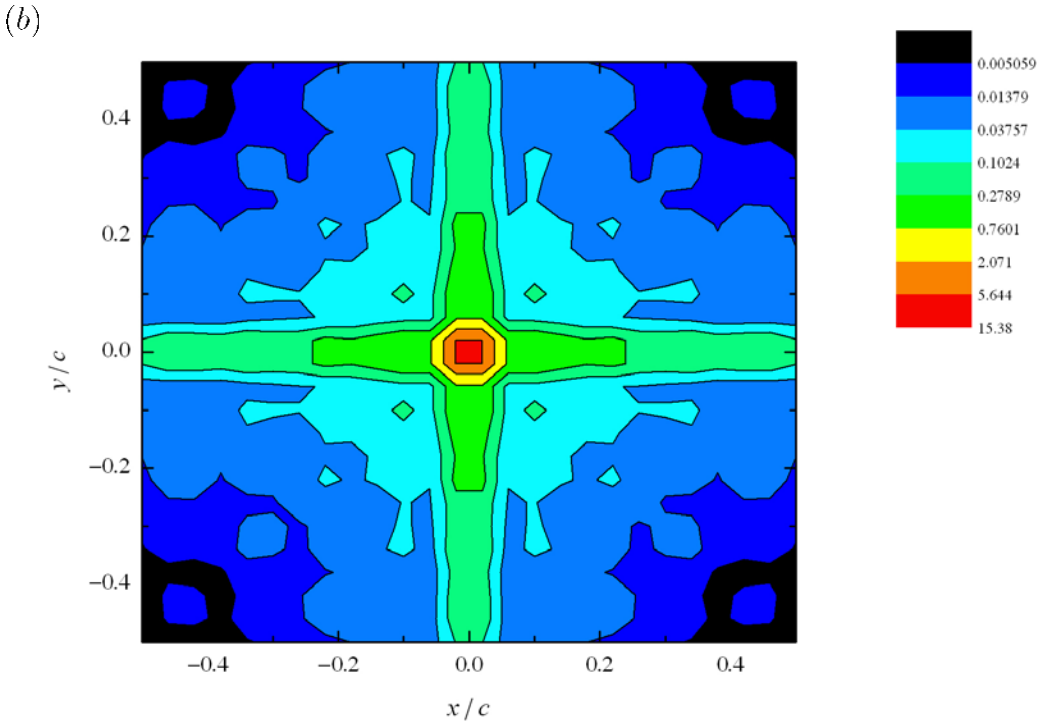

Fig. 4.11 Continued.

(c)

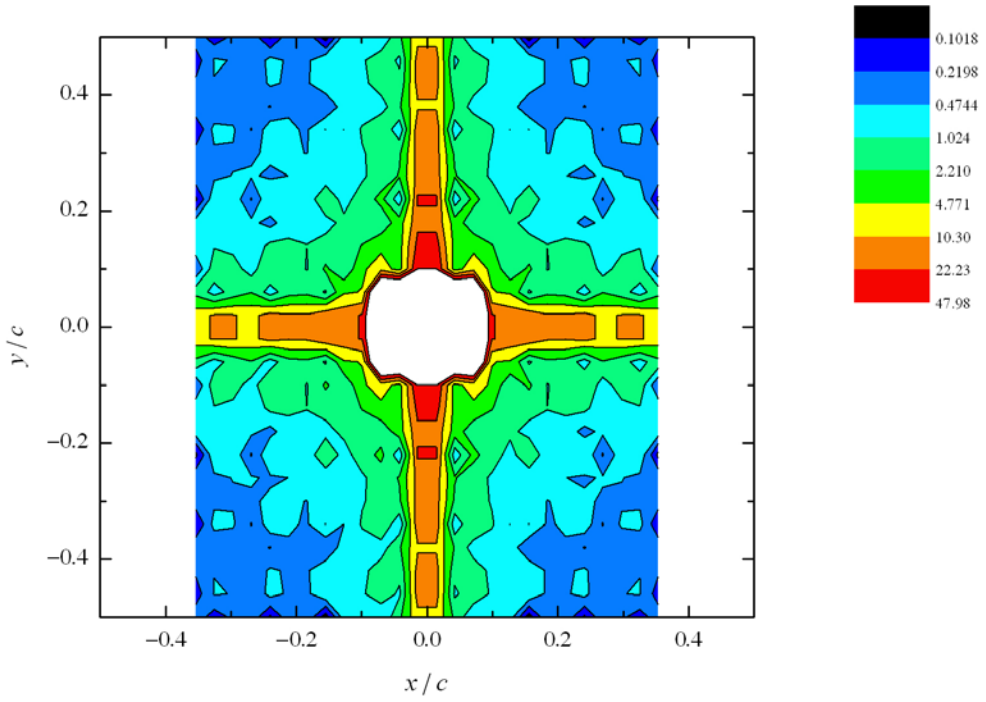

Fig. 4.11 Continued.

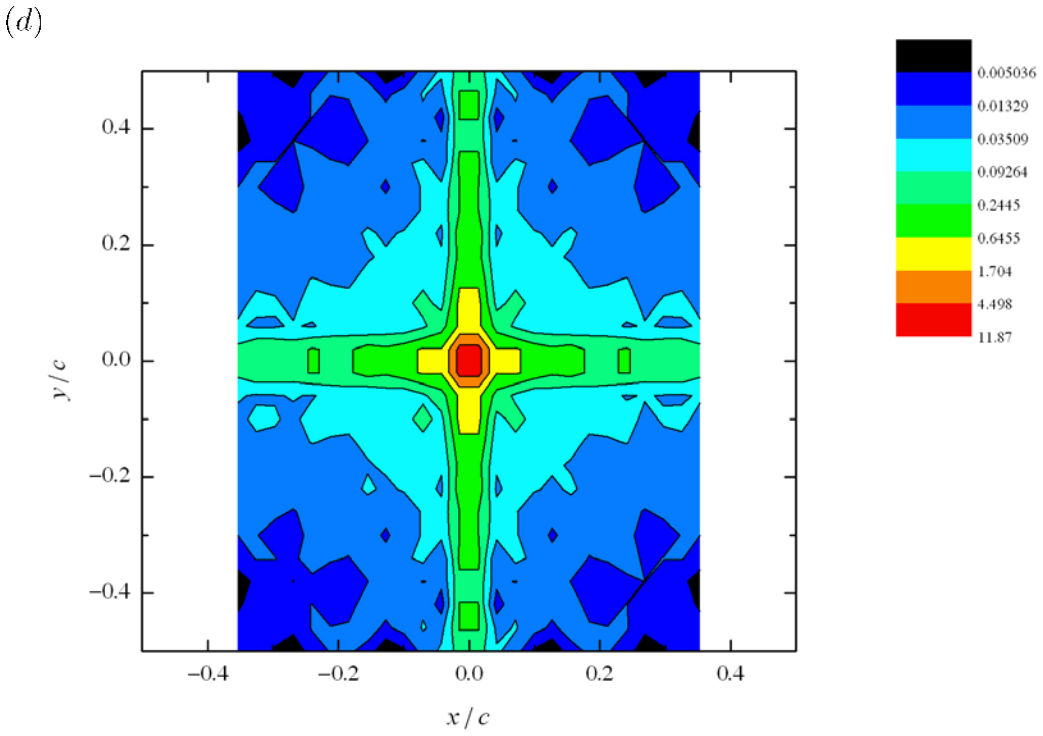

Fig. 4.11 Continued.

4.2.5 Comparison between the Results Obtained Using the Cuboid and Non-Cuboid RVEs

In this part, two types of cuboid RVEs are employed to represent the typical properties of porous metals. Using the cuboid RVEs leads to sophisticated microscopic rate of deformation fields and hereby reliable results. However, it also prohibits one from obtaining analytical macroscopic yield criterions and introduces additional computational efforts. It is hereby of interest whether using the cuboid RVEs is worth the cost. Here compare the results obtained using the cuboid RVE as shown in Fig. 4.2

with those obtained using the corresponding non-cuboid RVE in the Gurson model as an example. For comparison purposes, let the cuboid RVE be subject to in-plane equal biaxial extension, and let its cross section be square ($b = c$). This RVE now turns out to correspond to the cylindrical RVE in the Gurson model. According to Gurson [74], the cylindrical RVEs can be set to be subject to in-plane axisymmetric extension to approximate in-plane equal biaxial extension. Therefore, let the cylindrical RVE be subject to in-plane axisymmetric extension. Fig. 4.12 plots Σ_e/σ_Y versus Σ_m/σ_Y obtained using the cuboid and cylindrical RVEs. Here the other parameters are set to be $f = 0.02$, $\rho = 1$, and $b/c = 1$, and the cylindrical RVE is set to subject axisymmetric extension. As can be seen, for low values of Σ_m/σ_Y, the two curves lie quite close to each other; as Σ_m/σ_Y increases, the curve obtained using the cuboid RVEs deviates more and more from that obtained using the cylindrical RVEs. It is also worth notice that the curve obtained using the cuboid RVEs mostly lies exterior to that obtained using the cylindrical RVEs. Note that the Gurson model provides a rigorous upper bound. Fig. 4.12 hereby indicates that the proposed model also provides a rigorous upper bound.

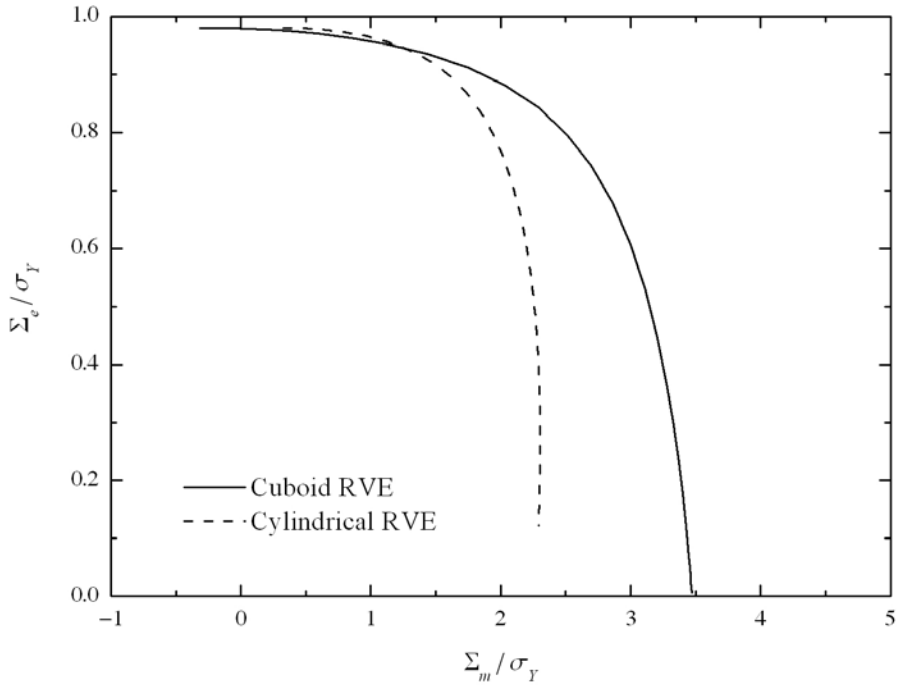

Fig. 4.12. Σ_e/σ_Y versus Σ_m/σ_Y predicted by the proposed model and the Gurson model.

4.3 Conclusions

This part provides an alternative approach to deriving the constitutive relations for porous metals. The hollow cuboid RVEs and the corresponding microscopic velocity fields are found to be able to well represent the typical properties of porous metals with cylindrical and spherical voids. The microscopic velocity fields are found to be able to be obtained using generalized Green's functions. The macroscopic mean stress, the porosity, the unperturbed velocity field, and the void distribution anisotropy are found to affect the macroscopic effective stress and the microscopic effective rate of deformation

field for porous metals with cylindrical voids. The macroscopic effective stress is found to decrease with increasing macroscopic mean stress. As the macroscopic mean stress increases, the microscopic effective rate of deformation is found to concentrate more and more around the void and at the ligaments. The unperturbed velocity field is found to affect not the macroscopic effective stress but the microscopic effective rate of deformation field. The void distribution anisotropy is found to affect the macroscopic effective stress and also the microscopic effective rate of deformation field. The proposed model is found to provide a rigorous upper bound.

The following conclusions can be drawn from the above findings:

- The microscopic velocity fields for other RVE shapes (e.g., elliptic cylindrical and ellipsoidal RVEs) can also be obtained using generalized Green's functions;
- The assumed sophisticated microscopic velocity fields can be employed to estimate the void growth;
- The proposed model is able to predict the upper bounds of the macroscopic yield and mean stresses that a porous metal can endure and hereby to provide an estimation of the safe loads;
- More complicated matrix properties (e.g., plastic anisotropy) and void shapes can be implemented into the proposed model.

5. SUMMARY AND FUTURE WORK

This work consists of three parts corresponding to three different failure mechanisms in metals. The first part provides an alternative approach to modeling the localized necking in anisotropic sheet metals to demonstrate that localized geometric softening at a certain stage of deformation rather than the initial defects is the main cause of localized necking. Its future work includes modeling the localized necking in steel sheet metals by looking into the steel's microstructure and modeling the localized necking in pipes subject to internal pressure. The second part provides an applicable approach to modeling the bifurcation in anisotropic bulk and sheet metals to couple plastic anisotropy and the strain hardening/softening behavior and also to identify different bifurcation modes in sheet metals. Its future work includes implementing more realistic material properties (e.g., the yield criterion, the flow rule, and the strain hardening/softening behavior) into the proposed approach. The applicability of the methodology outlined here in the content of modeling the void coalescence remains to be investigated. One challenge is to be able to describe the velocity field corresponding to the highly localized deformation. The third part provides an alternative approach to deriving the constitutive relations for porous metals to better understand the micromechanism of the ductile fracture in metals. Its future work includes extending the proposed approach to be applicable to porous metals of high porosity and implementing more complicated matrix properties (e.g., plastic anisotropy) and void shapes into the

proposed approach. This work provides novel insights into these three failure mechanisms.

REFERENCES

[1] Keeler SP. Determination of forming limits in automotive stampings. Sheet Metal Industries 1965; 42(461): 683-91.

[2] Goodwin GM. Application of strain analysis to sheet metal forming problems in press shop. Metallurgia Italiana 1968; 60(8): 767-74.

[3] Chung K, Ahn K, Yoo DH, Chung KH, Seo MH, et al. Formability of TWIP (twinning induced plasticity) automotive sheets. International Journal of Plasticity 2011; 27(1): 52-81.

[4] Graf A, Hosford W. The influence of strain-path changes on forming limit diagrams of Al 6111 T4. International Journal of Mechanical Sciences 1994; 36(10): 897-910.

[5] Hecker SS. Simple technique for determining forming limit curves. Sheet Metal Industries 1975; 52: 671-76.

[6] Hecker SS. Experimental studies of sheet stretchability. In: Hecker SS, Ghosh AK, Gegel HL, editors. Formability Analysis, Modeling, and Experimentation. New York: AIME; 1978. p. 150-82.

[7] Keeler SP, Brazier WG. Relationship between laboratory material characterization and press-shop formability. In: Korchynsky M, editor. Microalloying 75. New York: Union Carbide; 1977. p. 517-30.

[8] Lee W, Chung KH, Kim D, Kim J, Kim C, et al., Experimental and numerical study on formability of friction stir welded TWB sheets based on hemispherical dome stretch tests. International Journal of Plasticity 2009; 25(9): 1626-54.

[9] Smith PE, Lee D. Determination of forming limits for aluminum alloys. SAE #982375 1998.

[10] Graf A, Hosford W. Influence of strain-path changes on forming limit diagrams of Al 6111 T4. International Journal of Mechanical Sciences 1994; 36(10): 897-910.

[11] Hill R. On discontinuous plastic states, with special reference to localized necking in thin sheets. Journal of the Mechanics and Physics of Solids 1952; 1(1): 19-30.

[12] Swift HW. Plastic instability under plane stress. Journal of the Mechanics and Physics of Solids 1952; 1(1): 1-18.

[13] Storen S, Rice JR. Localized necking in thin sheets. Journal of the Mechanics and Physics of Solids 1975; 23(6): 421-41.

[14] Chow CL, Jie M, Hu SJ. Forming limit analysis of sheet metals based on a generalized deformation theory. Journal of Engineering Materials and Technology, Transactions of the ASME 2003; 125(3): 260-65.

[15] Chow CL, Jie M, Wu X. Localized necking criterion for strain-softening materials. Journal of Engineering Materials and Technology, Transactions of the ASME 2005; 127(3): 273-8.

[16] Marciniak Z, Kuczynski K. Limit strains in processes of stretch-forming sheet metal. International Journal of Mechanical Sciences 1967; 9(9): 609-20.

[17] Marciniak Z, Kuczynski K, Pokora T. Influence of the plastic properties of a material on the forming limit diagram for sheet metal in tension. International Journal of Mechanical Sciences 1973; 15(10): 789-805.

[18] Hutchinson JW, Neale KW. Sheet necking-II. Time-independent behavior. In: Koistinen DP, Wang NM, editors. Mechanics of Sheet Metal Forming. New York: Plenum Press; 1978. p. 127-53.

[19] Hutchinson JW, Neale KW. Sheet necking-III. Strain-rate effects. In: Koistinen DP, Wang NM, editors. Mechanics of Sheet Metal Forming. New York: Plenum Press; 1978. p. 269-85.

[20] Barata da Rocha A, Barlat F, Jalinier JM. Prediction of the forming limit diagrams of anisotropic sheets in linear and non-linear loading. Materials Science and Engineering 1985; 68(2): 151-64.

