Micromechanics Analysis Code With Generalized Method of Cells (MAC/GMC)
User Guide: Version 3.0

S.M. Arnold
Glenn Research Center, Cleveland, Ohio

B.A. Bednarcyk
Ohio Aerospace Institute, Cleveland, Ohio

T.E. Wilt and D. Trowbridge
The University of Akron, Akron, Ohio

March 1999
Acknowledgments

The authors would like to thank Mr. Robert Goldberg and Mrs. Cheryl Bowman for their diligent efforts and assistance in checking the accuracy and consistency of this manual by running numerous test cases. Also, we would like to thank Prof. J. Aboudi for many fruitful discussions and technical assistance.

Trade names or manufacturers' names are used in this report for identification only. This usage does not constitute an official endorsement, either expressed or implied, by the National Aeronautics and Space Administration.

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076
Price Code: A11

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100
Price Code: A11
MAC/GMC: Micromechanics Analysis Code

1.0 Introduction ............................................................................................................ 3

2.0 Current Capabilities ............................................................................................... 6

3.0 Background ........................................................................................................... 7
3.1 Micromechanics Models ..................................................................................... 7
3.2 Integration Algorithms ....................................................................................... 14
3.3 Available Constituent Constitutive Models ....................................................... 16
  3.3.1 Transversely Isotropic Elastic Model .................................................. 16
  3.3.2 Anisotropic Elastic Model .................................................................. 17
  3.3.3 Bodner-Partom Model ....................................................................... 18
  3.3.4 Modified Bodner-Partom Model .......................................................... 19
  3.3.5 Robinson Viscoelastic Model ............................................................. 20
  3.3.6 Generalized Viscoplastic Potential Structure (GVIPS) Model ............. 22
  3.3.7 Transversely Isotropic GVIPS Model (TGVIPS) ............................. 24

3.4 Laminate Theory ............................................................................................... 26

3.5 Fatigue Damage Analysis .................................................................................... 30
  3.5.1 Above Initial Fatigue Limit ............................................................... 31
  3.5.2 Below Initial Fatigue Limit ................................................................ 32

3.6 User Defined Inelastic Material Model ............................................................. 34

3.7 User Defined Functional Form For Material Properties ................................... 37

3.8 Interface Modeling ............................................................................................ 42

3.9 References ......................................................................................................... 45

4.0 Running MAC/GMC .......................................................................................... 46
  4.0.1 MAC/GMC - stand-alone: .................................................................. 46
  4.0.2 FEAMAC - Accessing MAC through ABAQUS: .................................. 46

4.1 Input and Output Files ....................................................................................... 47

4.2 Input Requirements (for both MAC/GMC and FEAMAC) ................................. 51
  4.2.1 Header Line: ..................................................................................... 53
  4.2.2 Output Print Level: ........................................................................... 53
  4.2.3 Load Type: ....................................................................................... 54
  4.2.4 Mechanical Load Control: ................................................................. 57
  4.2.5 Temperature Control: ...................................................................... 58
  4.2.6 Fatigue Damage Option: ................................................................... 60
  4.2.7 Yield and Damage Surface Generation Analysis ................................... 62
  4.2.8 Micromechanics Model Identification: .............................................. 63
  4.2.9 PATRAN Output: (Optional) ............................................................. 64
  4.2.10 Integrator Identification ................................................................. 66
  4.2.11 Thermal Conductivity Calculation: (Optional) .................................. 67
  4.2.12 Constituent Material Model Identification: ....................................... 68
  4.2.13 RVE Data: ...................................................................................... 70
  4.2.14 Interface Data: (Optional, for IDP=1, 2, 3, 6 or 11) ......................... 94
  4.2.15 Debond Data: (Optional) ................................................................. 95
  4.2.16 Plot Point Information: ................................................................... 99
  4.2.17 Curve Data: ................................................................................... 100

5.0 Conclusion/Future Modifications ........................................................................ 104
  5.1 Acknowledgment ............................................................................................. 104

6.0 EXAMPLE PROBLEMS ....................................................................................... 105
  6.1 Example A: Pure Mechanical Load .................................................................. 105
<table>
<thead>
<tr>
<th>Section</th>
<th>Example</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.2</td>
<td>Example B: Pure Thermal Problem</td>
<td>108</td>
</tr>
<tr>
<td>6.3</td>
<td>Example C: Thermomechanical Load Problem</td>
<td>111</td>
</tr>
<tr>
<td>6.4</td>
<td>Example D: Transverse Debonding</td>
<td>114</td>
</tr>
<tr>
<td>6.5</td>
<td>Example E: General Loading</td>
<td>120</td>
</tr>
<tr>
<td>6.6</td>
<td>Example F: Unidirectional Laminate Problem</td>
<td>123</td>
</tr>
<tr>
<td>6.7</td>
<td>Example G: A Cross-Ply Laminate Problem</td>
<td>126</td>
</tr>
<tr>
<td>6.8</td>
<td>Example H: Triple Periodic GMC; [0/90]</td>
<td>129</td>
</tr>
<tr>
<td>6.9</td>
<td>Example I: A User Defined RVE</td>
<td>132</td>
</tr>
<tr>
<td>6.10</td>
<td>Example J: A Biaxial Load</td>
<td>135</td>
</tr>
<tr>
<td>6.11</td>
<td>Example K: User Defined Inelastic Material Model</td>
<td>138</td>
</tr>
<tr>
<td>6.12</td>
<td>Example L: Fatigue Damage Analysis</td>
<td>156</td>
</tr>
<tr>
<td>6.13</td>
<td>Example M: Longitudinal Discrete Fiber Breakage</td>
<td>168</td>
</tr>
<tr>
<td>6.14</td>
<td>Example N: Plain Weave Composite</td>
<td>174</td>
</tr>
<tr>
<td>6.15</td>
<td>Example O: FEAMAC Analysis</td>
<td>181</td>
</tr>
<tr>
<td>6.16</td>
<td>Example P: Non-Symmetric Laminate</td>
<td>205</td>
</tr>
<tr>
<td>6.17</td>
<td>Example Q: Yield Surface Plot</td>
<td>209</td>
</tr>
<tr>
<td>6.18</td>
<td>Example R: Ellipsoidal Inclusions</td>
<td>212</td>
</tr>
<tr>
<td>6.19</td>
<td>Example S: PATRAN/MSC MACPOST Output</td>
<td>215</td>
</tr>
<tr>
<td>6.20</td>
<td>Example T: Effective Thermal Conductivity Calculation</td>
<td>219</td>
</tr>
</tbody>
</table>
1.0 Introduction

The ability to accurately predict the thermomechanical deformation response of advanced heterogeneous materials (e.g., multi-phase composites, cellular solids, etc.) plays a critical role in the development and practical implementation of these strategic materials. Analytical models that predict the effective behavior of composites are used not only by engineers performing structural analysis of large-scale composite components, but also by materials scientists in developing new material systems. For an analytical model to fulfill these two distinct functions it must be based on a micromechanics approach which utilizes physically based deformation and life constitutive models and generates the average (macro) response of a composite material given the properties of the individual constituents and their geometric arrangement. Only then can such a model be used by a materials scientist to investigate the effect(s) of different deformation mechanisms on the overall response of the composite in order to identify the appropriate constituents for a given application. Alternatively, if a micromechanical model is to be used in a large-scale structural analysis it must be 1) computationally efficient, 2) able to generate accurate displacement and stress fields at both the macro and the micro levels and 3) be compatible with the finite element method. Also, as advancements in processing and fabrication techniques make it possible to more accurately engineer (tailor) the architectures of these advanced composite systems, development of a computationally efficient micromechanics analysis tool capable of accurately predicting the effect of microstructural details on the internal and macroscopic behavior of composites becomes even more important. Such computational efficiency is absolutely required because 1) the large number of parameters which must be varied in the course of engineering (or designing) composite (heterogeneous) materials, and 2) the optimization of a material's microstructure requires the integration of the micromechanics model with optimization algorithms. From this perspective, analytical approaches which use closed-form expressions to describe the effect of a material's internal architecture on the overall material behavior are preferable to numerical methods, such as finite element or finite difference schemes.

A number of models presently exist that can fulfill certain aspects of the aforementioned tasks. However, there are very few working models that are both computationally efficient and sufficiently accurate both at the micro- and macro-levels. One such micromechanics model with the potential of fulfilling both tasks is Aboudi's method of cells [1] and its continuous reinforcement [2] and discontinuous reinforcement [3] generalizations. The comprehensive capabilities and efficiency of this method have been documented in references [4] and [5]. Consequently, the recently developed, computationally efficient and comprehensive micromechanics analysis code, MAC/GMC, will now be described: with a predictive capability resting entirely upon the fully analytical micromechanics model, herein referred to as the generalized method of cells, GMC, [2 and 3]. MAC/GMC is a versatile form of research software that "drives" the double or triple periodic micromechanics constitutive models based upon GMC. GMC is capable of pre-
dicting the response of both continuous and discontinuous multi-phase composites with arbitrary internal microstructures and reinforcement shapes. GMC is a continuum-based micromechanics model that provides closed-form expressions for the macroscopic composite response in terms of the properties, size, shape, distribution, and response of the individual constituents or phases that make up the material. GMC also utilizes physically based viscoplastic deformation and life models for each constituent. Furthermore, expressions relating the internal stress and strain fields in the individual constituents in terms of the macroscopically applied stresses and strains are available through strain or stress concentration factors. These expressions make possible the investigation of failure processes at the microscopic level at each step of an applied load history. Similarly, GMC provides the capability of studying the influence of bond strength at the fiber/matrix interface which has been shown to be an important damage mechanism in composites [6].