[21] Chan KS. Marciniak-Kuczynski approach to calculating forming limit diagrams. In: Wagoner RH, Chan KS, Keeler SP, editors. Forming Limit Diagrams: Concepts, Methods, and Applications. Warrendale: TMS Press; 1989. p. 73-110.

[22] Chan KS, Koss DA, Ghosh AK. Localized necking of sheet at negative minor strains. Metallurgical Transactions. A, Physical Metallurgy and Materials Science 1984; 15(2): 323-9.

[23] Lian J, Baudelet B. Forming limit diagram of sheet metal in the negative minor strain region. Materials Science and Engineering 1987; 86(1-2): 137-44.

[24] Ghosh AK. Plastic flow properties in relation to localized necking in sheets. In: Koistinen DP, Wang NM, editors. Mechanics of Sheet Metal Forming. New York: Plenum Press; 1978. p. 287-311.

[25] Painter MJ, Pearce R. Instability and fracture in sheet metal. Journal of Physics D: Applied Physics 1974; 7: 992-1002.

[26] Tadros AK, Mellor PB. An experimental study of the in-plane stretching of sheet metal. International Journal of Mechanical Sciences 1978; 20(2): 121-34.

[27] Tadros AK, Mellor PB. Some comments on the limit strains in sheet metal stretching. International Journal of Mechanical Sciences 1975; 17: 203-10.

[28] Sowerby R, Duncan JL. Failure in sheet metal in biaxial tension. International Journal of Mechanical Sciences 1971; 13(3): 217-29.

[29] Barlat F, Brem JC, Lege DJ, Chung K, Predeleanu M, et al. Characterization of the formability for aluminum alloy and steel sheets. Studies in Applied Mechanics 1997; 45: 265-72.

[30] Cao J, Yao H, Karafillis A, Boyce MC. Prediction of localized thinning in sheet metal using a general anisotropic yield criterion. International Journal of Plasticity 2000; 16(9): 1105-29.

[31] Graf AF, Hosford WF. Calculations of forming limit diagrams for changing strain paths. Metallurgical Transactions A (Physical Metallurgy and Materials Science) 1993; 24A(11): 2497-501.

[32] Kuroda M, Tvergaard V. Forming limit diagrams for anisotropic metal sheets with different yield criteria. International Journal of Solids and Structures 2000; 37(37): 5037-59.

[33] Lian J, Zhou D, Baudelet B. Application of Hill's new yield theory to sheet metal forming. Part I. Hill's 1979 criterion and its application to predicting sheet forming limit. International Journal of Mechanical Sciences 1989; 31(4): 237-47.

[34] Xu S, Weinmann KJ. Prediction of forming limit curves of sheet metals using Hill's 1993 user-friendly yield criterion of anisotropic materials. International Journal of Mechanical Sciences 1998; 40(9): 913-25.

[35] Yao H, Cao J. Prediction of forming limit curves using an anisotropic yield function with prestrain induced backstress. International Journal of Plasticity 2002; 18(8): 1013-38.

[36] Asaro RJ, Needleman A. Texture development and strain hardening in rate-dependent polycrystals. Acta Metallurgica 1985; 33(6): 923-53.

[37] Barlat F. Crystallographic texture, anisotropic yield surfaces and forming limits of sheet metals. Material Science and Engineering 1987; 91: 55-72.

[38] Ganser HP, Werner EA, Fischer FD. Forming limit diagrams: a micromechanical approach. International Journal of Mechanical Sciences 2000; 42(10): 2041-54.

[39] Tvergaard V, Needleman A. Shear band development in polycrystals. Proceedings of the Royal Society of London, Series A (Mathematical and Physical Sciences) 1993; 443(1919): 547-62.

[40] Wu PD, Lloyd DJ, Jain M, Neale KW, Huang Y. Effects of spatial grain orientation distribution and initial surface topography on sheet metal necking. International Journal of Plasticity 2007; 23(6): 1084-104.

[41] Wu PD, MacEwen SR, Lloyd DJ, Neale KW. A mesoscopic approach for predicting sheet metal formability. Modelling and Simulation in Materials Science and Engineering 2004; 12(3): 511-27.

[42] Wilson DV, Mirshams AR, Roberts WT. An experimental study of the effect of sheet thickness and grain size on limit-strains in biaxial stretching. International Journal of Mechanical Sciences 1983; 25(12): 859-70.

[43] Wilson DV, Roberts WT, Rodrigues PMB. Effect of grain anisotropy on limit strains in biaxial stretching. I. Influence of sheet thickness and grain size in weakly textured sheets. Metallurgical Transactions A (Physical Metallurgy and Materials Science) 1981; 12A(9): 1595-602.

[44] Kuwabara T, Yoshida K, Kuroda M. Path-dependence of the forming limit stresses in a sheet metal. International Journal of Plasticity 2007; 23(3): 361-84.

[45] Korkolis YP, Kyriakides S. Path-dependent failure of inflated aluminum tubes. International Journal of Plasticity 2009; 25(11): 2059-80.

[46] Eyckens P, Bael AV, Houtte PV. An extended Marciniak-Kuczynski model for anisotropic sheet subjected to monotonic strain paths with through-thickness shear. International Journal of Plasticity in press.

[47] Eyckens P, Van Bael A, Van Houtte P. Marciniak-Kuczynski type modelling of the effect of Through-Thickness Shear on the forming limits of sheet metal. International Journal of Plasticity 2009; 25(12): 2249-68.

[48] Allwood JM, Shouler DR. Generalised forming limit diagrams showing increased forming limits with non-planar stress states. International Journal of Plasticity 2009; 25(7): 1207-30.

[49] Wu PD, Embury JD, Lloyd DJ, Huang Y, Neale KW. Effects of superimposed hydrostatic pressure on sheet metal formability. International Journal of Plasticity 2009; 25(9): 1711-25.

[50] Stoughton TB. General forming limit criterion for sheet metal forming. International Journal of Mechanical Sciences 2000; 42(1): 1-17.

[51] Stoughton TB, Yoon JW. Sheet metal formability analysis for anisotropic materials under non-proportional loading. International Journal of Mechanical Sciences 2005; 47(12): 1972-2002.

[52] Stoughton TB, Yoon JW. A new approach for failure criterion for sheet metals. International Journal of Plasticity 2011; 27(3): 440-59.

[53] Aretz H. A simple isotropic-distortional hardening model and its application in elastic-plastic analysis of localized necking in orthotropic sheet metals. International Journal of Plasticity 2008; 24(9): 1457-80.

[54] Kobayashi M. Analysis of deformation localization based on proposed theory of ultrasonic wave velocity propagating in plastically deformed solids. International Journal of Plasticity 2010; 26(1): 107-25.

[55] Borja RI. Bifurcation of elastoplastic solids to shear band mode at finite strain. Computer Methods in Applied Mechanics and Engineering 2002; 191(46): 5287-314.

[56] Hill R. A general theory of uniqueness and stability in elastic-plastic solids. Journal of the Mechanics and Physics of Solids 1958; 6(3): 236-49.

[57] Rice JR, Rudnicki JW. A note on some features of the theory of localization of deformation. International Journal of Solids and Structures 1980; 16(7): 597-605.

[58] Rudnicki JW, Rice JR. Conditions for the localization of deformation in pressure-sensitive dilatant materials. Journal of the Mechanics and Physics of Solids 1975; 23(6): 371-94.

[59] Thomas TY. Plastic flow and fracture of solids. New York: Academic Press; 1961.

[60] Steinmann P, Miehe C, Stein E. On the localization analysis of orthotropic Hill type elastoplastic solids. Journal of the Mechanics and Physics of Solids 1994; 42(12): 1969-94.

[61] Drucker DC, Prager W. Soil mechanics and plastic analysis or limit design. Quarterly of Applied Mathematics 1952. 10(2): 157-65.

[62] Lade PV, Duncan JM. Elastioplastic stress-strain theory for cohesionless soil. American Society of Civil Engineers, Journal of the Geotechnical Engineering Division 1975; 101(10): 1037-53.

[63] Matsuoka H, Nakai T. Stress-deformation and strength characteristics of soil under three different principal stresses. Doboku Gakkai Rombun-

Hokokushu/Proceedings of the Japan Society of Civil Engineers 1974; 232: 59-70.

[64] Ogden RW. Local and global bifurcation phenomena in plane-strain finite elasticity. International Journal of Solids and Structures 1985; 21(2): 121-232.

[65] Armero F, Garikipati K. Analysis of strong discontinuities in multiplicative finite strain plasticity and their relation with the numerical simulation of strain localization in solids. International Journal of Solids and Structures 1996; 33(20-22): 2863-85.

[66] Larsson R, Steinmann P, Runesson K. Finite element embedded localization band for finite strain plasticity based on a regularized strong discontinuity. Mechanics of Cohesive-Frictional Materials 1999; 4(2): 171-94.

[67] Steinmann P, Larsson R, Runesson K. On the localization properties of multiplicative hyperelasto-plastic continua with strong discontinuities. International Journal of Solids and Structures 1997; 34(8): 969-90.

[68] Beremin FM. Cavity formation from inclusions in ductile fracture of A508 steel. Metallurgical Transactions A (Physical Metallurgy and Materials Science) 1981; 12A(5): 723-31.

[69] Budiansky B, Hutchinson JW, Slutsky S. Void growth and collapse in viscous solids. In: Hopkins HG, Sowell MJ, editors. Mechanics of Solids, The Rodney Hill 60th Anniversary Volume. Oxford: Pergamon Press; 1982. p. 13-45.

[70] Lee BJ, Mear ME. Axisymmetric deformation of power-law solids containing a dilute concentration of aligned spheroidal voids. Journal of the Mechanics and Physics of Solids 1992; 40(8): 1805-36.

[71] McClintock FA. A criterion for ductile fracture by the growth of holes. Transactions of the ASME, Journal of Applied Mechanics 1968; 17: 363-71.

[72] Rice JR, Tracey DM. On the ductile enlargement of voids in triaxial stress fields. Journal of the Mechanics and Physics of Solids 1969; 17(3): 201-17.

[73] Tracey DM. Strain-hardening and interaction effects on the growth of voids in ductile fracture. Engineering Fracture Mechanics 1971; 3(3): 301-15.

[74] Gurson AL. Continuum theory of ductile rupture by void nucleation and growth: Part I - Yield criteria and flow rules for porous ductile media. Transactions of the ASME, Series H, Journal of Engineering Materials and Technology 1977; 99(1): 2-15.

[75] Leblond JB. Me´canique de la rupture fragile et ductile. Lavoisier: Hermes Science Publications; 2003.

[76] Hill R. The essential structure of constitutive laws for metal composites and polycrystals. Journal of the Mechanics and Physics of Solids 1967; 15: 79-95.

[77] Mandel J. Contribution the´orique a´ l'e´tude de l'e´crouissage et des lois d'e´coulement plastique. Proceeding of the 11th International Congress on Applied Mechanics. Berlin: Springer; 1964. p. 502-9.

[78] Suquet P. Plasticite´ et homoge´ne´ isation. Universite´ Pierre et Marie Curie–Paris VI; 1982.

[79] Benzerga AA. Micromechanics of coalescence in ductile fracture. Journal of the Mechanics and Physics of Solids 2002; 50(6): 1331-62.

[80] Benzerga AA, Besson J, Pineau A. Anisotropic ductile fracture: Part II: Theory. Acta Materialia 2004; 52(15): 4639-50.

[81] Gologanu M, Leblond JB, Perrin G, Devaux J. Theoretical models for void coalescence in porous ductile solids. I. Coalescence "in layers". International Journal of Solids and Structures 2001; 38(32-33): 5581-94.

[82] Needleman A, Tvergaard V. An analysis of ductile rupture in notched bars. Journal of the Mechanics and Physics of Solids 1984; 32(6): 461-90.

[83] Pardoen T, Hutchinson JW. An extended model for void growth and coalescence. Journal of the Mechanics and Physics of Solids 2000; 48(12): 2467-512.

[84] Tvergaard V. Influence of voids on shear band instabilities under plane strain conditions. International Journal of Fracture 1981; 17(4): 389-407.

[85] Tvergaard V. On localization in ductile materials containing spherical voids. International Journal of Fracture 1982; 18(4): 237-52.

[86] Benzerga AA, Besson J. Plastic potentials for anisotropic porous solids. European Journal of Mechanics, A/Solids, 2001; 20(3): 397-434.

[87] Garajeu M, Michel JC, Suquet P. A micromechanical approach of damage in viscoplastic materials by evolution in size, shape and distribution of voids. Computer Methods in Applied Mechanics and Engineering 2000; 183(3-4): 223-46.

[88] Gologanu M, Leblond JB, Devaux J. Approximate models for ductile metals containing non-spherical voids-case of axisymmetric prolate ellipsoidal cavities. Journal of the Mechanics and Physics of Solids 1993; 41(11): 1723-54.

[89] Gologanu M, Leblond JB, Perrin G, Devaux J. Recent extensions of Gurson's model for porous ductile metals. In: Suquet P, editor.Continuum Micromechanics. New York: Springer; 1997. p. 61-130.

[90] Gologanu M, Leblond JB, Devaux J. Approximate models for ductile metals containing nonspherical voids - case of axisymmetric oblate ellipsoidal cavities. Journal of Engineering Materials and Technology, Transactions of the ASME 1994; 116(3): 290-7.

[91] Keralavarma SM, Benzerga AA. An approximate yield criterion for anisotropic porous media. Comptes Rendus de l'Academie des Sciences Serie II b/Mecanique 2008; 336(9): 685-92.

[92] Keralavarma SM, Benzerga AA. A constitutive model for plastically anisotropic solids with non-spherical voids. Journal of the Mechanics and Physics of Solids 2010; 58(6): 874-901.

[93] Monchiet V, Cazacu O, Charkaluk E, Kondo D. Macroscopic yield criteria for plastic anisotropic materials containing spheroidal voids. International Journal of Plasticity 2008; 24(7): 1158-89.

[94] Hill R. On discontinuous plastic states, with special reference to localized necking in thin sheets. Journal of Mechanics and Physics of Solids 1952; 1(1): 19-30.

[95] Hosford WF, Caddell RM. Metal Forming: Mechanics and Metallurgy. 3rd ed. Cambridge: Cambridge University Press; 2007.

[96] Press WH, Teukolsky SA, Vetterling WT. Numerical recipes in C: the art of scientific computing. 2nd ed. Cambridge: Cambridge University Press;1992.

[97] Graf A, Hosford WF. Calculations of forming limit diagrams. Metallurgical transactions. A, Physical metallurgy and materials science 1990; 21(1): 87-94.

[98] Narasimhan K, Wagoner RH. Finite element modeling simulation of in-plane forming limit diagrams of sheets containing finite defects. Metallurgical transactions. A, Physical metallurgy and materials science 1991; 22(11): 2655-65.

[99] Needleman A, Rice JR. Limits to ductility set by plastic flow localization. In: Koistinen DP, Wang NM, editors. Mechanics of Sheet Metal Forming. New York: Plenum Press; 1978. p. 237-67.

[100] Benzerga AA. Ductile fracture by void growth to coalescence. In: Aref H, Giessen EVD, editors. Advances in applied mechanics. New York: Academic Press; 2010. p. 169-305.

[101] Haberman R. Applied partial differential equations: with Fourier series and boundary value problems. 4 ed. Upper Saddle River: Prentice Hall; 2003.

[102] Gao XL, Rowlands RE. Hybrid method for stress analysis of finite three-dimensional elastic components. International Journal of Solids and Structures 2000; 37(19): 2727-51.