MAC/GMC enhances the basic capabilities of GMC by providing a modular framework wherein 1) various thermal, mechanical (stress or strain control) and thermomechanical load histories can be imposed, 2) different integration algorithms may be selected, 3) a variety of material constitutive models (both deformation and life) may be utilized and/or implemented and 4) a variety of fiber architectures (both unidirectional, laminate and woven) may be easily accessed through their corresponding representative volume elements contained within the supplied library of RVEs or input directly by the user, and 5) graphical post processing of the macro and/or micro field quantities is made available. Consequently, the availability of MAC/GMC now provides industry, academia and government engineers and materials scientists with a comprehensive, computationally efficient, user-friendly micromechanics analysis tool that can easily and accurately design/analyze multi-phase (composite) materials for a given application. MAC/GMC is also ideally suited for conducting sensitivity/parametric studies (i.e., “what-if” scenarios) in the design/analysis of advanced composite materials (e.g., MMC/IMCs, PMCs, and CMCs). Figure 1 illustrates the basic flow diagram for this modular framework code. Furthermore, since GMC is a continuum-based micromechanics model that provides closed form expressions for the macroscopic composite response in terms of the properties, size, shape, distribution, and response of the individual constituents or phases that make up the material, MAC/GMC can be interfaced directly with standard linear and nonlinear finite element analysis packages (through their respective user definable constitutive routines) for cost-effective large scale component design and analysis. Currently, such an interface exists only for HKS’s nonlinear finite element code, ABAQUS. For clarity purposes, when this finite element implementation of MAC/GMC is discussed within this manual it will be referred to as FEAMAC, for example see section 4.1.2.

The capabilities of this version of MAC/GMC are listed in section 2, whereas theoretical and background information on the basic capabilities previously itemized are given in section 3. Section 4 describes in detail the required input for using MAC/GMC and will be the most referenced section in the entire manual. Section 5 gives some insights into the future modifications planned for MAC/GMC and in section 6 examples A-S illustrate and describe the various features of MAC/GMC.
Figure 1: MAC Flowchart
2.0 Current Capabilities

In this section the current features/capabilities of MAC/GMC are itemized, with those new to this version being denoted by **bold type face**.

- **Load Types:**
  - Thermal, Mechanical, Thermomechanical; **Generalized Loading Option**

- **Aboudi GMC Models (Reformulated for computational speed):**
  - Double Periodicity Model for continuous reinforcement
  - Triple Periodicity Model for discontinuous/woven reinforcement

- **Library of Representative Volume Elements:**
  - Double Periodicity Model: **circular fiber representations**
  - Triple Periodicity Model: **ellipsoidal inclusion, open cell**

- **Input / Graphical Output**
  - Free Format Input Data
  - Up to 5 x-y data plot files generated for macro and micro (subcell) quantities
  - PATRAN for subcell geometry and color results evaluation, e.g. stress, strains, inelastic strains, $J_2$, etc., via **MACPOST**

- **Integration Options:**
  - Forward Euler
  - Predictor/Corrector

- **Constitutive Models (expanded internal material database)**
  - elastic: transversely isotropic and **anisotropic**
  - inelastic viscoplastic models:
    - Bodner-Partom
    - Robinson
    - GVIPS isotropic form
    - **TGVIPS transversely isotropic form**
  - user defined constitutive model via subroutines USRMAT (inelastic), USRFORMDE and USRCPEVAL (elastic); user defined functional dependence of material constants via subroutine USRFUN

- **Interface Modeling**
  - fiber/matrix interface layer (distinct interfacial material)
  - **fiber/matrix debonding conditions** (interfacial displacement discontinuity)

- **Fatigue Damage Analysis**

- **Yield Surface Calculations**

- **Symmetric and Nonsymmetric Laminate Analysis**

- **FEAMAC** - allows easy finite element access to MAC/GMC capabilities
3.0 Background

As stated in the introduction, MAC/GMC's predictive capabilities rest upon the fully analytical micromechanics model known as GMC which is capable of predicting the inelastic response of both continuous (double periodicity) and discontinuous/woven (triple periodicity) multi-phase composites with arbitrary internal microstructures and reinforcement shapes. Included in this section is a brief overview of the theoretical foundations of GMC; a more complete discussion is given in references [1-3]. Also, included in this section are discussions of 1) the available integration algorithms, 2) various constitutive models, 3) theory underlying the laminate, 4) fatigue damage, and 5) interfacial modeling approaches incorporated into MAC/GMC. Besides describing the available constitutive models, explanations of the user defined inelastic and elastic material models are included. Finally, the available architectures (Repeating Volume Elements, or RVEs) and required input are then described in section 4.0.

3.1 Micromechanics Models

In the original formulation of the method of cells, a continuously (or discontinuously) reinforced, unidirectional fibrous composite was modeled as a rectangular, double-periodic (or triple-periodic) array of fibers embedded in a matrix phase. The periodic character of the assemblage allowed identification of a repeating unit cell that can be used as a building block to construct the entire composite. The properties of the repeating cell were thus representative of the properties of the entire assemblage. The unit cell consisted of a single fiber subcell surrounded by three matrix subcells for continuous, and seven for discontinuous, composites, hence the name method of cells. The rectangular geometry of the repeating unit cell allowed one to obtain an approximate solution for the stresses and strains in the individual subcells given some macroscopically homogeneous state of strain or stress applied to the composite. The approximate solution to the posed boundary value problem was, in turn, used to determine macroscopic (average) properties, traditionally referred to as effective properties, of the composite and the macro (global) stress-strain response in the inelastic region.

In the generalized method of cells for continuous (or discontinuous) fibrous composites, the repeating unit cell can consist of an arbitrary number of phases, see Fig. 2. Hence, the generalized method of cells is capable of modeling a multi-phase composite. This generalization extends the modeling capability of the original method of cells to include the following: 1) inelastic thermomechanical response of multi-phase metal matrix composite, 2) modeling of various fiber architectures (including both shape and packing arrangements), 3) modeling of porosities and damage, and 4) modeling of interfacial regions around inclusions including interfacial degradation.

The basic homogenization approach taken in the micromechanical analysis consists essentially of four steps. First, the RVE, of the periodic composite is iden-
tified. Second, the macroscopic or average stress and strain states in terms of the individual microscopic (subcell) stress and strain states is defined. Third, the continuity of tractions and displacements are imposed at the boundaries between the constituents. These three steps, in conjunction with micro-equilibrium, establish the relationship between micro (subcell) total, thermal and inelastic strains and macro (composite) strains via the relevant concentration tensors. In the fourth and final step, the overall macro constitutive equations of the composite are determined. These four steps form the basis of the micro-to-macromechanics analysis which describes the behavior of heterogeneous media. The resulting micromechanical analysis establishes the overall (macro) behavior of the multi-phase composite which is expressed as a constitutive relation between the average stress, strain, thermal, and inelastic strains, in conjunction with the effective elastic stiffness matrix.

That is,

$$\bar{\sigma} = B^* (\bar{\varepsilon} - \bar{\varepsilon}' - \bar{\varepsilon}^T)$$  \hspace{1cm} (EQ 1)

where, for the most general case of discontinuous reinforcement with \(N_\alpha\) by \(N_\beta\) by \(N_\gamma\) number of subcells, the effective elastic stiffness matrix, \(B^*\), of the composite is given by,

$$B^* = \frac{1}{dh} \sum_{\alpha=1}^{N_\alpha} \sum_{\beta=1}^{N_\beta} \sum_{\gamma=1}^{N_\gamma} d_\alpha h_\beta l_\gamma C^{(\alpha\beta\gamma)} A^{(\alpha\beta\gamma)}$$  \hspace{1cm} (EQ 2)

the composite inelastic strain vector is defined as,

$$\bar{\varepsilon}' = \frac{-B^*}{dh} \sum_{\alpha=1}^{N_\alpha} \sum_{\beta=1}^{N_\beta} \sum_{\gamma=1}^{N_\gamma} d_\alpha h_\beta l_\gamma C^{(\alpha\beta\gamma)} (D^{(\alpha\beta\gamma)} \bar{\varepsilon}_s - \bar{\varepsilon}^{(\alpha\beta\gamma)}_s)$$  \hspace{1cm} (EQ 3)

the average thermal strain vector as,

$$\bar{\varepsilon}^T = \frac{-B^*}{dh} \sum_{\alpha=1}^{N_\alpha} \sum_{\beta=1}^{N_\beta} \sum_{\gamma=1}^{N_\gamma} d_\alpha h_\beta l_\gamma C^{(\alpha\beta\gamma)} (D^{(\alpha\beta\gamma)} \bar{\varepsilon}_s - \bar{\varepsilon}^{(\alpha\beta\gamma)}_s)$$  \hspace{1cm} (EQ 4)

and \(\bar{\varepsilon}\) is the uniform applied macro (composite) strain. For the case of continuous reinforcements with \(N_\beta\) by \(N_\gamma\) number of subcells, Eqs. 2 - 4 reduce to the following:

$$B^* = \frac{1}{h} \sum_{\beta=1}^{N_\beta} \sum_{\gamma=1}^{N_\gamma} h_\beta l_\gamma C^{(\beta\gamma)} A^{(\beta\gamma)}$$  \hspace{1cm} (EQ 5)
Figure 2: Subcell Dimension Nomenclature
\[ \mathbf{e}^I = -\frac{B^*}{h_l} \sum_{\beta = 1}^{N_\beta} \sum_{\gamma = 1}^{N_\gamma} h_{\beta \gamma} C^{(\beta \gamma)} (\mathbf{D}^{(\beta \gamma)} (\mathbf{e}^I_{s} - \mathbf{e}^I_{\beta \gamma})) \]  