APPENDIX A

C PROGRAM FOR SECTION 2

“nrutil.h”

```
#ifndef _NR_UTILS_H_
#define _NR_UTILS_H_

static float sqrarg;
#define SQR(a) ((sqrarg=(a)) == 0.0 ? 0.0 : sqrarg*sqrarg)

static double dsqrarg;
#define DSQR(a) ((dsqrarg=(a)) == 0.0 ? 0.0 : dsqrarg*dsqrarg)

static double dmaxarg1,dmaxarg2;
#define DMAX(a,b) (dmaxarg1=(a),dmaxarg2=(b),(dmaxarg1) > (dmaxarg2) ?\
      (dmaxarg1) : (dmaxarg2))

static double dminarg1,dminarg2;
#define DMIN(a,b) (dminarg1=(a),dminarg2=(b),(dminarg1) < (dminarg2) ?\
      (dminarg1) : (dminarg2))

static float maxarg1,maxarg2;
#define FMAX(a,b) (maxarg1=(a),maxarg2=(b),(maxarg1) > (maxarg2) ?\
      (maxarg1) : (maxarg2))

static float minarg1,minarg2;
#define FMIN(a,b) (minarg1=(a),minarg2=(b),(minarg1) < (minarg2) ?\
      (minarg1) : (minarg2))

static long lmaxarg1,lmaxarg2;
#define LMAX(a,b) (lmaxarg1=(a),lmaxarg2=(b),(lmaxarg1) > (lmaxarg2) ?\
      (lmaxarg1) : (lmaxarg2))

static long lminarg1,lminarg2;
#define LMIN(a,b) (lminarg1=(a),lminarg2=(b),(lminarg1) < (lminarg2) ?\
      (lminarg1) : (lminarg2))

static int imaxarg1,imaxarg2;
#define IMAX(a,b) (imaxarg1=(a),imaxarg2=(b),(imaxarg1) > (imaxarg2) ?\
      (imaxarg1) : (imaxarg2))
```

```
static int iminarg1,iminarg2;
#define IMIN(a,b) (iminarg1=(a),iminarg2=(b),(iminarg1) < (iminarg2) ?\
        (iminarg1) : (iminarg2))

#define SIGN(a,b) ((b) >= 0.0 ? fabs(a) : -fabs(a))

#if defined(__STDC__) || defined(ANSI) || defined(NRANSI) /* ANSI */

void nrerror(char error_text[]);
float *vector(long nl, long nh);
int *ivector(long nl, long nh);
float **matrix(long nrl, long nrh, long ncl, long nch);
void free_vector(float *v, long nl, long nh);
void free_ivector(int *v, long nl, long nh);
void free_matrix(float **m, long nrl, long nrh, long ncl, long nch);

#else /* ANSI */
/* traditional - K&R */

void nrerror();
float *vector();
float **matrix();
int *ivector();
void free_vector();
void free_ivector();
void free_matrix();

#endif /* ANSI */

#endif /* _NR_UTILS_H_ */
```

“nutril.cpp”

```
#include <stdio.h>
#include <stddef.h>
#include <stdlib.h>
#define NR_END 1
#define FREE_ARG char*

void nrerror(char error_text[])
/* Numerical Recipes standard error handler */
{
        fprintf(stderr,"Numerical Recipes run-time error...\n");
```

```
	fprintf(stderr,"%s\n",error_text);
	fprintf(stderr,"...now exiting to system...\n");
	exit(1);
}

double *vector(long nl, long nh)
/* allocate a double vector with subscript range v[nl..nh] */
{
	double *v;

	v=(double *)malloc((size_t) ((nh-nl+1+NR_END)*sizeof(double)));
	if (!v) nrerror("allocation failure in vector()");
	return v-nl+NR_END;
}

int *ivector(long nl, long nh)
/* allocate an int vector with subscript range v[nl..nh] */
{
	int *v;

	v=(int *)malloc((size_t) ((nh-nl+1+NR_END)*sizeof(int)));
	if (!v) nrerror("allocation failure in ivector()");
	return v-nl+NR_END;
}

double **matrix(long nrl, long nrh, long ncl, long nch)
/* allocate a double matrix with subscript range m[nrl..nrh][ncl..nch] */
{
	long i, nrow=nrh-nrl+1,ncol=nch-ncl+1;
	double **m;

	/* allocate pointers to rows */
	m=(double **) malloc((size_t)((nrow+NR_END)*sizeof(double*)));
	if (!m) nrerror("allocation failure 1 in matrix()");
	m += NR_END;
	m -= nrl;

	/* allocate rows and set pointers to them */
	m[nrl]=(double *) malloc((size_t)((nrow*ncol+NR_END)*sizeof(double)));
	if (!m[nrl]) nrerror("allocation failure 2 in matrix()");
	m[nrl] += NR_END;
	m[nrl] -= ncl;

	for(i=nrl+1;i<=nrh;i++) m[i]=m[i-1]+ncol;
```

```
	/* return pointer to array of pointers to rows */
	return m;
}

void free_vector(double *v, long nl, long nh)
/* free a double vector allocated with vector() */
{
	free((FREE_ARG) (v+nl-NR_END));
}

void free_ivector(int *v, long nl, long nh)
/* free an int vector allocated with ivector() */
{
	free((FREE_ARG) (v+nl-NR_END));
}

void free_matrix(double **m, long nrl, long nrh, long ncl, long nch)
/* free a double matrix allocated by matrix() */
{
	free((FREE_ARG) (m[nrl]+ncl-NR_END));
	free((FREE_ARG) (m+nrl-NR_END));
}
```

“main.cpp”

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include "nrutil.h"
#include "nrutil.cpp"
#define PI 3.14159265
#define DEG 0.01745329

#define ITMAX 100 //Maximum allowed number of iterations.
#define EPS 3.0e-8 //Machine doubleing-point precision.

#define CGOLD 0.3819660
#define ZEPS 1.0e-10
/*Here ITMAX is the maximum allowed number of iterations; CGOLD is the golden
ratio; ZEPS is a small number that protects against trying to achieve fractional accuracy
for a minimum that happens to be exactly zero.*/
#define SHFT(a,b,c,d) (a)=(b);(b)=(c);(c)=(d);
```

```
#define TOL 3.0e-8 //Tolerance passed to brent.

FILE *fp;

double r;
double f_1_00;
double m,n;
double **A,**A_hat;
double R_0,R_90;

int nvar; //Variables that you must define and set in your main program.
int kmax,kount;
double *xp,**yp;

double ea_I,ea_II,ea_III,ea_e;
double dea_I,dea_II,dea_III,dea_e;
double sa_I,sa_II,sa_III,sa_e,sa_n;
double **ea,**dea,**sa;
double eb_I,eb_II,eb_III,eb_e;
double deb_I,deb_II,deb_III,deb_e;
double sb_I,sb_II,sb_III,sb_e,sb_n;
double **eb,**deb,**sb;
double ra,rb,Pa_n,Pb_n;
double psi;
double chi_1,f_1,f;
double ba,bb,mua,mub;
double dea_n,Dde_n;

double increment(double dea_I)
//Given dea_I, solve for deb_I.
{
        dea_II=ra*dea_I;
        dea_e=sqrt(2.0/3.0*(A_hat[1][1]+ra*ra*A_hat[2][2]+pow(1.0+ra,2.0)*A_hat[3][
3]))*dea_I;
        dea[2][2]=dea_I*pow(sin(psi),2.0)+dea_II*pow(cos(psi),2.0);

        sa_I=2.0/9.0*(2.0*A_hat[1][1]-ra*A_hat[2][2]+4.0*(1.0+ra)*A_hat[3][3]);
        sa_II=2.0/9.0*(-A_hat[1][1]+2.0*ra*A_hat[2][2]+4.0*(1.0+ra)*A_hat[3][3]);
        sa_e=sqrt(2.0/3.0*(A_hat[1][1]+ra*ra*A_hat[2][2]+pow(1.0+ra,2.0)*A_hat[3][3
]));
        sa[1][1]=sa_I*pow(cos(psi),2.0)+sa_II*pow(sin(psi),2.0);
        sa[1][2]=-(sa_I-sa_II)*cos(psi)*sin(psi);
        sa_n=sqrt(pow(sa[1][1],2.0)+pow(sa[1][2],2.0));
        mua=sa_n/sa_e;
```

```
	if (dea[2][2]<TOL)
		Pa_n=mua*pow(dea_e,m)*pow(ea_e+dea_e,n);
	else {
		ba=dea_e/dea[2][2];
		Pa_n=mua*pow(ba,m)*pow(ea_e+dea_e,n);
	}

	deb[2][2]=dea[2][2];
	deb_II=(deb[2][2]-deb_I*pow(sin(psi),2.0))/pow(cos(psi),2.0);
	rb=deb_II/deb_I;
	deb_e=sqrt(2.0/3.0*(A_hat[1][1]+rb*rb*A_hat[2][2]+pow(1.0+rb,2.0)*A_hat[3]
[3]))*deb_I;

	sb_I=2.0/9.0*(2.0*A_hat[1][1]-rb*A_hat[2][2]+4.0*(1.0+rb)*A_hat[3][3]);
	sb_II=2.0/9.0*(-A_hat[1][1]+2.0*rb*A_hat[2][2]+4.0*(1.0+rb)*A_hat[3][3]);
	sb_e=sqrt(2.0/3.0*(A_hat[1][1]+rb*rb*A_hat[2][2]+pow(1.0+rb,2.0)*A_hat[3][3
]));
	sb[1][1]=sb_I*pow(cos(psi),2.0)+sb_II*pow(sin(psi),2.0);
	sb[1][2]=-(sb_I-sb_II)*cos(psi)*sin(psi);
	sb_n=sqrt(pow(sb[1][1],2.0)+pow(sb[1][2],2.0));
	mub=sb_n/sb_e;

	if (dea[2][2]<TOL)
		Pb_n=f*mub*pow(deb_e,m)*pow(eb_e+deb_e,n);
	else {
		bb=deb_e/deb[2][2];
		Pb_n=f*mub*pow(bb,m)*pow(eb_e+deb_e,n);
	}

	return Pb_n-Pa_n;
}

double neck(double psi_1)
//Given the strain ratio and the neck orientation, solve for the limiting strain.
{
	double increment(double dea_I);
	double zbrent(double (*func)(double), double x1, double x2, double tol);
	int flag; //i,j;

	ea=matrix(1,3,1,3);
	dea=matrix(1,3,1,3);
	sa=matrix(1,3,1,3);
	eb=matrix(1,3,1,3);
```

```
	deb=matrix(1,3,1,3);
	sb=matrix(1,3,1,3);

	flag=1; //Flag.
	ra=r;
	psi=psi_1;

	ea_I=n/(pow(cos(psi),2.0)+ra*pow(sin(psi),2.0)); //Principal strain at the
formation of the neck.
	ea_e=sqrt(2.0/3.0*(A_hat[1][1]+ra*ra*A_hat[2][2]+pow(1.0+ra,2.0)*A_hat[3][3
]))*ea_I;
	ea[3][3]=-(1.0+ra)*ea_I;

	eb_e=ea_e;
	eb[3][3]=ea[3][3];

	chi_1=(1.0-pow(tan(psi),2.0))*sqrt((A_hat[1][1]+A_hat[3][3])/
		(A_hat[1][1]+pow(tan(psi),4.0)*A_hat[2][2]+pow(1.0-
pow(tan(psi),2.0),2.0)*A_hat[3][3]))*

	exp((pow(sin(psi),2.0)+ra*pow(cos(psi),2.0))/(pow(cos(psi),2.0)+ra*pow(sin(psi
),2.0))*n);
	f_1=1.0-chi_1*(1-f_1_00);
	f=f_1; //Initialize.

	deb_I=0.001; //Major step size.

	do {
		dea_I=zbrent(increment,0.0,f*deb_I,EPS);


		dea[1][1]=dea_I*pow(cos(psi),2.0)+dea_II*pow(sin(psi),2.0);
		dea[1][2]=-(dea_I-dea_II)*cos(psi)*sin(psi);
		deb[1][1]=deb_I*pow(cos(psi),2.0)+deb_II*pow(sin(psi),2.0);
		deb[1][2]=-(deb_I-deb_II)*cos(psi)*sin(psi);
		dea_n=sqrt(pow(dea[1][1],2.0)+pow(dea[1][2],2.0));
		Dde_n=sqrt(pow(deb[1][1]-dea[1][1],2.0)+pow(deb[1][2]-dea[1][2],2.0));

		if (Dde_n/dea_n<9.0) { //Continue loading.
			ea_I=ea_I+dea_I;
			ea_e=ea_e+dea_e;
			ea[3][3]=ea[3][3]-(1+ra)*dea_I;
			eb_e=eb_e+deb_e;
			eb[3][3]=eb[3][3]-(1+rb)*deb_I;
```

```
                        psi=atan(tan(psi)*(1+dea_I)/(1+dea_II));
                        f=f_1*exp(eb[3][3]-ea[3][3]);
                }
                else { //Localization occurs.
                        flag=0;
                        ea_I=ea_I+dea_I;
                        psi=atan(tan(psi)*(1+dea_I)/(1+dea_II));
                }
        } while (flag);

        free_matrix(ea,1,3,1,3);
        free_matrix(dea,1,3,1,3);
        free_matrix(sa,1,3,1,3);
        free_matrix(eb,1,3,1,3);
        free_matrix(deb,1,3,1,3);
        free_matrix(sb,1,3,1,3);

        return ea_I;
}

double zbrent(double (*func)(double), double x1, double x2, double tol)
/*Using Brent's method, find the root of a function func known to lie between x1 and x2.
The root, returned as zbrent, will be refined until its accuracy is tol.*/
{
        int iter;
        double a=x1,b=x2,c=x2,d,e,min1,min2;
        double fa=(*func)(a),fb=(*func)(b),fc,p,q,r,s,tol1,xm;

        if ((fa > 0.0 && fb > 0.0) || (fa < 0.0 && fb < 0.0)) {
                printf("fa:\n");
                printf("%f\n",fa);
                printf("fb:\n");
                printf("%f\n",fb);
                nrerror("Root must be bracketed in zbrent");
        }
        fc=fb;
        for (iter=1;iter<=ITMAX;iter++) {
                if ((fb > 0.0 && fc > 0.0) || (fb < 0.0 && fc < 0.0)) {
                        c=a; //Rename a, b, c and adjust bounding interval
                        fc=fa; //d.
                        e=d=b-a;
                }
                if (fabs(fc) < fabs(fb)) {
                        a=b;
```

```
			b=c;
			c=a;
			fa=fb;
			fb=fc;
			fc=fa;
		}
		tol1=2.0*EPS*fabs(b)+0.5*tol; //Convergence check.
		xm=0.5*(c-b);
		if (fabs(xm) <= tol1 || fb == 0.0) return b;
		if (fabs(e) >= tol1 && fabs(fa) > fabs(fb)) {
			s=fb/fa; //Attempt inverse quadratic interpolation.
			if (a == c) {
				p=2.0*xm*s;
				q=1.0-s;
			} else {
				q=fa/fc;
				r=fb/fc;
				p=s*(2.0*xm*q*(q-r)-(b-a)*(r-1.0));
				q=(q-1.0)*(r-1.0)*(s-1.0);
			}
			if (p > 0.0) q = -q; //Check whether in bounds.
			p=fabs(p);
			min1=3.0*xm*q-fabs(tol1*q);
			min2=fabs(e*q);
			if (2.0*p < (min1 < min2 ? min1 : min2)) {
				e=d; //Accept interpolation.
				d=p/q;
			} else {
				d=xm; //Interpolation failed, use bisection.
				e=d;
			}
		} else { //Bounds decreasing too slowly, use bisection.
			d=xm;
			e=d;
		}
		a=b; //Move last best guess to a.
		fa=fb;
		if (fabs(d) > tol1) //Evaluate new trial root.
			b += d;
		else
			b += SIGN(tol1,xm);
		fb=(*func)(b);
	}
	nrerror("Maximum number of iterations exceeded in zbrent");
```

```
	return 0.0; //Never get here.
}

double brent(double ax, double bx, double cx, double (*f)(double), double tol,
		double *xmin)
/*Given a function f, and given a bracketing triplet of abscissas ax, bx, cx (such that bx
is between ax and cx, and f(bx) is less than both f(ax) and f(cx)), this routine isolates the
minimum to a fractional precision of about tol using Brent's method. The abscissa of the
minimum is returned as xmin, and the minimum function value is returned as brent, the
returned function value.*/
{
	int iter;
	double a,b,d,etemp,fu,fv,fw,fx,p,q,r,tol1,tol2,u,v,w,x,xm;
	double e=0.0; //This will be the distance moved on
	//the step before last.
	a=(ax < cx ? ax : cx); //a and b must be in ascending order,
	b=(ax > cx ? ax : cx); //but input abscissas need not be.
	x=w=v=bx; //Initializations...
	fw=fv=fx=(*f)(x);
	for (iter=1;iter<=ITMAX;iter++) { //Main program loop.
		xm=0.5*(a+b);
		tol2=2.0*(tol1=tol*fabs(x)+ZEPS);
		if (fabs(x-xm) <= (tol2-0.5*(b-a))) { //Test for done here.
			*xmin=x;
			return fx;
		}
		if (fabs(e) > tol1) { //Construct a trial parabolic fit.
			r=(x-w)*(fx-fv);
			q=(x-v)*(fx-fw);
			p=(x-v)*q-(x-w)*r;
			q=2.0*(q-r);
			if (q > 0.0) p = -p;
			q=fabs(q);
			etemp=e;
			e=d;
			if (fabs(p) >= fabs(0.5*q*etemp) || p <= q*(a-x) || p >= q*(b-x))
				d=CGOLD*(e=(x >= xm ? a-x : b-x));
			//The above conditions determine the acceptability of the
parabolic fit. Here we
			//take the golden section step into the larger of the two segments.
			else {
				d=p/q; //Take the parabolic step.
				u=x+d;
				if (u-a < tol2 || b-u < tol2)
```

```
                                d=SIGN(tol1,xm-x);
                        }
                } else {
                        d=CGOLD*(e=(x >= xm ? a-x : b-x));
                }
                u=(fabs(d) >= tol1 ? x+d : x+SIGN(tol1,d));
                fu=(*f)(u);
                //This is the one function evaluation per iteration.
                if (fu <= fx) { //Now decide what to do with our func-
                        if (u >= x) a=x; else b=x; //tion evaluation.
                        SHFT(v,w,x,u) //Housekeeping follows:
                                SHFT(fv,fw,fx,fu)
                } else {
                        if (u < x) a=u; else b=u;
                        if (fu <= fw || w == x) {
                                v=w;
                                w=u;
                                fv=fw;
                                fw=fu;
                        } else if (fu <= fv || v == x || v == w) {
                                v=u;
                                fv=fu;
                        }
                        } //Done with housekeeping. Back for
        } //another iteration.
        nrerror("Too many iterations in brent");
        *xmin=x; //Never get here.
        return fx;
}

int main(void)
// Main program.
{
        double brent(double ax, double bx, double cx,
                double (*f)(double), double tol, double *xmin);
        double neck(double psi_1);
        double increment(double dea_I);
        double zbrent(double (*func)(double), double x1, double x2, double tol);

        double xx,dx,xmin,fx,fb,fa,bx,ax;
        double temp;
        int flag,i,j,kk;

        A=matrix(1,3,1,3);
```

```
        A_hat=matrix(1,3,1,3);

//      f_1_00=0.998;

        m=0.012;
        n=0.325269;