(EQ 6)

\[ \mathbf{e}^T = -\frac{B^*}{h_l} \sum_{\beta = 1}^{N_\beta} \sum_{\gamma = 1}^{N_\gamma} h_{\beta \gamma} C^{(\beta \gamma)} (\mathbf{D}^{(\beta \gamma)} (\mathbf{e}^T_{s} - \mathbf{e}^T_{\beta \gamma})) \]  

(EQ 7)

In the above equations matrix notation is employed; where, for example, the average stress, \( \mathbf{\bar{\sigma}} \), average strain, \( \mathbf{\bar{\varepsilon}} \), and inelastic subcell strain, \( \mathbf{\varepsilon}^I \), vectors represent,

\[ \mathbf{\bar{\sigma}} = \{ \bar{\sigma}_{11}, \bar{\sigma}_{22}, \bar{\sigma}_{33}, \bar{\sigma}_{13}, \bar{\sigma}_{13} \} \]  

(EQ 8)

\[ \mathbf{\bar{\varepsilon}} = \{ \bar{\varepsilon}_{11}, \bar{\varepsilon}_{22}, \bar{\varepsilon}_{33}, 2\bar{\varepsilon}_{13}, 2\bar{\varepsilon}_{13}, \bar{\varepsilon}_{12} \} \]  

(EQ 9)

\[ \mathbf{\varepsilon}^I_s = \{ \varepsilon^{(111)}_s, \ldots, \varepsilon^{(N_\alpha N_\beta N_\gamma)}_s \} \]  

(EQ 10)

where the six components of the vector \( \mathbf{\varepsilon}^{I(\alpha \beta \gamma)} \) are arranged as in EQ. 9. Similar definitions for \( \mathbf{\varepsilon}^T \) and \( \mathbf{\varepsilon}^{T(\alpha \beta \gamma)} \) also exist. Note that the key ingredient in the construction of this macro constitutive law is the derivation of the appropriate concentration matrices, \( \mathbf{A}^{(\alpha \beta \gamma)} \) and \( \mathbf{D}^{(\alpha \beta \gamma)} \) having the dimensions 6 by 6 and 6 by \( N_\alpha N_\beta N_\gamma \) respectively, at the micro (subcell) level. The definitions of \( \mathbf{A} \) and \( \mathbf{D} \), although not given here, may be found in references [2] and [3]. Finally, the matrix \( \mathbf{C}^{(\alpha \beta \gamma)} \) represents the elastic stiffness tensor of each subcell (\( \alpha \beta \gamma \)) and \( d_\alpha, h_{\beta \gamma}, l_\gamma \) the respective subcell dimensions (see Fig. 2) wherein,

\[ d = \sum_{\alpha = 1}^{N_\alpha} d_\alpha \quad h = \sum_{\beta = 1}^{N_\beta} h_{\beta \gamma} \quad l = \sum_{\gamma = 1}^{N_\gamma} l_\gamma \]

Similarly, given the concentration matrices \( \mathbf{A}^{(\alpha \beta \gamma)} \) and \( \mathbf{D}^{(\alpha \beta \gamma)} \), expressions for the average strain in each subcell can be constructed, i.e.,

\[ \varepsilon^{(\alpha \beta \gamma)} = \mathbf{A}^{(\alpha \beta \gamma)} \mathbf{\bar{\varepsilon}} + \mathbf{D}^{(\alpha \beta \gamma)} (\varepsilon^I_s + \varepsilon^T_s) \]  

(EQ 11)

as well as average stress

\[ \mathbf{\sigma}^{(\alpha \beta \gamma)} = \mathbf{C}^{(\alpha \beta \gamma)} [\mathbf{A}^{(\alpha \beta \gamma)} \mathbf{\bar{\varepsilon}} + \mathbf{D}^{(\alpha \beta \gamma)} (\varepsilon^I_s + \varepsilon^T_s) - (\varepsilon^I_s + \varepsilon^T_s) - (\varepsilon^I_s + \varepsilon^T_s)] \]  

(EQ 12)

The analytic constitutive law, see EQ. 1, may be readily applied to investigate the behavior of various types of composites, given knowledge of the behavior of the individual phases. Note that within MAC/GMC an incremental tangent solution scheme is utilized, therefore the appropriate rate forms of EQS. 1-12 are employed.
Numerous advantages can be stated regarding the current macro/micro constitutive laws as compared to the other numerical micromechanical approaches in the literature, e.g. the finite element unit cell approach. One advantage is that any type of simple or combined loading (multiaxial state of stress) can be applied irrespective of whether symmetry exists or not, as well as without resorting to different boundary condition application strategies as in the case of the finite element unit cell procedure. Another advantage concerns the availability of an analytical expression representing the macro elastic-thermo-inelastic constitutive law, thus ensuring a reduction in memory requirements when implementing this formulation into a structural finite element analysis code. Furthermore, this formulation has been shown to predict accurate macro behavior given only a few subcells, within the repeating cell (see references [2], and [4]). Whereas, if one employs the finite element unit cell procedure, a significant number of finite elements are required within a given repeating unit cell to obtain the same level of global accuracy as with the present formulation. Consequently, it is possible to utilize this formulation to efficiently analyze metal matrix composite structures subjected to complex thermomechanical load histories. This is particularly important when analyzing realistic structural components, since different loading conditions exist throughout the structure, thus necessitating the application of the macromechanical equations repeatedly at these locations.

The equations of GMC-3D (and consequently through appropriate specialization GMC-2D) have recently been reformulated [7] in a way that significantly increases the computational efficiency of the model. This new reformulation has now been implemented into MAC/GMC and is the default computational mode. By nature of the traction continuity conditions within the original generalized method, all six stress components are not unique in every subcell. Normal stress components are constant in certain rows of subcells, while shear stress components are constant in certain layers of subcells. The unique subcell stress components are denoted as,

\[ T^{(\beta\gamma)}_{11}, T^{(\alpha\gamma)}_{22}, T^{(\alpha\beta)}_{33}, T^{(\alpha)}_{23}, T^{(\beta)}_{13}, T^{(\gamma)}_{12}. \]  

Consequently, a more efficient formulation of GMC can be obtained by applying traction continuity directly (i.e. recognizing that traction continuity conditions require no more and no less than the aforementioned reduction in subcell stress components) and using subcell stresses rather than strains as the basic unknown quantities. Accordingly, the continuity of displacement conditions are formed in terms of subcell stresses (through the use of the subcell constitutive and kinematic equations), and the mixed concentration equations for the unit cell are constructed,

\[ T = G^{(\alpha\beta\gamma)}_e \xi + G^{JT(\alpha\beta\gamma)}(\xi_s + \xi_s^T). \]  

(EQ 14)
Here, $\mathbf{T}$ is the vector of all subcell stress components (listed in EQ 13), $\mathbf{G}^{(\alpha \beta \gamma)}$ is the subcell mixed concentration matrix, and $\mathbf{G}^{IT(\alpha \beta \gamma)}$ is the subcell inelastic-thermal mixed concentration matrix. The term *mixed* is used here because EQ 14 relates local (subcell) stresses to global strains. Clearly this equation contrasts with its original formulation counterpart, EQ 11, which relates local strains to global strains and is thus the unit cell strain concentration equation. As one could obtain EQ 1 from EQ. 12 in the original formulation, similar expressions for the global quantities ($\mathbf{B}^{*}$, $\mathbf{e}^{I}$ and $\mathbf{\varepsilon}^{T}$) can be determined easily from EQ 14 (see [7] for details).

The increased efficiency of the reformulation of GMC emerges mainly due to the increased efficiency of forming EQ 14 versus forming EQ 11. The formation of EQ 14 requires solution (of linear equations) for the unknown independent subcell stress components listed in EQ 13, numbering $N_{\beta}N_{\gamma} + N_{\alpha}N_{\gamma} + N_{\alpha}N_{\beta} + N_{\alpha} + N_{\beta} + N_{\gamma}$. The formation of EQ 11 requires solution for 6 unknown strain components for each subcell, or a total of $6N_{\alpha}N_{\beta}N_{\gamma}$ unknowns. Solution of linear equations, in essence, amounts to the inversion of a matrix which has the rank of the number of unknown quantities. Since the computational effort associated with matrix inversion increases approximately as the cube of the matrix rank, reducing the number of unknown quantities has a major impact on computational efficiency. The reduction in unknowns due to the reformulation is shown in Figure 3 and can be quite significant. For example, a 10x10 x10 unit cell originally required solution for 6000 unknowns. In the reformulated version of GMC-3D the number of unknowns is reduced to 330. Consequently, the corresponding execution times are reduced as well. Table I gives sample CPU times for identical cases executed using the original and reformulated versions of GMC with continuous reinforcement. Clearly, as the number of subcells in the repeating unit cell increases, the increase in efficiency attributable to the reformulation becomes astronomical. It is important to note that:

1. The reformulation of GMC gives identical results as the original formulation of GMC in every case.
2. The significant increase in the computational efficiency of GMC due to the reformulation allows:
   a. Analysis of simple unit cells (i.e. few subcells) in a fraction of the time.
   b. Analysis of refined unit cells (i.e. many subcells) which was previously impossible (due to excessive execution times and memory requirements) can now be performed in times comparable to those previously required for analysis of simple unit cell, see Table I.
Figure 3. Number of subcells vs. number of unknown variables for the original and reformulated versions of GMC-3D for $N_{\alpha} = N_{\beta} = N_{\gamma}$