        R_0=1.0;
        R_90=1.0;

        dx=1.0*DEG;

        nvar=4;
        kmax=1000;

        xp=vector(1,kmax);
        yp=matrix(1,nvar,1,kmax);

        for (i=1;i<=3;i++) {
                for (j=1;j<=3;j++) {
                        A[i][j]=0.0;
                        A_hat[i][j]=0.0;
                }
        }
        A[1][1]=2.0/3.0*(-R_0*R_90+2.0*R_0+2.0)/(R_0+1.0);
        A[2][2]=2.0/3.0*(2.0*R_0*R_90+2.0*R_0-1.0)/(R_0+1.0);
        A[3][3]=2.0/3.0*(2.0*R_0*R_90-R_0+2.0)/(R_0+1.0);

        temp=A[1][1]*A[2][2]+A[2][2]*A[3][3]+A[3][3]*A[1][1];
        A_hat[1][1]=(-A[1][1]+2.0*A[2][2]+2.0*A[3][3])/temp;
        A_hat[2][2]=(2.0*A[1][1]-A[2][2]+2.0*A[3][3])/temp;
        A_hat[3][3]=(2.0*A[1][1]+2.0*A[2][2]-A[3][3])/temp;

        r=-0.6; //Jog rightward.
        //Precondition.
        xmin=atan(sqrt(-r+0.01));

        f_1_00=0.989;
        do {
                flag=1;
                xx=atan(sqrt(-r))+(xmin-atan(sqrt(-r+0.001)));
                do {
                        ax=xx-dx;
                        bx=xx+dx;
```

```
                temp=brent(ax,xx,bx,neck,TOL,&xmin);
                if (fabs(xmin-xx)<TOL)
                        flag=0;
                else xx=xmin;
        } while (flag);
        r=r+0.01;
} while (r<-0.5);

r=-0.5; //Left-hand side.
xmin=atan(sqrt(-r+0.01));

do {
        flag=1;
        xx=atan(sqrt(-r))+(xmin-atan(sqrt(-r+0.001)));
        do {
                ax=xx-dx;
                bx=xx+dx;
                temp=brent(ax,xx,bx,neck,TOL,&xmin);
                if (fabs(xmin-xx)<TOL)
                        flag=0;
                else xx=xmin;
        } while (flag);
        xp[++kount]=r;
        yp[2][kount]=temp;
        yp[1][kount]=r*yp[2][kount];
        yp[3][kount]=xmin;
        r=r+0.01;
} while (r<0.0);

r=0.0; //Right-hand side.
xmin=0.0;

do {
        if (xmin<=dx) {
                ax=0.0;
                xx=0.01*DEG;
                fa=neck(ax);
                fx=neck(xx);
                if (fa<=fx) { //xmin=0.0.
                        xx=xmin=ax;
                        xp[++kount]=r;
                        yp[2][kount]=fa;
                        yp[1][kount]=r*fa;
                        yp[3][kount]=xmin;
```

```
                r=r+0.01;
            } else { //xmin!=0.0.
                flag=1;
                xx=xmin;
                do {
                    bx=xx+dx;
                    temp=brent(ax,xx,bx,neck,TOL,&xmin);
                    if (fabs(xmin-xx)<TOL)
                        flag=0;
                    else xx=xmin;
                } while (flag);
                xp[++kount]=r;
                yp[2][kount]=temp;
                yp[1][kount]=r*yp[2][kount];
                yp[3][kount]=xmin;
                r=r+0.01;
            }
        } else {
            flag=1;
            xx=xmin;
            do {
                ax=xx-dx;
                bx=xx+dx;
                temp=brent(ax,xx,bx,neck,TOL,&xmin);
                if (fabs(xmin-xx)<TOL)
                    flag=0;
                else xx=xmin;
            } while (flag);
            xp[++kount]=r;
            yp[2][kount]=temp;
            yp[1][kount]=r*yp[2][kount];
            yp[3][kount]=xmin;
            r=r+0.01;
        }
} while (r<1.0);

kk=kount;

r=0.99; //Jog leftward.
//Precondition.
flag=1;
ax=0.0;
xx=37.0*DEG;
bx=45.0*DEG;
```

```
do {
	temp=brent(ax,xx,bx,neck,TOL,&xmin);
	if (fabs(xmin-xx)<TOL)
		flag=0;
	else {
		ax=0.0;
		xx=xmin;
		bx=45.0*DEG;
	}
} while (flag);
xp[kount]=r;
xmin=(temp < yp[2][kount] ? xmin : yp[3][kount]);
yp[2][kount]=(temp < yp[2][kount] ? temp : yp[2][kount]);
yp[1][kount]=r*yp[2][kount];
yp[3][kount--]=xmin;
r=r-0.01;

do { //Left-hand side.
	if (xmin<=dx) {
		ax=0.0;
		xx=0.01*DEG;
		fa=neck(ax);
		fx=neck(xx);
		if (fa<=fx) { //xmin=0.0.
			xx=xmin=ax;
			xp[kount]=r;
			xmin=(fa < yp[2][kount] ? xmin : yp[3][kount]);
			yp[2][kount]=(fa < yp[2][kount] ? fa : yp[2][kount]);
			yp[1][kount]=r*yp[2][kount];
			yp[3][kount--]=xmin;
			r=r-0.01;
		} else { //xmin!=0.0.
			flag=1;
			xx=xmin;
			do {
				bx=xx+dx;
				temp=brent(ax,xx,bx,neck,TOL,&xmin);
				if (fabs(xmin-xx)<TOL)
					flag=0;
				else xx=xmin;
			} while (flag);
			xp[kount]=r;
			xmin=(temp < yp[2][kount] ? xmin : yp[3][kount]);
			yp[2][kount]=(temp < yp[2][kount] ? temp : yp[2][kount]);
```

```
                yp[1][kount]=r*yp[2][kount];
                yp[3][kount--]=xmin;
                r=r-0.01;
            }
        } else {
            flag=1;
            xx=xmin;
            do {
                ax=xx-dx;
                bx=xx+dx;
                temp=brent(ax,xx,bx,neck,TOL,&xmin);
                if (fabs(xmin-xx)<TOL)
                    flag=0;
                else xx=xmin;
            } while (flag);
            xp[kount]=r;
            xmin=(temp < yp[2][kount] ? xmin : yp[3][kount]);
            yp[2][kount]=(temp < yp[2][kount] ? temp : yp[2][kount]);
            yp[1][kount]=r*yp[2][kount];
            yp[3][kount--]=xmin;
            r=r-0.01;
        }
} while (r>-0.01);

r=-0.01; //Right-hand side.
xmin=0.0;

do {
        flag=1;
        xx=atan(sqrt(-r))+(xmin-atan(sqrt(-r-0.01)));
        do {
            ax=xx-dx;
            bx=xx+dx;
            temp=brent(ax,xx,bx,neck,TOL,&xmin);
            if (fabs(xmin-xx)<TOL)
                flag=0;
            else xx=xmin;
        } while (flag);
        xp[kount]=r;
        xmin=(temp < yp[2][kount] ? xmin : yp[3][kount]);
        yp[2][kount]=(temp < yp[2][kount] ? temp : yp[2][kount]);
        yp[1][kount]=r*yp[2][kount];
        yp[3][kount--]=xmin;
        r=r-0.01;
```

```
        } while (r>-0.51);

        fp=fopen("data.txt","w");
        printf("epsilon_II:\n");
        fprintf(fp,"epsilon_II:\n");
        for (i=1;i<=kk;i++) {
                printf("%f\n",yp[1][i]);
                fprintf(fp,"%f\n",yp[1][i]);
        }
        printf("epsilon_I:\n");
        fprintf(fp,"epsilon_I:\n");
        for (i=1;i<=kk;i++) {
                printf("%f\n",yp[2][i]);
                fprintf(fp,"%f\n",yp[2][i]);
        }
        printf("rho:\n");
        fprintf(fp,"rho:\n");
        for (i=1;i<=kk;i++) {
                printf("%f\n",xp[i]);
                fprintf(fp,"%f\n",xp[i]);
        }
        printf("psi_1:\n");
        fprintf(fp,"psi_1:\n");
        for (i=1;i<=kk;i++) {
                printf("%f\n",yp[3][i]/DEG);
                fprintf(fp,"%f\n",yp[3][i]/DEG);
        }

        return 0;
}
```

APPENDIX B

C PROGRAM FOR SECTION 3

“nrutil.h” //The same as that in Appednix A.

“nutril.cpp” //The same as that in Appednix A.

“main.cpp”

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include "nrutil.h"
#include "nrutil.cpp"
#include <algorithm>
#include <ctime>
using namespace std;
#define NMAX 732
#define PI 3.14159265
#define DEG 0.01745329
#define BIG 1.0e30
#define IT 10
#define AMIN -1.2

#define ITMAX 100 //Maximum allowed number of iterations.
#define EPS 3.0e-8 //Machine doubleing-point precision.

#define CGOLD 0.3819660
#define ZEPS 1.0e-10
/*Here ITMAX is the maximum allowed number of iterations; CGOLD is the golden
ratio; ZEPS is a small number that protects against trying to achieve fractional accuracy
for a minimum that happens to be exactly zero.*/
#define SHFT(a,b,c,d) (a)=(b);(b)=(c);(c)=(d);

#define TOL 3.0e-8 //Tolerance passed to brent.

FILE *fp;

double *et,*st,*Ht;
double *et1,*st1,*Ht1,*et2,*st2,*Ht2;
int k1,k2;
```

```
double yp11,ypn1,*y21,yp12,ypn2,*y22,yp13,ypn3,*y23;
double Hc1,thc1,s1,xx,xt;
double E,nu,G,K,mu,beta;
double alpha,*ep,rho;
double **A,**Ah;
double R_0,R_90;

int nvar; //Variables that you must define and set in your main program.
int kmax,kount;
double *xp,**yp;

double H3(double theta)
{
	double *spp,*sp,*shp,**s,**sh;
	double *stemp,sm,sy,shm,omega2;
	double costh,sinth,temp1,temp2,temp3,temp4,temp5,temp6;
	double H1,H2,H3;
	int i;

	spp=vector(1,3);
	sp=vector(1,3);
	shp=vector(1,3);
	s=matrix(1,2,1,2);
	sh=matrix(1,2,1,2);
	stemp=vector(1,3);

	spp[1]=1.0;
	spp[2]=alpha*spp[1];
	spp[3]=-(1.0+alpha)*spp[1];
	sm=-spp[3];
	for (i=1;i<=3;i++) sp[i]=spp[i]+sm;
	sy=0.0;
	shm=0.0;
	for (i=1;i<=3;i++) {
		stemp[i]=A[i][i]*spp[i];
		sy += spp[i]*stemp[i];
		shm += stemp[i];
	}
	sy=sqrt(3.0/2.0*sy);
	shm /= 3.0;
	omega2=0.0;
	for (i=1;i<=3;i++) {
		shp[i]=stemp[i]-shm;
		omega2 += shp[i]*shp[i];
```

```
        }
        omega2=3.0/2.0*omega2/(sy*sy);

        costh=cos(theta);
        sinth=sin(theta);
        temp1=2.0*shp[1]-shp[2]-shp[3]-2.0/3.0*sy*(beta+mu);
        temp2=s1/sy;
        if ((shp[2] >= shp[3] && temp1 >= 0.0) || (shp[2] <= shp[3] && temp1 <= 0.0))
{
                s[1][1]=temp2*(sp[1]*costh*costh+sp[3]*sinth*sinth);
                s[1][2]=temp2*(-(sp[1]-sp[3])*costh*sinth);
                s[2][1]=s[1][2];
                s[2][2]=temp2*(sp[1]*sinth*sinth+sp[3]*costh*costh);
                sh[1][1]=shp[1]*costh*costh+shp[3]*sinth*sinth;
                sh[1][2]=-(shp[1]-shp[3])*costh*sinth;
                sh[2][1]=sh[1][2];
                sh[2][2]=shp[1]*sinth*sinth+shp[3]*costh*costh;
        }
        else {
                s[1][1]=temp2*(sp[1]*costh*costh+sp[2]*sinth*sinth);
                s[1][2]=temp2*(-(sp[1]-sp[2])*costh*sinth);
                s[2][1]=s[1][2];
                s[2][2]=temp2*(sp[1]*sinth*sinth+sp[2]*costh*costh);
                sh[1][1]=shp[1]*costh*costh+shp[2]*sinth*sinth;
                sh[1][2]=-(shp[1]-shp[2])*costh*sinth;
                sh[2][1]=sh[1][2];
                sh[2][2]=shp[1]*sinth*sinth+shp[2]*costh*costh;
        }
        temp1=K+4.0/3.0*G;
        temp2=G*sh[2][2]/sy+K*mu;
        temp3=1.0+(s[2][2]-s[1][1])/(2.0*G);
        temp4=sh[1][2]/sy;
        temp5=3.0*G*omega2+9.0*K*mu*beta;
        H1=9.0/temp1*(G*sh[2][2]/sy+K*beta)*temp2+9.0*G*temp4*temp4-temp5;
        H2=9.0/temp1*(G*sh[2][2]/sy+K*beta)*temp2+9.0*G/temp3*temp4*temp4-
temp5;
        temp6=3.0*(3.0/2.0*shp[3]/sy+beta);
        H3=H2-temp6*(s[2][2]/temp1*temp2+s[1][2]/temp3*temp4)-
temp6*s[1][2]*s[1][2]/(G*temp1*temp3)*temp2;

        free_vector(spp,1,3);
        free_vector(sp,1,3);
        free_vector(shp,1,3);
        free_matrix(s,1,2,1,2);
```

```
	free_matrix(sh,1,2,1,2);
	free_vector(stemp,1,3);

	return -H3;
}

void strain(double alpha, double *ep)
{
	double *spp,*sp,*shp,**s,**sh;
	double *stemp,sm,sy,shm,ee;
	int i;

	spp=vector(1,3);
	sp=vector(1,3);
	shp=vector(1,3);
	s=matrix(1,2,1,2);
	sh=matrix(1,2,1,2);
	stemp=vector(1,3);

	spp[1]=1.0;
	spp[2]=alpha*spp[1];
	spp[3]=-(1.0+alpha)*spp[1];
	sm=-spp[3];
	for (i=1;i<=3;i++) sp[i]=spp[i]+sm;
	sy=0.0;
	shm=0.0;
	for (i=1;i<=3;i++) {
		stemp[i]=A[i][i]*spp[i];
		sy += spp[i]*stemp[i];
		shm += stemp[i];
	}
	sy=sqrt(3.0/2.0*sy);
	shm /= 3.0;
	ee=0.0;
	for (i=1;i<=3;i++) {
		shp[i]=stemp[i]-shm;
		ep[i]=3.0/2.0*shp[i]/sy;
		ee += ep[i]*Ah[i][i]*ep[i];
		}
	ee=sqrt(2.0/3.0*ee);
	for (i=1;i<=3;i++) ep[i] /= ee;
}

void trial(double alpha, double *Hc1, double *thc1)
```

```
{
        double *spp,*sp,*shp,**s,**sh;
        double *stemp,sm,sy,shm;
        double temp1,temp2;
        int i;

        spp=vector(1,3);
        sp=vector(1,3);
        shp=vector(1,3);
        s=matrix(1,2,1,2);
        sh=matrix(1,2,1,2);
        stemp=vector(1,3);

        spp[1]=1.0;
        spp[2]=alpha*spp[1];
        spp[3]=-(1.0+alpha)*spp[1];
        sm=-spp[3];
        for (i=1;i<=3;i++) sp[i]=spp[i]+sm;
        sy=0.0;
        shm=0.0;
        for (i=1;i<=3;i++) {
                stemp[i]=A[i][i]*spp[i];
                sy += spp[i]*stemp[i];
                shm += stemp[i];
        }
        sy=sqrt(3.0/2.0*sy);
        shm /= 3.0;
        for (i=1;i<=3;i++) shp[i]=stemp[i]-shm;

        temp1=2.0*shp[1]-shp[2]-shp[3]-2.0/3.0*sy*(beta+mu);
        if ((shp[2] >= shp[3] && temp1 >= 0.0) || (shp[2] <= shp[3] && temp1 <= 0.0))
{
                temp1=beta-mu;
                temp2=shp[2]/sy+(beta+mu)/3.0;
                *Hc1=G*(1.0+nu)/(1.0-nu)*temp1*temp1-
9.0/2.0*G*(1.0+nu)*temp2*temp2;
                *thc1=0.5*acos(((1.0-2.0*nu)*shp[2]-
2.0/3.0*sy*(1+nu)*(beta+mu))/(shp[1]-shp[3]));
        }
        else {
                temp1=beta-mu;
                temp2=shp[3]/sy+(beta+mu)/3.0;
                *Hc1=G*(1.0+nu)/(1.0-nu)*temp1*temp1-
9.0/2.0*G*(1.0+nu)*temp2*temp2;
```

```
                *thc1=0.5*acos(((1.0-2.0*nu)*shp[3]-
2.0/3.0*sy*(1+nu)*(beta+mu))/(shp[1]-shp[2]));
        }

        free_vector(spp,1,3);
        free_vector(sp,1,3);
        free_vector(shp,1,3);
        free_matrix(s,1,2,1,2);
        free_matrix(sh,1,2,1,2);
        free_vector(stemp,1,3);
}


int zarr(double xa[], int n)
{
        void nrerror(char error_text[]);
        int klo,khi,k;
        double h;

        klo=1;
        khi=n;
        while (khi-klo > 1) {
                k=(khi+klo) >> 1;
                if (xa[k] < 0.0) khi=k; //Modified due to a decreasing function.
                else klo=k;
        } //klo and khi now bracket the input value of x.
        h=xa[khi]-xa[klo];
        if (h == 0.0) nrerror("Bad xa input to routine zarr"); //The xa's must be distinct.
        return klo;
}

double random(int x)
//Generate a random number between 0 and Pi/4.
{
        return PI/2.0*double(rand()%x)/double(x);
}

double fz1(int k)
{
        double H3(double theta);
        double random(int x);
        void mnbrak(double *ax, double *bx, double *cx, double *fa, double *fb, double
*fc, double (*func)(double));
```

```
	double golden(double ax, double bx, double cx, double (*f)(double), double tol,
double *xmin);

	double ax,bx,cx,fa,fb,fc,xmin;
	double Hc,temp;
	int i;

	s1=st1[k];
	Hc=-BIG;
	for (i=1;i<=IT;i++) {
		ax=random(100);
		bx=random(100);
		mnbrak(&ax,&bx,&cx,&fa,&fb,&fc,H3);
		temp=-golden(ax,bx,cx,H3,TOL,&xmin);
		if (temp > Hc) {
			Hc=temp;
			xx=xmin;
		}
	}
	return Hc-Ht1[k];
}

double fz2(int k)
{
	double H3(double theta);
	double random(int x);
	void mnbrak(double *ax, double *bx, double *cx, double *fa, double *fb, double
*fc, double (*func)(double));
	double golden(double ax, double bx, double cx, double (*f)(double), double tol,
double *xmin);

	double ax,bx,cx,fa,fb,fc,xmin;
	double Hc,temp;
	int i;

	s1=st2[k];
	Hc=-BIG;
	for (i=1;i<=IT;i++) {
		ax=random(100);
		bx=random(100);
		mnbrak(&ax,&bx,&cx,&fa,&fb,&fc,H3);
		temp=-golden(ax,bx,cx,H3,TOL,&xmin);
		if (temp > Hc) {
			Hc=temp;
```

```
                        xx=xmin;
                }
        }
        return Hc-Ht2[k];
}

double fr1(double H)
{
        double H3(double theta);
        double random(int x);
        void splint1(double xa[], double ya[], double y2a[], int n, double x, double *y);
        void mnbrak(double *ax, double *bx, double *cx, double *fa, double *fb, double
*fc, double (*func)(double));
        double golden(double ax, double bx, double cx, double (*f)(double), double tol,
double *xmin);

        double ax,bx,cx,fa,fb,fc,xmin;
        double Hc,temp;
        int i;

        splint1(Ht1,st1,y21,k1,H,&s1);
        Hc=-BIG;
        for (i=1;i<=IT;i++) {
                ax=random(100);
                bx=random(100);
                mnbrak(&ax,&bx,&cx,&fa,&fb,&fc,H3);
                temp=-golden(ax,bx,cx,H3,TOL,&xmin);
                if (temp > Hc) {
                        Hc=temp;
                        xx=xmin;
                }
        }
        return Hc-H;
}

double fr2(double H)
{
        double H3(double theta);
        double random(int x);
        void splint2(double xa[], double ya[], double y2a[], int n, double x, double *y);
        void mnbrak(double *ax, double *bx, double *cx, double *fa, double *fb, double
*fc, double (*func)(double));
        double golden(double ax, double bx, double cx, double (*f)(double), double tol,
double *xmin);
```

```
	double ax,bx,cx,fa,fb,fc,xmin;
	double Hc,temp;
	int i;

	splint2(Ht2,st2,y22,k2,H,&s1);
	Hc=-BIG;
	for (i=1;i<=IT;i++) {
		ax=random(100);
		bx=random(100);
		mnbrak(&ax,&bx,&cx,&fa,&fb,&fc,H3);
		temp=-golden(ax,bx,cx,H3,TOL,&xmin);
		if (temp > Hc) {
			Hc=temp;
			xx=xmin;
		}
	}
	return Hc-H;
}

void spline(double x[], double y[], int n, double yp1, double ypn, double y2[])
/*Given arrays x[1..n] and y[1..n] containing a tabulated function, i.e., yi = f(xi), with x1
<x2 < :: : < xN, and given values yp1 and ypn for the first derivative of the interpolating
function at points 1 and n, respectively, this routine returns an array y2[1..n] that
contains the second derivatives of the interpolating function at the tabulated points xi. If
yp1 and/or ypn are equal to 1 . 1030 or larger, the routine is signaled to set the
corresponding boundary condition for a natural spline, with zero second derivative on
that boundary.*/
{
	int i,k;
	double p,qn,sig,un,*u;

	u=vector(1,n-1);
	if (yp1 > 0.99e30) //The lower boundary condition is set either to be "nat
		y2[1]=u[1]=0.0; //ural"
	else { //or else to have a specified first derivative.
		y2[1] = -0.5;
		u[1]=(3.0/(x[2]-x[1]))*((y[2]-y[1])/(x[2]-x[1])-yp1);
	}
	for (i=2;i<=n-1;i++) { //This is the decomposition loop of the tridiagonal
algorithm.
		//y2 and u are used for temporary storage of the decomposed factors.
		sig=(x[i]-x[i-1])/(x[i+1]-x[i-1]);
		p=sig*y2[i-1]+2.0;
```

```
                y2[i]=(sig-1.0)/p;
                u[i]=(y[i+1]-y[i])/(x[i+1]-x[i]) - (y[i]-y[i-1])/(x[i]-x[i-1]);
                u[i]=(6.0*u[i]/(x[i+1]-x[i-1])-sig*u[i-1])/p;
        }
        if (ypn > 0.99e30) //The upper boundary condition is set either to be
                qn=un=0.0; //"natural"
        else { //or else to have a specified first derivative.
                qn=0.5;
                un=(3.0/(x[n]-x[n-1]))*(ypn-(y[n]-y[n-1])/(x[n]-x[n-1]));
        }
        y2[n]=(un-qn*u[n-1])/(qn*y2[n-1]+1.0);
        for (k=n-1;k>=1;k--) //This is the backsubstitution loop of the tridiagonal
                y2[k]=y2[k]*y2[k+1]+u[k]; //algorithm.
                free_vector(u,1,n-1);
}

void splint1(double xa[], double ya[], double y2a[], int n, double x, double *y)
/*Given the arrays xa[1..n] and ya[1..n], which tabulate a function (with the xai's in
order), and given the array y2a[1..n], which is the output from spline above, and given a
value of x, this routine returns a cubic-spline interpolated value y.*/
{
        void nrerror(char error_text[]);
        int klo,khi,k;
        double h,b,a;

        klo=1; /*We will find the right place in the table by means of
                  bisection. This is optimal if sequential calls to this
                  routine are at random values of x. If sequential calls
                  are in order, and closely spaced, one would do better
                  to store previous values of klo and khi and test if
        they remain appropriate on the next call.*/
        khi=n;
        while (khi-klo > 1) {
                k=(khi+klo) >> 1;
                if (xa[k] < x) khi=k; //Modified due to the trends of Ht1[] and st1[].
                else klo=k;
        } //klo and khi now bracket the input value of x.
        h=xa[khi]-xa[klo];
        if (h == 0.0) nrerror("Bad xa input to routine splint1"); //The xa's must be dis-
        a=(xa[khi]-x)/h; //tinct.
        b=(x-xa[klo])/h; //Cubic spline polynomial is now evaluated.
        *y=a*ya[klo]+b*ya[khi]+((a*a*a-a)*y2a[klo]+(b*b*b-b)*y2a[khi])*(h*h)/6.0;
}
```

```
void splint2(double xa[], double ya[], double y2a[], int n, double x, double *y)
/*Given the arrays xa[1..n] and ya[1..n], which tabulate a function (with the xai's in
order), and given the array y2a[1..n], which is the output from spline above, and given a
value of x, this routine returns a cubic-spline interpolated value y.*/
{
	void nrerror(char error_text[]);
	int klo,khi,k;
	double h,b,a;

	klo=1; /*We will find the right place in the table by means of
		  bisection. This is optimal if sequential calls to this
		  routine are at random values of x. If sequential calls
		  are in order, and closely spaced, one would do better
		  to store previous values of klo and khi and test if
	they remain appropriate on the next call.*/
	khi=n;
	while (khi-klo > 1) {
		k=(khi+klo) >> 1;
		if (xa[k] > x) khi=k; //Modified due to the trends of Ht2[] and st2[].
		else klo=k;
	} //klo and khi now bracket the input value of x.
	h=xa[khi]-xa[klo];
	if (h == 0.0) nrerror("Bad xa input to routine splint2"); //The xa's must be dis-
	a=(xa[khi]-x)/h; //tinct.
	b=(x-xa[klo])/h; //Cubic spline polynomial is now evaluated.
	*y=a*ya[klo]+b*ya[khi]+((a*a*a-a)*y2a[klo]+(b*b*b-b)*y2a[khi])*(h*h)/6.0;
}

void zbrak(double (*func)(int), int n, int *xb1, int *xb2)
/*Given a function fx defined on the interval from x1-x2 subdivide the interval into n
equally spaced segments, and search for zero crossings of the function. nb is input as the
maximum number of roots sought, and is reset to the number of bracketing pairs
xb1[1..nb], xb2[1..nb] that are found.*/
{
	int klo,khi,k;
	double x,fp,fc,fmid,dx;

	klo=1;
	khi=n;
	fp=(*func)(klo);
	fc=(*func)(khi);
	if (fp*fc >= 0.0) {
		if (fp >= 0.0) { //Applicable to H3-Ht[i];
			*xb1=1;
```

```
                *xb2=2;
                return;
        }
        else {
                *xb1=n-1;
                *xb2=n;
                return;
        }
    }
    while (khi-klo > 1) {
        k=(khi+klo) >> 1;
        fmid=(*func)(k);
        if (fp*fmid < 0.0) khi=k;
        else klo=k;
        fp=(*func)(klo);
        fc=(*func)(khi);
    } //klo and khi now bracket the input value of x.
    *xb1=klo;
    *xb2=khi;
}

#define JMAX 40 //Maximum allowed number of bisections.

double rtbis1(double (*func)(double), int x1, int x2, double xacc)
/*Using bisection, find the root of a function func known to lie between x1 and x2. The
root, returned as rtbis, will be refined until its accuracy is .xacc.*/
{
    void nrerror(char error_text[]);
    int j;
    double dx,f,fmid,xmid,rtb;

    f=(*func)(Ht1[x1]);
    fmid=(*func)(Ht1[x2]);
    if (f*fmid >= 0.0) return Ht1[x1]; //nrerror("Root must be bracketed for bisection
in rtbis");
    rtb = f < 0.0 ? (dx=Ht1[x2]-Ht1[x1],Ht1[x1]) : (dx=Ht1[x1]-Ht1[x2],Ht1[x2]);
//Orient the search so that f>0
    for (j=1;j<=JMAX;j++) { //lies at x+dx.
        fmid=(*func)(xmid=rtb+(dx *= 0.5)); //Bisection loop.
        if (fmid <= 0.0) rtb=xmid;
        if (fabs(dx) < xacc || fmid == 0.0) return rtb;
    }
    nrerror("Too many bisections in rtbis");
    return 0.0; //Never get here.
```

```
}

double rtbis2(double (*func)(double), int x1, int x2, double xacc)
/*Using bisection, find the root of a function func known to lie between x1 and x2. The
root, returned as rtbis, will be refined until its accuracy is .xacc.*/
{
	void nrerror(char error_text[]);
	int j;
	double dx,f,fmid,xmid,rtb;

	f=(*func)(Ht2[x1]);
	fmid=(*func)(Ht2[x2]);
	if (f*fmid >= 0.0) return Ht2[x1]; //nrerror("Root must be bracketed for bisection
in rtbis");
	rtb = f < 0.0 ? (dx=Ht2[x2]-Ht2[x1],Ht2[x1]) : (dx=Ht2[x1]-Ht2[x2],Ht2[x2]);
//Orient the search so that f>0
	for (j=1;j<=JMAX;j++) { //lies at x+dx.
		fmid=(*func)(xmid=rtb+(dx *= 0.5)); //Bisection loop.
		if (fmid <= 0.0) rtb=xmid;
		if (fabs(dx) < xacc || fmid == 0.0) return rtb;
	}
	nrerror("Too many bisections in rtbis");
	return 0.0; //Never get here.
}

#include <math.h>
#include "nrutil.h"
#define GOLD 1.618034
#define GLIMIT 100.0
#define TINY 1.0e-20
#define SHFT(a,b,c,d) (a)=(b);(b)=(c);(c)=(d);
/*Here GOLD is the default ratio by which successive intervals are magnified; GLIMIT
is the maximum magnification allowed for a parabolic-fit step.*/

#define ITMAX 100
#define CGOLD 0.3819660
#define ZEPS 1.0e-10
/*Here ITMAX is the maximum allowed number of iterations; CGOLD is the golden
ratio; ZEPS is a small number that protects against trying to achieve fractional accuracy
for a minimum that happens to be exactly zero.*/

#define R 0.61803399 //The golden ratios.
#define C (1.0-R)
#define SHFT2(a,b,c) (a)=(b);(b)=(c);
```

```
#define SHFT3(a,b,c,d) (a)=(b);(b)=(c);(c)=(d);

void mnbrak(double *ax, double *bx, double *cx, double *fa, double *fb, double *fc,
                    double (*func)(double))
/*Given a function func, and given distinct initial points ax and bx, this routine searches
in the downhill direction (defined by the function as evaluated at the initial points) and
returns new points ax, bx, cx that bracket a minimum of the function. Also returned are
the function values at the three points, fa, fb, and fc.*/
{
        double ulim,u,r,q,fu,dum;

        *fa=(*func)(*ax);
        *fb=(*func)(*bx);
        if (*fb > *fa) { //Switch roles of a and b so that we can go
                SHFT(dum,*ax,*bx,dum) //downhill in the direction from a to b.
                SHFT(dum,*fb,*fa,dum)
        }
        *cx=(*bx)+GOLD*(*bx-*ax); //First guess for c.
        *fc=(*func)(*cx);
        while (*fb > *fc) { //Keep returning here until we bracket.
                r=(*bx-*ax)*(*fb-*fc); //Compute u by parabolic extrapolation from
                q=(*bx-*cx)*(*fb-*fa); //a, b, c. TINY is used to prevent any possible
division by zero.
                u=(*bx)-((*bx-*cx)*q-(*bx-*ax)*r)/(2.0*SIGN(FMAX(fabs(q-
r),TINY),q-r));
                ulim=(*bx)+GLIMIT*(*cx-*bx);
                //We won't go farther than this. Test various possibilities:
                if ((*bx-u)*(u-*cx) > 0.0) { //Parabolic u is between b and c: try it.
                        fu=(*func)(u);
                        if (fu < *fc) { //Got a minimum between b and c.
                                *ax=(*bx);
                                *bx=u;
                                *fa=(*fb);
                                *fb=fu;
                                return;
                        } else if (fu > *fb) { //Got a minimum between between a and u.
                                *cx=u;
                                *fc=fu;
                                return;
                        }
                        u=(*cx)+GOLD*(*cx-*bx); //Parabolic fit was no use. Use default
mag-
                        fu=(*func)(u); //nification.
                } else if ((*cx-u)*(u-ulim) > 0.0) { //Parabolic fit is between c and its
```

```
                        fu=(*func)(u); //allowed limit.
                        if (fu < *fc) {
                                SHFT(*bx,*cx,u,*cx+GOLD*(*cx-*bx))
                                        SHFT(*fb,*fc,fu,(*func)(u))
                        }
                } else if ((u-ulim)*(ulim-*cx) >= 0.0) { //Limit parabolic u to maximum
                        u=ulim; //allowed value.
                        fu=(*func)(u);
                } else { //Reject parabolic u, use default magnifica-
                        u=(*cx)+GOLD*(*cx-*bx); //tion.
                        fu=(*func)(u);
                }
                SHFT(*ax,*bx,*cx,u) //Eliminate oldest point and continue.
                        SHFT(*fa,*fb,*fc,fu)
        }
}

double golden(double ax, double bx, double cx, double (*f)(double), double tol,
                         double *xmin)
/*Given a function f, and given a bracketing triplet of abscissas ax, bx, cx (such that bx
is between ax and cx, and f(bx) is less than both f(ax) and f(cx)), this routine performs a
golden section search for the minimum, isolating it to a fractional precision of about tol.
The abscissa of the minimum is returned as xmin, and the minimum function value is
returned as golden, the returned function value.*/
{
        double f1,f2,x0,x1,x2,x3;
        x0=ax; //At any given time we will keep track of four
        x3=cx; //points, x0,x1,x2,x3.
        if (fabs(cx-bx) > fabs(bx-ax)) { //Make x0 to x1 the smaller segment,
                x1=bx;
                x2=bx+C*(cx-bx); //and fill in the new point to be tried.
        } else {
                x2=bx;
                x1=bx-C*(bx-ax);
        }
        f1=(*f)(x1); //The initial function evaluations. Note that
        f2=(*f)(x2); //we never need to evaluate the function
        while (fabs(x3-x0) > tol*(fabs(x1)+fabs(x2))) { //at the original endpoints.
                if (f2 < f1) { //One possible outcome,
                        SHFT3(x0,x1,x2,R*x1+C*x3) //its housekeeping,
                                SHFT2(f1,f2,(*f)(x2)) //and a new function evaluation.
                } else { //The other outcome,
                        SHFT3(x3,x2,x1,R*x2+C*x0)
                                SHFT2(f2,f1,(*f)(x1)) //and its new function evaluation.
```

```
                        }
        } //Back to see if we are done.
        if (f1 < f2) { //We are done. Output the best of the two
                *xmin=x1; //current values.
                return f1;
        } else {
                *xmin=x2;
                return f2;
        }
}

int main(void)
// Main program.
{
        double zbrent(double (*func)(double), double x1, double x2, double tol);
        double brent(double ax, double bx, double cx,
                double (*f)(double), double tol, double *xmin);

        double temp;
        int xb1,xb2,i,j;

        double et0[NMAX]={0.00530,      0.00539,      0.00544,      0.00543,
        …
        0.29940,      0.29984,      0.30030,      0.30071,      0.30121};
        double st0[NMAX]={224.80029,    225.92985,    227.02087,    228.10968,
        …
        177.91770,    176.03557,    173.69792,    172.60690,    171.39155};
        double Ht0[NMAX]={13188.99168, 12555.19453,  11921.07304,  11401.57595,
…
-3146.85009,  -3189.18008,  -3225.36108,  -3267.17647,  -3300.81183};
        et=vector(1,NMAX);
        st=vector(1,NMAX);
        Ht=vector(1,NMAX);
        for (i=1;i<=NMAX;i++) {
                et[i]=et0[i-1];
                st[i]=st0[i-1];
                Ht[i]=Ht0[i-1];
        }
        k1=zarr(Ht,NMAX);
        k2=NMAX-k1;
        et1=vector(1,k1);
        st1=vector(1,k1);
        Ht1=vector(1,k1);
        et2=vector(1,k2);
```

```
st2=vector(1,k2);
Ht2=vector(1,k2);
for (i=1;i<=k1;i++) {
        et1[i]=et[i];
        st1[i]=st[i];
        Ht1[i]=Ht[i];
}
for (i=1;i<=k2;i++) {
        et2[i]=et[k1+i];
        st2[i]=st[k1+i];
        Ht2[i]=Ht[k1+i];
}
y21=vector(1,k1);
y22=vector(1,k2);
yp11=(st[2]-st[1])/(Ht[2]-Ht[1]);
ypn1=(st[k1+1]-st[k1-1])/(Ht[k1+1]-Ht[k1-1]);
yp12=(st[k1+2]-st[k1])/(Ht[k1+2]-Ht[k1]);
ypn2=(st[NMAX]-st[NMAX-1])/(Ht[NMAX]-Ht[NMAX-1]);
spline(Ht1,st1,k1,yp11,ypn1,y21);
spline(Ht2,st2,k2,yp12,ypn2,y22);
y23=vector(1,NMAX);
yp13=(et[2]-et[1])/(Ht[2]-Ht[1]);
ypn3=(et[NMAX]-et[NMAX-1])/(Ht[NMAX]-Ht[NMAX-1]);
spline(Ht,et,NMAX,yp13,ypn3,y23);

E=55000.0;
nu=0.3;
G=E/(2.0*(1.0+nu));
K=E/(3.0*(1.0-2.0*nu));
mu=0.02;
beta=0.0;
A=matrix(1,3,1,3);
Ah=matrix(1,3,1,3);
ep=vector(1,3);
R_0=0.5;
R_90=R_0;
nvar=8;
kmax=1000;
xp=vector(1,kmax);
yp=matrix(1,nvar,1,kmax);

for (i=1;i<=3;i++) {
        for (j=1;j<=3;j++) {
                A[i][j]=0.0;
```

```
                Ah[i][j]=0.0;
        }
}
A[1][1]=2.0/3.0*(-R_0*R_90+2.0*R_0+2.0)/(R_0+1.0);
A[2][2]=2.0/3.0*(2.0*R_0*R_90+2.0*R_0-1.0)/(R_0+1.0);
A[3][3]=2.0/3.0*(2.0*R_0*R_90-R_0+2.0)/(R_0+1.0);
temp=A[1][1]*A[2][2]+A[2][2]*A[3][3]+A[3][3]*A[1][1];
Ah[1][1]=(-A[1][1]+2.0*A[2][2]+2.0*A[3][3])/temp;
Ah[2][2]=(2.0*A[1][1]-A[2][2]+2.0*A[3][3])/temp;
Ah[3][3]=(2.0*A[1][1]+2.0*A[2][2]-A[3][3])/temp;

alpha=AMIN;
srand(unsigned(time(NULL)));
do {
        trial(alpha,&Hc1,&thc1);
        zbrak(fz1,k1,&xb1,&xb2);
        if (xb2 != k1) {
                temp=rtbis1(fr1,xb1,xb2,TOL);
                xp[++kount]=alpha;
                yp[1][kount]=Hc1;
                yp[2][kount]=thc1;
                yp[3][kount]=temp;
                temp=fabs(fmod(xx,PI));
                yp[4][kount]=(temp <= PI/2.0 ? temp : PI-temp);
                for (i=1;i<=IT-1;i++) {
                        zbrak(fz1,k1,&xb1,&xb2);
                        temp=rtbis1(fr1,xb1,xb2,TOL);
                        if (temp > yp[3][kount]) {
                                yp[3][kount]=temp;
                                temp=fabs(fmod(xx,PI));
                                yp[4][kount]=(temp <= PI/2.0 ? temp : PI-temp);
                        }
                }
        }
        else {
                zbrak(fz2,k2,&xb1,&xb2);
                if (xb2 != k2) {
                        temp=rtbis2(fr2,xb1,xb2,TOL);
                        xp[++kount]=alpha;
                        yp[1][kount]=Hc1;
                        yp[2][kount]=thc1;
                        yp[3][kount]=temp;
                        temp=fabs(fmod(xx,PI));
                        yp[4][kount]=(temp <= PI/2.0 ? temp : PI-temp);
```

```
                        for (i=1;i<=IT-1;i++) {
                                zbrak(fz2,k2,&xb1,&xb2);
                                temp=rtbis2(fr2,xb1,xb2,TOL);
                                if (temp > yp[3][kount]) {
                                        yp[3][kount]=temp;
                                        temp=fabs(fmod(xx,PI));
                                        yp[4][kount]=(temp <= PI/2.0 ? temp : PI-
temp);
                                }
                        }
                }
                else {
                        xp[++kount]=alpha;
                        yp[1][kount]=Hc1;
                        yp[2][kount]=thc1;
                        yp[3][kount]=Hc1;
                        yp[4][kount]=thc1;
                }
        }
        alpha=alpha+0.01;
} while (alpha<1.01);

for (i=1;i<=kount;i++) {
        strain(xp[i],ep);
        rho=ep[2]/ep[1];
        if (yp[3][i]<Ht[NMAX]) temp=1.0;
        else splint1(Ht,et,y23,NMAX,yp[3][i],&temp);
        yp[5][i]=rho;
        yp[6][i]=temp*ep[1];
        yp[7][i]=temp*ep[2];
        yp[8][i]=temp;
}

fp=fopen("data.txt","w");
printf("alpha:\n");
fprintf(fp,"alpha:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",xp[i]);
        fprintf(fp,"%f\n",xp[i]);
}
printf("Hc1:\n");
fprintf(fp,"Hc1:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[1][i]);
```

```
        fprintf(fp,"%f\n",yp[1][i]);
}
printf("thc1:\n");
fprintf(fp,"thc1:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[2][i]/DEG);
        fprintf(fp,"%f\n",yp[2][i]/DEG);
}
printf("Hc3:\n");
fprintf(fp,"Hc3:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[3][i]);
        fprintf(fp,"%f\n",yp[3][i]);
}
printf("thc3:\n");
fprintf(fp,"thc3:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[4][i]/DEG);
        fprintf(fp,"%f\n",yp[4][i]/DEG);
}
printf("rho:\n");
fprintf(fp,"rho:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[5][i]);
        fprintf(fp,"%f\n",yp[5][i]);
}
printf("e1:\n");
fprintf(fp,"e1:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[6][i]);
        fprintf(fp,"%f\n",yp[6][i]);
}
printf("e2:\n");
fprintf(fp,"e2:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[7][i]);
        fprintf(fp,"%f\n",yp[7][i]);
}
printf("ee:\n");
fprintf(fp,"ee:\n");
for (i=1;i<=kount;i++) {
        printf("%f\n",yp[8][i]);
        fprintf(fp,"%f\n",yp[8][i]);
}
```

```
        return 0;
}
```