Table I: CPU Times (seconds) for Heat-Up of Unidirectional SiC/TiAl

<table>
<thead>
<tr>
<th>GMC Version</th>
<th>Subcell Discretization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2x2  4x4  6x6  8x8  10x10  12x12  20x20  100x100</td>
</tr>
<tr>
<td>original</td>
<td>0.87  19   182  508  8,679  43,781  -      -</td>
</tr>
<tr>
<td>reformulated</td>
<td>0.18  0.25  0.5   0.9   1.5   2.3   8.3    796</td>
</tr>
<tr>
<td>speed-up ratio</td>
<td>4.8   76   364   564  5,786  19,035   -      -</td>
</tr>
</tbody>
</table>

Note: These execution times are for a specialized version of GMC-2D and not MAC/GMC, therefore one should not expect to see the same execution times when running MAC/GMC due to increased overhead. However, speed-up ratios should be comparable.
3.2 Integration Algorithms

There are two integration algorithms currently available within MAC/GMC. The first is the standard, explicit Forward Euler algorithm, which can be expressed as,

\[ w_{i+1} = w_i + \Delta t f(t_i, w_i) \]

where

\[ w_i = \dot{y}(t_i) \]
\[ w_{i+1} = \dot{y}(t_{i+1}) \]

and \( f(t_i, w_i) \) is the rate of change with respect to time of the vector, \( \dot{y}(t_i) \), i.e.

\[ f(t_i, w_i) = \ddot{y} = \frac{d\dot{y}(t_i)}{dt} \]

The second is a predictor/corrector algorithm with a self-adaptive time step which uses:

1) a 4th order Runge-Kutta starter:

\[ w_i = w_{i-1} + \frac{(K_1 + 2K_2 + 2K_3 + K_4)}{6} \]

where

\[ K_1 = \Delta t f(t_{i-1}, w_{i-1}) \]
\[ K_2 = \Delta t f(t_{i-1} + \frac{\Delta t}{2}, w_{i-1} + \frac{K_1}{2}) \]
\[ K_3 = \Delta t f(t_{i-1} + \frac{\Delta t}{2}, w_{i-1} + \frac{K_2}{2}) \]
\[ K_4 = \Delta t f(t_{i-1} + \Delta t, w_{i-1} + K_3) \]

2) with an Adams-Bashforth four-step predictor:

\[ w_0 = \alpha_1 \quad w_1 = \alpha_2 \quad w_2 = \alpha_3 \quad w_3 = \alpha_4 \]
\[ w_i^p = w_i + \frac{\Delta t}{24} [55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) + 37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})] \]
and 3) an Adams-Moulton four step corrector:

\[ W_{i+1}^c = W_i + \frac{\Delta t}{24} \left[ 9 f(t_{i+1}, W_{i+1}^p) + 19 f(t_i, W_i) - 5 f(t_{i-1}, W_{i-1}) ight. \\
\left. + f(t_{i-2}, W_{i-2}) \right] \]

where the \( \alpha \)'s come from the 4th order Runge-Kutta starter. Further details may be found in [8].

\[ \text{Note:} \quad \text{It has been found, based on experience, that for relatively rapid monotonic or cyclic loadings it may be more efficient to use the Forward Euler integrator since the predictor/corrector requires 5 evaluations per step, as shown above. However, in the case of creep, relaxation or slow monotonic or cyclic loading histories, significant increases in solution speeds can be obtained using the predictor/corrector algorithm.} \]

Finally, within MAC/GMC the vector \( W \) as used above contains the following macro quantities:

<table>
<thead>
<tr>
<th>position</th>
<th>contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 6</td>
<td>Macro Total Strain</td>
</tr>
<tr>
<td>7 - 12</td>
<td>Macro Stress</td>
</tr>
<tr>
<td>13 - 18</td>
<td>Macro Inelastic Strain</td>
</tr>
<tr>
<td>19 - 30</td>
<td>(currently empty space for 2 6x1 vectors)</td>
</tr>
<tr>
<td>31 - 36</td>
<td>Macro Thermal Strain</td>
</tr>
<tr>
<td>37</td>
<td>Current Temperature</td>
</tr>
<tr>
<td>38-46</td>
<td>Future Use</td>
</tr>
</tbody>
</table>

and the quantities associated with each subcell are stored sequentially in \( W \), such that

<table>
<thead>
<tr>
<th>position</th>
<th>contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>47 - 52</td>
<td>Micro Total Strain</td>
</tr>
<tr>
<td>53 - 58</td>
<td>Micro Stress</td>
</tr>
<tr>
<td>59 - 64</td>
<td>Micro Inelastic Strain</td>
</tr>
<tr>
<td>65 - 76</td>
<td>Micro Internal State Variables</td>
</tr>
<tr>
<td>77 - 82</td>
<td>Micro Thermal Strain</td>
</tr>
<tr>
<td>83 - 88</td>
<td>Debonding Parameters</td>
</tr>
</tbody>
</table>