APPENDIX C1

GREEN'S AND GENERALIZED GREEN'S FUNCTIONS

C1.1 Green's Function on an Unbounded Domain

The fundamental interpretation of the Green's function is the response at $\mathbf{r}$ due to a point source located at $\boldsymbol{\rho}$, where $\mathbf{r}$ and $\boldsymbol{\rho}$ denote the radius vectors of an arbitrary point and the point source, respectively. In general, the Green's function should satisfy

$$L\left[G\left(\mathbf{r};\boldsymbol{\rho}\right)\right]=\delta\left(\mathbf{r}-\boldsymbol{\rho}\right), \tag{C1.1}$$

where L denotes a differential operator, which varies from problem to problem, and $\delta\left(\mathbf{r}-\boldsymbol{\rho}\right)$ denotes a one-dimensional or multidimensional Dirac delta function. It can be proven that $G\left(\mathbf{r};\boldsymbol{\rho}\right)=G\left(\boldsymbol{\rho};\mathbf{r}\right)$. This symmetry of the Green's function is referred to as Maxwell's reciprocity. It physically means that the response at $\mathbf{r}$ due to a point source located at $\boldsymbol{\rho}$ is the same as that at $\boldsymbol{\rho}$ due to a point source located at $\mathbf{r}$.

Let $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ denote the unit vectors in the x, y, and z directions, respectively. The Green's function on an unbounded two-dimensional domain should satisfy the two-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G\left(\mathbf{r};\boldsymbol{\rho}\right)=\delta\left(\mathbf{r}-\boldsymbol{\rho}\right) \tag{C1.2}$$

on the unbounded domain, where here $\mathbf{r}$ and $\boldsymbol{\rho}$ take the forms of $\mathbf{r}=x\mathbf{i}+y\mathbf{j}$ and $\boldsymbol{\rho}=\xi\mathbf{i}+\eta\mathbf{j}$, respectively, and $\delta\left(\mathbf{r}-\boldsymbol{\rho}\right)$ takes the form of

$$\delta\left(\mathbf{r}-\boldsymbol{\rho}\right)=\delta\left(x-\xi\right)\delta\left(y-\eta\right). \tag{C1.3}$$

The solution to Eq. (C1.2) can be obtained as

$$G(\mathbf{r};\boldsymbol{\rho}) = \frac{1}{2\pi}\ln|\mathbf{r}-\boldsymbol{\rho}|, \tag{C1.4}$$

where

$$|\mathbf{r}-\boldsymbol{\rho}| = \sqrt{(x-\xi)^2+(y-\eta)^2}. \tag{C1.5}$$

The Green's function on an unbounded three-dimensional domain should satisfy the three-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G(\mathbf{r};\boldsymbol{\rho}) = \delta(\mathbf{r}-\boldsymbol{\rho}) \tag{C1.6}$$

on the unbounded domain, where here $\mathbf{r}$ and $\boldsymbol{\rho}$ take the forms of $\mathbf{r} = x\mathbf{i}+y\mathbf{j}+z\mathbf{k}$ and $\boldsymbol{\rho} = \xi\mathbf{i}+\eta\mathbf{j}+\zeta\mathbf{k}$, respectively, and $\delta(\mathbf{r}-\boldsymbol{\rho})$ takes the form of

$$\delta(\mathbf{r}-\boldsymbol{\rho}) = \delta(x-\xi)\,\delta(y-\eta)\,\delta(z-\zeta). \tag{C1.7}$$

The solution to Eq. (C1.6) can be obtained as

$$G(\mathbf{r};\boldsymbol{\rho}) = -\frac{1}{4\pi|\mathbf{r}-\boldsymbol{\rho}|}, \tag{C1.8}$$

where

$$|\mathbf{r}-\boldsymbol{\rho}| = \sqrt{(x-\xi)^2+(y-\eta)^2+(z-\zeta)^2}. \tag{C1.9}$$

C1.2 Generalized Green's Function on a Rectangular Domain

The Green's fuctions satisfying Poisson's equations on various domains have been extensively studied. For example, Gao and Rowlands [102] derived the Green's functions satisfying Poisson's equations on cuboid, cylindrical, and spherical domains, respectively, subject to homogeneous Dirichlet boundary conditions. According to

Section 4.1.3 and 4.1.4, the required generalized Green's functions are subject to nonhomogeneous Neumann boundary conditions. For this reason, these generalized Green's functions should be of different forms compared with the aforementioned Green's functions and need to be derived. Specifically, according to Section 4.1.3, the generalized Green's function on a rectangular domain should satisfy the two-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G\left(x,y;\xi,\eta\right)=\delta\left(x-\xi\right)\delta\left(y-\eta\right) \tag{C1.10}$$

subject to the nonhomogeneous boundary conditions

$$\begin{aligned}\left.\frac{\partial G}{\partial x}\right|_{x=0}&=0,\quad \left.\frac{\partial G}{\partial x}\right|_{x=b}=\frac{1}{2c},\\ \left.\frac{\partial G}{\partial y}\right|_{y=0}&=0,\quad \left.\frac{\partial G}{\partial y}\right|_{y=c}=\frac{1}{2b}.\end{aligned} \tag{C1.11}$$

G can be decomposed into two parts, that is,

$$G\left(x,y;\xi,\eta\right)=G_1\left(x,y;\xi,\eta\right)+G_2\left(x,y;\xi,\eta\right), \tag{C1.12}$$

where G_1 satisfies the two-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G_1\left(x,y;\xi,\eta\right)=\delta\left(x-\xi\right)\delta\left(y-\eta\right)-\frac{1}{bc} \tag{C1.13}$$

subject to the homogenous boundary conditions

$$\begin{aligned}-\left.\frac{\partial G_1}{\partial x}\right|_{x=-\frac{b}{2}}&=\left.\frac{\partial G_1}{\partial x}\right|_{x=\frac{b}{2}}=0,\\ -\left.\frac{\partial G_1}{\partial y}\right|_{y=-\frac{c}{2}}&=\left.\frac{\partial G_1}{\partial y}\right|_{y=\frac{c}{2}}=0.\end{aligned} \tag{C1.14}$$

and G_2 satisfies the two-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G_2\left(x,y;\xi,\eta\right)=\frac{1}{bc} \tag{C1.15}$$

subject to the nonhomogeneous boundary conditions

$$\begin{aligned} -\left.\frac{\partial G_2}{\partial x}\right|_{x=-\frac{b}{2}} &= \left.\frac{\partial G_2}{\partial x}\right|_{x=\frac{b}{2}} = \frac{1}{4c}, \\ -\left.\frac{\partial G_2}{\partial y}\right|_{y=-\frac{c}{2}} &= \left.\frac{\partial G_2}{\partial y}\right|_{y=\frac{c}{2}} = \frac{1}{4b}. \end{aligned} \tag{C1.16}$$

G_2 can be easily obtained as

$$G_2\left(x, y; \xi, \eta\right) = \frac{x^2 + \xi^2 + y^2 + \eta^2}{4bc}. \tag{C1.17}$$

To reduce the complexity of derivations, introduce a coordinate system $Ox'y'z'$, where x', y', and z' can be related to x, y, and z by

$$x' = x + \frac{b}{2}, \quad y' = y + \frac{c}{2}, \quad z' = z. \tag{C1.18}$$

Now G_1 should satisfy the two-dimensional Poisson's equation

$$\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} G_1\left(x', y'; \xi', \eta'\right) = \delta\left(x' - \xi'\right)\delta\left(y' - \eta'\right) - \frac{1}{bc} \tag{C1.19}$$

subject to the homogenous boundary conditions

$$\begin{aligned} -\left.\frac{\partial G_1}{\partial x'}\right|_{x'=0} &= \left.\frac{\partial G_1}{\partial x'}\right|_{x'=b} = 0, \\ -\left.\frac{\partial G_1}{\partial y'}\right|_{y'=0} &= \left.\frac{\partial G_1}{\partial y'}\right|_{y'=c} = 0. \end{aligned} \tag{C1.20}$$

G_1 can be solved by expanding it in terms of the eigenfunctions $\phi_{mn}\left(x', y'\right)$ of the partial differential equation

$$\frac{\partial^2 \phi}{\partial x'^2} + \frac{\partial^2 \phi}{\partial y'^2} = -\lambda \phi \tag{C1.21}$$

and Eq. (C1.14). The eigenvalues and their corresponding eigenfunctions of Eq. (C1.21) can be obtained as

$$\lambda_{mn} = \left(\frac{m\pi}{b}\right)^2 + \left(\frac{n\pi}{c}\right)^2, \tag{C1.22}$$

$$\phi_{mn}\left(x', y'\right) = \cos\frac{m\pi x'}{b}\cos\frac{n\pi y'}{c}, \tag{C1.23}$$

where $m = 0, 1, 2\ldots$ and $n = 0, 1, 2\ldots$. G_1 can then be expressed as

$$G_1\left(x', y'; \xi', \eta'\right) = \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} A_{mn}\cos\frac{m\pi x'}{b}\cos\frac{n\pi y'}{c}. \tag{C1.24}$$

Substituting Eq. (C1.24) into Eq. (C1.13) gives

$$\begin{aligned} &-\sum_{n=1}^{\infty}\left(\frac{n\pi}{c}\right)^2 A_{0n}\cos\frac{n\pi y'}{c} - \sum_{m=1}^{\infty}\left(\frac{m\pi}{b}\right)^2 A_{m0}\cos\frac{m\pi x'}{b} \\ &-\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\left[\left(\frac{m\pi}{b}\right)^2 + \left(\frac{n\pi}{c}\right)^2\right] A_{mn}\cos\frac{m\pi x'}{b}\cos\frac{n\pi y'}{c} \\ &\qquad = \delta\left(x' - \xi'\right)\delta\left(y' - \eta'\right) - \frac{1}{bc}. \end{aligned} \tag{C1.25}$$

Eq. (C1.25) indicates that A_{mn} can be obtained by finding the corresponding double Fourier coefficients for the function to the right of the equal sign. A_{0n}, A_{m0}, and A_{mn} can hereby be obtained as

$$\begin{aligned} A_{0n} &= -\frac{2}{bc}\frac{\cos\dfrac{n\pi\eta'}{c}}{\left(\dfrac{n\pi}{c}\right)^2}, \quad A_{m0} = -\frac{2}{bc}\frac{\cos\dfrac{m\pi\xi'}{b}}{\left(\dfrac{m\pi}{b}\right)^2}, \\ A_{mn} &= -\frac{4}{bc}\frac{\cos\dfrac{m\pi\xi'}{b}\cos\dfrac{n\pi\eta'}{c}}{\left(\dfrac{m\pi}{b}\right)^2 + \left(\dfrac{n\pi}{c}\right)^2}. \end{aligned} \tag{C1.26}$$

Especially, for $m = 0$ and $n = 0$,

$$\frac{1}{bc}\int_0^c\int_0^b\left[\delta\left(x' - \xi'\right)\delta\left(y' - \eta'\right) - \frac{1}{bc}\right]dx'dy' = 0. \tag{C1.27}$$

In Eq. (C1.25), the term corresponding to A_{00} also vanishes. This implies that, no matter which value A_{00} takes, Eq. (C1.25) can always be satisfied. Therefore, it is doable to set A_{00} to be zero here. It now turns out that the term $-1/bc$ in Eq. (C1.13) is specially selected to let the integral in Eq. (C1.27) vanish. If this term took another value, Eq. (C1.25) would never be satisfied, and G_1 would not exist. Substituting Eq. (C1.26) into Eq. (C1.24) and noting that $A_{00}=0$ give G_1 as

$$G_1\left(x',y';\xi',\eta'\right)=-\frac{2}{bc}\sum_{n=1}^{\infty}\frac{\cos\dfrac{n\pi y'}{c}\cos\dfrac{n\pi\eta'}{c}}{\left(\dfrac{n\pi}{c}\right)^2}-\frac{2}{bc}\sum_{m=1}^{\infty}\frac{\cos\dfrac{m\pi x'}{b}\cos\dfrac{m\pi\xi'}{b}}{\left(\dfrac{m\pi}{b}\right)^2}$$
$$-\frac{4}{bc}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{\cos\dfrac{m\pi x'}{b}\cos\dfrac{m\pi\xi'}{b}\cos\dfrac{n\pi y'}{c}\cos\dfrac{n\pi\eta'}{c}}{\left(\dfrac{m\pi}{b}\right)^2+\left(\dfrac{n\pi}{c}\right)^2}. \tag{C1.28}$$

Substituting Eqs. (C1.17) and (C1.28) into Eq. (C1.12) gives the expression for G. For the special case of Section 4.1.3, the point source is located at the origin in $Oxyz$ or at $(b/2,c/2)$ in $Ox'y'z'$. Substituting $\xi'=b/2$ and $\eta'=c/2$ into the expression for G gives

$$G\left(x',y';\frac{b}{2},\frac{c}{2}\right)=-\frac{2}{bc}\sum_{n=1}^{\infty}\frac{(-1)^n\cos\dfrac{2n\pi y'}{c}}{\left(\dfrac{2n\pi}{c}\right)^2}-\frac{2}{bc}\sum_{m=1}^{\infty}\frac{(-1)^m\cos\dfrac{2m\pi x'}{b}}{\left(\dfrac{2m\pi}{b}\right)^2}$$
$$-\frac{4}{bc}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{(-1)^{m+n}\cos\dfrac{2m\pi x'}{b}\cos\dfrac{2n\pi y'}{c}}{\left(\dfrac{2m\pi}{b}\right)^2+\left(\dfrac{2n\pi}{c}\right)^2}.$$
$$+\frac{\left(x'-\dfrac{b}{2}\right)^2+\left(\dfrac{b}{2}\right)^2+\left(y'-\dfrac{c}{2}\right)^2+\left(\dfrac{c}{2}\right)^2}{4bc} \tag{C1.29}$$

Fig. C1.1 shows the 3D color map surface of $G\left(x', y'; b/2, c/2\right)$. Unfortunately, the Fourier series for $G_{,xx}$, $G_{,yy}$, and $G_{,xy}$ do not converge. Although the Fejér sum can be used to approximate these Fourier series, it is more convenient to calculate the numerical derivatives of G. In this part, Ridders' method is used to calculate these numerical derivatives [96].

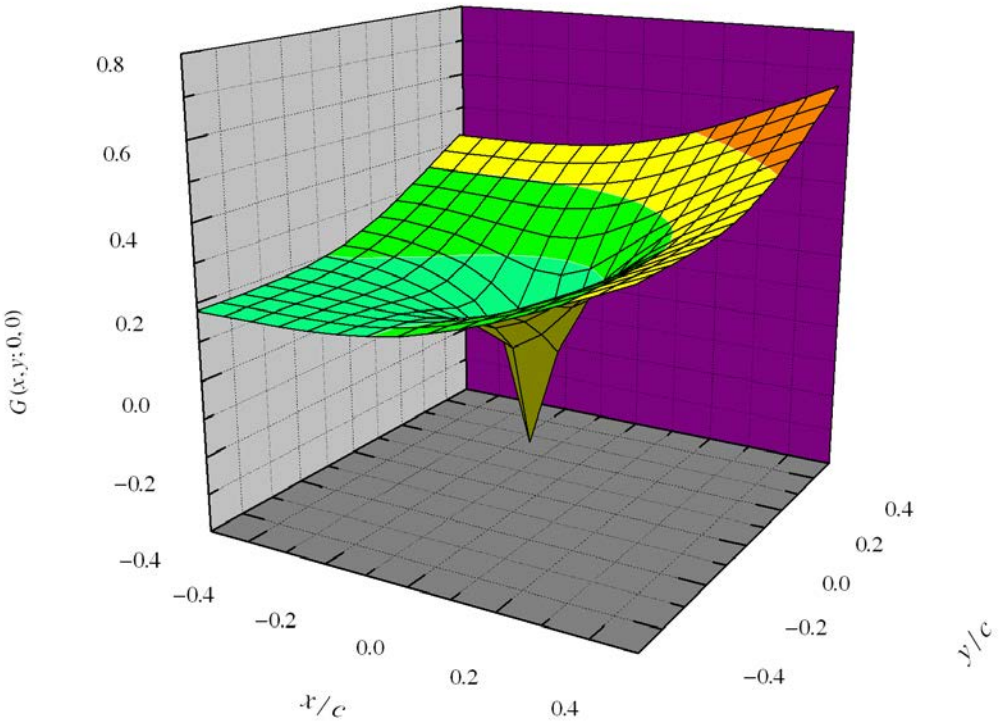

Fig. C1.1. 3D color map surface of $G\left(x, y; 0, 0\right)$ for $b/c = 1$.

C1.3 Generalized Green's Function on a Cuboid Domain

According to Section 4.1.5, the generalized Green's function on a cuboid domain should satisfy the three-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G\left(x,y,z;\xi,\eta,\zeta\right)=\delta\left(x-\xi\right)\delta\left(y-\eta\right)\delta\left(z-\zeta\right) \tag{C1.30}$$

subject to the nonhomogeneous boundary conditions

$$\begin{aligned}
-\left.\frac{\partial G}{\partial x}\right|_{x=-\frac{b}{2}} &= \left.\frac{\partial G}{\partial x}\right|_{x=\frac{b}{2}} = \frac{1}{6cd},\\
-\left.\frac{\partial G}{\partial y}\right|_{y=-\frac{c}{2}} &= \left.\frac{\partial G}{\partial y}\right|_{y=\frac{c}{2}} = \frac{1}{6bd},\\
-\left.\frac{\partial G}{\partial z}\right|_{z=-\frac{d}{2}} &= \left.\frac{\partial G}{\partial z}\right|_{z=\frac{d}{2}} = \frac{1}{6bc}.
\end{aligned} \tag{C1.31}$$

Similarly to Section C1.2, G can be decomposed into two parts, that is,

$$G\left(x,y,z;\xi,\eta,\zeta\right)=G_1\left(x,y,z;\xi,\eta,\zeta\right)+G_2\left(x,y,z;\xi,\eta,\zeta\right), \tag{C1.32}$$

where G_1 satisfies the two-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G_1\left(x,y,z;\xi,\eta,\zeta\right)=\delta\left(x-\xi\right)\delta\left(y-\eta\right)\delta\left(z-\zeta\right)-\frac{1}{bcd} \tag{C1.33}$$

subject to the homogeneous boundary conditions

$$\begin{aligned}
-\left.\frac{\partial G}{\partial x}\right|_{x=-\frac{b}{2}} &= \left.\frac{\partial G}{\partial x}\right|_{x=\frac{b}{2}} = 0,\\
-\left.\frac{\partial G}{\partial y}\right|_{y=-\frac{c}{2}} &= \left.\frac{\partial G}{\partial y}\right|_{y=\frac{c}{2}} = 0,\\
-\left.\frac{\partial G}{\partial z}\right|_{z=-\frac{d}{2}} &= \left.\frac{\partial G}{\partial z}\right|_{z=\frac{d}{2}} = 0.
\end{aligned} \tag{C1.34}$$

and G_2 satisfies the two-dimensional Poisson's equation

$$\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}G_2\left(x,y,z;\xi,\eta,\zeta\right)=\frac{1}{bcd} \tag{C1.35}$$

subject to the nonhomogeneous boundary conditions

$$
\begin{aligned}
-\left.\frac{\partial G_2}{\partial x}\right|_{x=-\frac{b}{2}} &= \left.\frac{\partial G_2}{\partial x}\right|_{x=\frac{b}{2}} = \frac{1}{6cd}, \\
-\left.\frac{\partial G_2}{\partial y}\right|_{y=-\frac{c}{2}} &= \left.\frac{\partial G_2}{\partial y}\right|_{y=\frac{c}{2}} = \frac{1}{6bd}, \\
-\left.\frac{\partial G_2}{\partial z}\right|_{z=-\frac{d}{2}} &= \left.\frac{\partial G_2}{\partial z}\right|_{z=\frac{d}{2}} = \frac{1}{6bc}.
\end{aligned} \tag{C1.36}
$$

G_2 can be easily obtained as

$$
G_2\left(x, y, z; \xi, \eta, \zeta\right) = \frac{x^2 + \xi^2 + y^2 + \eta^2 + z^2 + \zeta^2}{6bcd}. \tag{C1.37}
$$

To reduce the complexity of derivations, introduce a coordinate system $Ox'y'z'$, where x', y', and z' can be related to x, y, and z by

$$
x' = x + \frac{b}{2}, \quad y' = y + \frac{c}{2}, \quad z' = z + \frac{d}{2}. \tag{C1.38}
$$

Now G_1 should satisfy the two-dimensional Poisson's equation

$$
\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} G_1\left(x', y', z'; \xi', \eta', \zeta'\right) = \delta\left(x' - \xi'\right)\delta\left(y' - \eta'\right)\delta\left(z' - \zeta'\right) - \frac{1}{bcd} \tag{C1.39}
$$

subject to the homogenous boundary conditions

$$
\begin{aligned}
-\left.\frac{\partial G_1}{\partial x'}\right|_{x'=0} &= \left.\frac{\partial G_1}{\partial x'}\right|_{x'=b} = 0, \\
-\left.\frac{\partial G_1}{\partial y'}\right|_{y'=0} &= \left.\frac{\partial G_1}{\partial y'}\right|_{y'=c} = 0, \\
-\left.\frac{\partial G_1}{\partial z'}\right|_{z'=0} &= \left.\frac{\partial G_1}{\partial z'}\right|_{z'=d} = 0.
\end{aligned} \tag{C1.40}
$$

The derivation for G_1 is quite similar to that in Section C1.2. Finally, G_1 can be obtained as

$$G_1(x',y',z';\xi',\eta',\zeta') = -\frac{2}{bcd}\sum_{n=1}^{\infty}\frac{\cos\frac{n\pi z'}{d}\cos\frac{n\pi\zeta'}{d}}{\left(\frac{n\pi}{d}\right)^2}$$

$$-\frac{2}{bcd}\sum_{l=1}^{\infty}\frac{\cos\frac{l\pi x'}{b}\cos\frac{l\pi\xi'}{b}}{\left(\frac{l\pi}{b}\right)^2} - \frac{2}{bcd}\sum_{m=1}^{\infty}\frac{\cos\frac{m\pi y'}{c}\cos\frac{m\pi\eta'}{c}}{\left(\frac{m\pi}{c}\right)^2}$$

$$-\frac{4}{bcd}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{\cos\frac{m\pi y'}{c}\cos\frac{m\pi\eta'}{c}\cos\frac{n\pi z'}{d}\cos\frac{n\pi\zeta'}{d}}{\left(\frac{m\pi}{c}\right)^2+\left(\frac{n\pi}{d}\right)^2}$$

$$-\frac{4}{bcd}\sum_{l=1}^{\infty}\sum_{n=1}^{\infty}\frac{\cos\frac{l\pi x'}{b}\cos\frac{l\pi\xi'}{b}\cos\frac{n\pi z'}{d}\cos\frac{n\pi\zeta'}{d}}{\left(\frac{l\pi}{b}\right)^2+\left(\frac{n\pi}{d}\right)^2}$$

$$-\frac{4}{bcd}\sum_{l=1}^{\infty}\sum_{m=1}^{\infty}\frac{\cos\frac{l\pi x'}{b}\cos\frac{l\pi\xi'}{b}\cos\frac{m\pi y'}{c}\cos\frac{m\pi\eta'}{c}}{\left(\frac{l\pi}{b}\right)^2+\left(\frac{m\pi}{c}\right)^2}$$

$$-\frac{8}{bcd}\sum_{l=1}^{\infty}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{\cos\frac{l\pi x'}{b}\cos\frac{l\pi\xi'}{b}\cos\frac{m\pi y'}{c}\cos\frac{m\pi\eta'}{c}\cos\frac{n\pi z'}{d}\cos\frac{n\pi\zeta'}{d}}{\left(\frac{l\pi}{b}\right)^2+\left(\frac{m\pi}{c}\right)^2+\left(\frac{n\pi}{d}\right)^2}. \quad \text{(C1.41)}$$

Substituting Eqs. (C1.37) and (C1.41) into Eq. (C1.32) gives the expression for G. For the special case of Section 4.1.3, the point source is located at the origin in $Ox'y'z'$ or at $(b/2, c/2, d/2)$ in $Ox'y'z'$. Substituting $\xi' = b/2$, $\eta' = c/2$, and $\zeta' = d/2$ into the expression for G gives

$$G\left(x',y',z';\frac{b}{2},\frac{c}{2},\frac{d}{2}\right)=-\frac{2}{bcd}\sum_{n=1}^{\infty}\frac{(-1)^n\cos\dfrac{2n\pi z'}{d}}{\left(\dfrac{2n\pi}{d}\right)^2}-\frac{2}{bcd}\sum_{l=1}^{\infty}\frac{(-1)^l\cos\dfrac{2l\pi x'}{b}}{\left(\dfrac{2l\pi}{b}\right)^2}$$

$$-\frac{2}{bcd}\sum_{m=1}^{\infty}\frac{(-1)^m\cos\dfrac{2m\pi y'}{c}}{\left(\dfrac{2m\pi}{c}\right)^2}-\frac{4}{bcd}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{(-1)^{m+n}\cos\dfrac{2m\pi y'}{c}\cos\dfrac{2n\pi z'}{d}}{\left(\dfrac{2m\pi}{c}\right)^2+\left(\dfrac{2n\pi}{d}\right)^2}$$

$$-\frac{4}{bcd}\sum_{l=1}^{\infty}\sum_{n=1}^{\infty}\frac{(-1)^{l+n}\cos\dfrac{2l\pi x'}{b}\cos\dfrac{2n\pi z'}{d}}{\left(\dfrac{2l\pi}{b}\right)^2+\left(\dfrac{2n\pi}{d}\right)^2}$$

$$-\frac{4}{bcd}\sum_{l=1}^{\infty}\sum_{m=1}^{\infty}\frac{(-1)^{l+m}\cos\dfrac{2l\pi x'}{b}\cos\dfrac{2m\pi y'}{c}}{\left(\dfrac{2l\pi}{b}\right)^2+\left(\dfrac{2m\pi}{c}\right)^2}$$

$$-\frac{8}{bcd}\sum_{l=1}^{\infty}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{(-1)^{l+m+n}\cos\dfrac{2l\pi x'}{b}\cos\dfrac{2m\pi y'}{c}\cos\dfrac{2n\pi z'}{d}}{\left(\dfrac{2l\pi}{b}\right)^2+\left(\dfrac{2m\pi}{c}\right)^2+\left(\dfrac{2n\pi}{d}\right)^2}$$

$$+\frac{\left(x'-\dfrac{b}{2}\right)^2+\left(\dfrac{b}{2}\right)^2+\left(y'-\dfrac{c}{2}\right)^2+\left(\dfrac{c}{2}\right)^2+\left(z'-\dfrac{d}{2}\right)^2+\left(\dfrac{d}{2}\right)^2}{6bcd}.$$

(C1.42)

Similarly to Section C1.2, the numerical derivatives of G are calculated instead of the Fourier series for $G_{,xx}$, $G_{,yy}$, $G_{,zz}$, and etc..

APPENDIX C2

C PROGRAM FOR SECTION 4

“nrutil.h” //The same as that in Appednix A.

“nutril.cpp” //The same as that in Appednix A.

“main.cpp”

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include "nrutil.h"
#include "nrutil.cpp"
#define PI 3.14159265358979

#define ITMAX 100 //Maximum allowed number of iterations.
#define EPS 1.0e-4
//#define EPS 3.0e-8 //Machine doubleing-point precision.

#define TOL 3.0e-8 //Tolerance passed to brent.

FILE *fp;

double A,B,C,a,b,c,f;
int m,n;
double h;

int nvar; //Variables that you must define and set in your main program.
int kmax,kount; //Communicates with odeint.
double *xp,**yp,dxsav;
/*User storage for intermediate results. Preset kmax and dxsav in the calling program. If
kmax 6=0 results are stored at approximate intervals dxsav in the arrays xp[1..kount],
yp[1..nvar][1..kount], where kount is output by odeint. Defining declarations for these
variables, with memory allocations xp[1..kmax] and yp[1..nvar][1..kmax] for the arrays,
should be in the calling program.*/

double G(double x, double y)
{
        int i,j;
        double lambda,mu,sum;
```

```
	sum=0.0;

	for (i=1;i<=n;i++) {
		lambda=2*i*PI/b;
		sum += -2.0/(b*c)*pow(-1.0,i)*cos(lambda*x)/(lambda*lambda);
	}
	for (j=1;j<=n;j++) {
		mu=2*j*PI/c;
		sum += -2.0/(b*c)*pow(-1.0,j)*cos(mu*y)/(mu*mu);
	}
	for (i=1;i<=n;i++) {
		for (j=1;j<=n;j++) {
			lambda=2*i*PI/b;
			mu=2*j*PI/c;
			sum += -4.0/(b*c)*pow(-
1.0,i+j)*cos(lambda*x)*cos(mu*y)/(lambda*lambda+mu*mu);
		}
	}
	sum += (x*x+b*b/4.0+y*y+c*c/4.0)/(4.0*b*c);

	return sum;
}

double de_edA(double x, double y, double h)
{
	double G(double x, double y);
	double dxx(double (*func)(double, double), double x, double y, double h, double
*err);
	double dyy(double (*func)(double, double), double x, double y, double h, double
*err);
	double dxy(double (*func)(double, double), double x, double y, double h, double
*err);

	double G_xx,G_yy,G_xy,temp;
	double *err;

	err=vector(1,1);

//	G_xx=dxx(G,x,y,h,err);
	G_yy=dyy(G,x,y,h,err);
	G_xx=-G_yy;
//	G_xx=0.5*(dxx(G,x,y,h,err)-dyy(G,x,y,h,err));
//	G_yy=-G_xx; //G_xx+G_yy=0
```

```
	G_xy=dxy(G,x,y,h,err);
	temp=sqrt(6.0*A*A+B*B*(G_xx*G_xx+G_yy*G_yy+2.0*G_xy*G_xy));

	free_vector(err,1,1);

	return sqrt(2.0/3.0)*6.0*A/temp;
}

double de_edB(double x, double y, double h)
{
	double G(double x, double y);
	double dxx(double (*func)(double, double), double x, double y, double h, double
*err);
	double dyy(double (*func)(double, double), double x, double y, double h, double
*err);
	double dxy(double (*func)(double, double), double x, double y, double h, double
*err);

	double G_xx,G_yy,G_xy,temp;
	double *err;

	err=vector(1,1);

//	G_xx=dxx(G,x,y,h,err);
//	G_yy=dyy(G,x,y,h,err);
	G_xx=0.5*(dxx(G,x,y,h,err)-dyy(G,x,y,h,err));
	G_yy=-G_xx; //G_xx+G_yy=0
	G_xy=dxy(G,x,y,h,err);
	temp=sqrt(6.0*A*A+B*B*(G_xx*G_xx+G_yy*G_yy+2.0*G_xy*G_xy));

	free_vector(err,1,1);

	return sqrt(2.0/3.0)*B*(G_xx*G_xx+G_yy*G_yy+2.0*G_xy*G_xy)/temp;
}

#define CON 1.4 //Stepsize is decreased by CON at each iteration.
#define CON2 (CON*CON)
#define BIG 1.0e30
#define NTAB 10 //Sets maximum size of tableau.
#define SAFE 2.0 //Return when error is SAFE worse than the best so far.

double dxx(double (*func)(double, double), double x, double y, double h, double *err)
/*Returns the derivative of a function func at a point x by Ridders' method of polynomial
extrapolation. The value h is input as an estimated initial stepsize; it need not be small,
```

```
but rather should be an increment in x over which func changes substantially. An
estimate of the error in the derivative is returned as err.*/
{
	int i,j;
	double errt,fac,h1,h2,**a,ans;

	if (h == 0.0) nrerror("h must be nonzero in dfridr.");
	a=matrix(1,NTAB,1,NTAB);
	h1=h*b;
	h2=h*c;
	a[1][1]=((*func)(x+h1,y)-2.0*(*func)(x,y)+(*func)(x-h1,y))/(h1*h1);
	*err=BIG;
	for (i=2;i<=NTAB;i++) {
		//Successive columns in the Neville tableau will go to smaller stepsizes
and higher orders of
		//extrapolation.
		h1 /= CON;
		h2 /= CON;
		a[1][i]=((*func)(x+h1,y)-2.0*(*func)(x,y)+(*func)(x-h1,y))/(h1*h1);
//Try new, smaller step-
		fac=CON2; //size.
		for (j=2;j<=i;j++) { //Compute extrapolations of various orders, requiring
			//no new function evaluations.
			a[j][i]=(a[j-1][i]*fac-a[j-1][i-1])/(fac-1.0);
			fac=CON2*fac;
			errt=FMAX(fabs(a[j][i]-a[j-1][i]),fabs(a[j][i]-a[j-1][i-1]));
			//The error strategy is to compare each new extrapolation to one
order lower, both
			//at the present stepsize and the previous one.
			if (errt <= *err) { //If error is decreased, save the improved
answer.
				*err=errt;
				ans=a[j][i];
			}
		}
		if (fabs(a[i][i]-a[i-1][i-1]) >= SAFE*(*err)) break;
		//If higher order is worse by a significant factor SAFE, then quit early.
	}
	free_matrix(a,1,NTAB,1,NTAB);
	return ans;
}

double dyy(double (*func)(double, double), double x, double y, double h, double *err)
```

```
/*Returns the derivative of a function func at a point x by Ridders' method of polynomial
extrapolation. The value h is input as an estimated initial stepsize; it need not be small,
but rather should be an increment in x over which func changes substantially. An
estimate of the error in the derivative is returned as err.*/
{
	int i,j;
	double errt,fac,h1,h2,**a,ans;

	if (h == 0.0) nrerror("h must be nonzero in dfridr.");
	a=matrix(1,NTAB,1,NTAB);
	h1=h*b;
	h2=h*c;
	a[1][1]=((*func)(x,y+h2)-2.0*(*func)(x,y)+(*func)(x,y-h2))/(h2*h2);
	*err=BIG;
	for (i=2;i<=NTAB;i++) {
		//Successive columns in the Neville tableau will go to smaller stepsizes
and higher orders of
		//extrapolation.
		h1 /= CON;
		h2 /= CON;
		a[1][i]=((*func)(x,y+h2)-2.0*(*func)(x,y)+(*func)(x,y-h2))/(h2*h2);
//Try new, smaller step-
		fac=CON2; //size.
		for (j=2;j<=i;j++) { //Compute extrapolations of various orders, requiring
			//no new function evaluations.
			a[j][i]=(a[j-1][i]*fac-a[j-1][i-1])/(fac-1.0);
			fac=CON2*fac;
			errt=FMAX(fabs(a[j][i]-a[j-1][i]),fabs(a[j][i]-a[j-1][i-1]));
			//The error strategy is to compare each new extrapolation to one
order lower, both
			//at the present stepsize and the previous one.
			if (errt <= *err) { //If error is decreased, save the improved
answer.
				*err=errt;
				ans=a[j][i];
			}
		}
		if (fabs(a[i][i]-a[i-1][i-1]) >= SAFE*(*err)) break;
		//If higher order is worse by a significant factor SAFE, then quit early.
	}
	free_matrix(a,1,NTAB,1,NTAB);
	return ans;
}
```

```
double dxy(double (*func)(double, double), double x, double y, double h, double *err)
/*Returns the derivative of a function func at a point x by Ridders' method of polynomial
extrapolation. The value h is input as an estimated initial stepsize; it need not be small,
but rather should be an increment in x over which func changes substantially. An
estimate of the error in the derivative is returned as err.*/
{
	int i,j;
	double errt,fac,h1,h2,**a,ans;

	if (h == 0.0) nrerror("h must be nonzero in dfridr.");
	a=matrix(1,NTAB,1,NTAB);
	h1=h*b;
	h2=h*c;
	a[1][1]=(((*func)(x+h1,y+h2)-(*func)(x+h1,y-h2))-((*func)(x-h1,y+h2)-
(*func)(x-h1,y-h2)))/(4.0*h1*h2);
	*err=BIG;
	for (i=2;i<=NTAB;i++) {
		//Successive columns in the Neville tableau will go to smaller stepsizes
and higher orders of
		//extrapolation.
		h1 /= CON;
		h2 /= CON;
		a[1][i]=(((*func)(x+h1,y+h2)-(*func)(x+h1,y-h2))-((*func)(x-h1,y+h2)-
(*func)(x-h1,y-h2)))/(4.0*h1*h2); //Try new, smaller step-
		fac=CON2; //size.
		for (j=2;j<=i;j++) { //Compute extrapolations of various orders, requiring
			//no new function evaluations.
			a[j][i]=(a[j-1][i]*fac-a[j-1][i-1])/(fac-1.0);
			fac=CON2*fac;
			errt=FMAX(fabs(a[j][i]-a[j-1][i]),fabs(a[j][i]-a[j-1][i-1]));
			//The error strategy is to compare each new extrapolation to one
order lower, both
			//at the present stepsize and the previous one.
			if (errt <= *err) { //If error is decreased, save the improved
answer.
				*err=errt;
				ans=a[j][i];
			}
		}
		if (fabs(a[i][i]-a[i-1][i-1]) >= SAFE*(*err)) break;
		//If higher order is worse by a significant factor SAFE, then quit early.
	}
	free_matrix(a,1,NTAB,1,NTAB);
	return ans;
```

```
}

double qgaus(double (*func)(double, double), double a, double b, double h)
/*Returns the integral of the function func between a and b, by ten-point Gauss-
Legendre integration: the function is evaluated exactly ten times at interior points in the
range of integration.*/
{
	int j;
	double xr,xm,dx,s;
	static double x[]={0.0,0.1488743389,0.4333953941, //The abscissas and weights.
		//First value of each array
		//not used.
		0.6794095682,0.8650633666,0.9739065285};
	static double w[]={0.0,0.2955242247,0.2692667193,
		0.2190863625,0.1494513491,0.0666713443};

	xm=0.5*(b+a);
	xr=0.5*(b-a);
	s=0; //Will be twice the average value of the function, since the
	//ten weights (fove numbers above each used twice)
	//sum to 2.
	for (j=1;j<=5;j++) {
		dx=xr*x[j];
		s += w[j]*((*func)(xm+dx,h)+(*func)(xm-dx,h));
	}
	return s *= xr; //Scale the answer to the range of integration.
}

static double xsav;
static double (*nrfunc)(double, double, double);

double quad2d_1(double (*func)(double, double, double), double x1, double x2, double
h)
/*Returns the integral of a user-supplied function func over a three-dimensional region
specified
by the limits x1, x2, and by the user-supplied functions yy1, yy2, z1, and z2, as defined
in
(4.6.2). (The functions y1 and y2 are here called yy1 and yy2 to avoid conflict with the
names
of Bessel functions in some C libraries). Integration is performed by calling qgaus
recursively.*/
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f1_1(double x, double h);
```

```
	nrfunc=func;
	return qgaus(f1_1,x1,x2,h);
}

double f1_1(double x, double h) //This is H of eq. (4.6.5).
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f2_1(double y, double h);
	double yy1_1(double),yy2_1(double);

	xsav=x;
	return qgaus(f2_1,yy1_1(x),yy2_1(x),h);
}

double f2_1(double y, double h) //This is G of eq. (4.6.4).
{
	return (*nrfunc)(xsav,y,h);
}

double yy1_1(double x)
{
	return sqrt(a*a-(x-0.5*b)*(x-0.5*b))+0.5*c;
}

double yy2_1(double x)
{
	return a+0.5*c;
}

double quad2d_2(double (*func)(double, double, double), double x1, double x2, double
h)
/*Returns the integral of a user-supplied function func over a three-dimensional region
specified by the limits x1, x2, and by the user-supplied functions yy1, yy2, z1, and z2, as
defined in (4.6.2). (The functions y1 and y2 are here called yy1 and yy2 to avoid conflict
with the names of Bessel functions in some C libraries). Integration is performed by
calling qgaus recursively.*/
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f1_2(double x, double h);
	nrfunc=func;
	return qgaus(f1_2,x1,x2,h);
}

double f1_2(double x, double h) //This is H of eq. (4.6.5).
```

```
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f2_2(double y, double h);
	double yy1_2(double),yy2_2(double);

	xsav=x;
	return qgaus(f2_2,yy1_2(x),yy2_2(x),h);
}

double f2_2(double y, double h) //This is G of eq. (4.6.4).
{
	return (*nrfunc)(xsav,y,h);
}

double yy1_2(double x)
{
	return 0.5*c;
}

double yy2_2(double x)
{
	return a+0.5*c;
}

double quad2d_3(double (*func)(double, double, double), double x1, double x2, double
h)
/*Returns the integral of a user-supplied function func over a three-dimensional region
specified by the limits x1, x2, and by the user-supplied functions yy1, yy2, z1, and z2, as
defined in (4.6.2). (The functions y1 and y2 are here called yy1 and yy2 to avoid conflict
with the names of Bessel functions in some C libraries). Integration is performed by
calling qgaus recursively.*/
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f1_3(double x, double h);
	nrfunc=func;
	return qgaus(f1_3,x1,x2,h);
}

double f1_3(double x, double h) //This is H of eq. (4.6.5).
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f2_3(double y, double h);
	double yy1_3(double),yy2_3(double);
```

```
	xsav=x;
	return qgaus(f2_3,yy1_3(x),yy2_3(x),h);
}

double f2_3(double y, double h) //This is G of eq. (4.6.4).
{
	return (*nrfunc)(xsav,y,h);
}

double yy1_3(double x)
{
	return a+0.5*c;
}

double yy2_3(double x)
{
	return c;
}

double quad2d_4(double (*func)(double, double, double), double x1, double x2, double
h)
/*Returns the integral of a user-supplied function func over a three-dimensional region
specified by the limits x1, x2, and by the user-supplied functions yy1, yy2, z1, and z2, as
defined in (4.6.2). (The functions y1 and y2 are here called yy1 and yy2 to avoid conflict
with the names of Bessel functions in some C libraries). Integration is performed by
calling qgaus recursively.*/
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f1_4(double x, double h);
	nrfunc=func;
	return qgaus(f1_4,x1,x2,h);
}

double f1_4(double x, double h) //This is H of eq. (4.6.5).
{
	double qgaus(double (*func)(double, double), double a, double b, double h);
	double f2_4(double y, double h);
	double yy1_4(double),yy2_4(double);

	xsav=x;
	return qgaus(f2_4,yy1_4(x),yy2_4(x),h);
}

double f2_4(double y, double h) //This is G of eq. (4.6.4).
```

```
{
	return (*nrfunc)(xsav,y,h);
}

double yy1_4(double x)
{
	return a+0.5*c;
}

double yy2_4(double x)
{
	return c;
}

int main(void)
// Main program.
{
	double epsilon_e(double x, double y, double h);

	int i,j,k;
	double r,chi,temp;
	double *err,**ep;

	f=0.02;
	b=1.0;
	c=1.0;
	a=sqrt(f*b*c/PI);
	n=20;
	k=20;
	kmax=2*k+1;
	nvar=4;
	xp=vector(1,kmax);
	yp=matrix(1,nvar,1,kmax);
	r=pow(2.0,0.5);

	for (i=1;i<=kmax;i++) {
		chi=pow(r,-k-9+i);
		A=1.0;
		B=chi*6.0*b*c*A;
		xp[i]=chi;

	yp[1][i]=4.0/(b*c)*(quad2d_1(de_edA,0.5*b,a+0.5*b,0.01)+quad2d_2(de_edA,a
+0.5*b,b,0.01)
```

```
	+quad2d_3(de_edA,0.5*b,a+0.5*b,0.01)+quad2d_4(de_edA,a+0.5*b,b,0.01));
//dWdA

	yp[2][i]=4.0/(b*c)*(quad2d_1(de_edB,0.5*b,a+0.5*b,0.01)+quad2d_2(de_edB,a
+0.5*b,b,0.01)

	+quad2d_3(de_edB,0.5*b,a+0.5*b,0.01)+quad2d_4(de_edB,a+0.5*b,b,0.01));
//dWdB
		yp[3][i]=b*c*yp[2][i]-yp[1][i]/6.0; //Sigma_m
		yp[4][i]=0.5*yp[1][i]; //Sigma_e
	}

	fp=fopen("data.txt","w");
	printf("dWdA:\n");
	fprintf(fp,"dWdA:\n");
	for (i=1;i<=kmax;i++) {
		printf("%f\n",yp[1][i]);
		fprintf(fp,"%f\n",yp[1][i]);
	}
	printf("dWdB:\n");
	fprintf(fp,"dWdB:\n");
	for (i=1;i<=kmax;i++) {
		printf("%f\n",yp[2][i]);
		fprintf(fp,"%f\n",yp[2][i]);
	}
	printf("Sigma_m:\n");
	fprintf(fp,"Sigma_m:\n");
	for (i=1;i<=kmax;i++) {
		printf("%f\n",yp[3][i]);
		fprintf(fp,"%f\n",yp[3][i]);
	}
	printf("Sigma_e:\n");
	fprintf(fp,"Sigma_e:\n");
	for (i=1;i<=kmax;i++) {
		printf("%f\n",yp[4][i]);
		fprintf(fp,"%f\n",yp[4][i]);
	}

	return 0;
}
```

VITA

Liang Zhang received his Bachelor of Science degree in mechanical engineering from University of Science and Technology of China in 2005. He entered the Mechanical Engineering program at Texas A&M University in August 2006 and received his Master of Science degree in May 2008 and his Doctor of Philosophy degree in December 2011. His research interests include plasticity, fracture mechanics, mechanics of composite materials, viscoelasticity, computational mechanics, and experimental mechanics.

Dr. Zhang may be reached at Texas A&M University, Department of Mechanical Engineering, 3123 TAMU, College Station, TX 77843-3123. His email is zhangliang115@hotmail.com.