The above 42 positions (i.e., positions 47-88) are repeated for the total number of subcells (\( N \)) thus bringing the total length of the \( W \) vector to 46+42N. It follows that a second vector of similar length that contains the corresponding macro and micro rates, e.g., \( f(t_i, W_i) \), is also utilized.
3.3 Available Constituent Constitutive Models

Currently **MAC/GMC** provides two elastic and five inelastic constitutive models. These models have been selected purely based upon the availability of material parameters for the materials of interest. However, **MAC/GMC** is designed in a modular fashion thus allowing the implementation of additional constitutive models through a user defined subroutine. Two of the five available inelastic models are capable of representing transversely isotropic material behavior, thus allowing one to investigate the reinforcement of an anisotropic matrix allowing idealization of a heterogeneous material via a pseudo-homogenous anisotropic material (e.g. fiber tow). In all five inelastic models a purely elastic response is possible by modifying a single material parameter for each model as noted below.

### 3.3.1 Transversely Isotropic Elastic Model


The following transversely isotropic model is provided for those materials that have an elastic only response and whose strong direction is aligned with the 1 axis shown in Fig. 6, such as the fiber constituent in a composite.

\[
\begin{bmatrix}
\sigma_{11} \\
\sigma_{22} \\
\sigma_{33} \\
\sigma_{23} \\
\sigma_{13} \\
\sigma_{12}
\end{bmatrix} =
\begin{bmatrix}
C_{11} & C_{12} & 0 & 0 & 0 \\
C_{12} & C_{22} & C_{23} & 0 & 0 \\
C_{12} & C_{23} & C_{22} & 0 & 0 \\
0 & 0 & 0 & C_{44} & 0 \\
0 & 0 & 0 & 0 & C_{44} \\
0 & 0 & 0 & 0 & 0 & C_{66}
\end{bmatrix}
\begin{bmatrix}
\varepsilon_{11} \\
\varepsilon_{22} \\
\varepsilon_{33} \\
\gamma_{23} \\
\gamma_{13} \\
\gamma_{12}
\end{bmatrix} -
\begin{bmatrix}
\alpha_L \Delta T \\
\alpha_T \Delta T \\
\alpha_T \Delta T \\
0 \\
0 \\
0
\end{bmatrix}
\]

where components \( C_{ij} \) can be expressed in terms of five independent constants,

\[ E_A, E_T, v_A, v_T, G_A \]
thus,

\[ C_{11} = E_A + 4\kappa v_A^2 \]
\[ C_{12} = 2\kappa v_A \]
\[ C_{22} = \kappa + \frac{0.5E_T}{(1 + v_T)} \]
\[ C_{23} = \kappa - \frac{0.5E_A}{(1 + v_T)} \]
\[ C_{44} = G_A \]
\[ C_{66} = \frac{(C_{22} - C_{23})}{2} \]

with

\[ \kappa = 0.25E_A/[0.5(1 - v_T)(E_A/E_T) - v_A^2] \]

3.3.2 Anisotropic Elastic Model


An alternative transversely isotropic model has also been provided for those materials that have an elastic only response, but whose strong direction may not be aligned with the 1 axis shown in Fig. 6 (i.e., the plane of isotropy is allowed to rotate), for example in the case of woven composites, see Example N. This elastic model is consistent with the anisotropic inelastic models described subsequently and is defined within the global coordinate system via a direction cosine vector, \( d_i \).

\[ \sigma = [C]\epsilon - \epsilon^{th} \]

where the stiffness matrix, \( C \), is a function of the five independent material parameters \( E_A, E_T, v_A, v_T, G_A \) and the direction cosine vector, \( d_i \), and, in general is fully populated. Note the multiaxial thermal strain tensor is assumed to have the following form,

\[ \epsilon^{th}_{ij} = [(\alpha_L - \alpha_T)d_id_j + \delta_{ij}\alpha_T]\Delta T \]

Further details can be found in the above reference.
3.3.3 Bodner-Partom Model


This model represents a Bodner-Partom viscoplastic material with isotropic hardening, $Z$, and can be used for an initially isotropic metallic material.

The flow law is given as:

$$\dot{\epsilon}_{ij} = \Lambda s_{ij}$$

where

$$\Lambda = \sqrt{\frac{D_{2}^{PL}}{J_2}}$$

$$D_{2}^{PL} = D_{0}^{2} \exp \left(-\frac{A^{2/n}}{J_2}\right)$$

$$A^{2} = \frac{1}{3} z_{eff} \left(\frac{n+1}{n}\right)^{1/n}$$

$$J_2 = \frac{1}{2} S_{ij} s_{ij}$$

$$s_{ij} = \sigma_{ij} - \frac{1}{3} \sigma_{kk} \delta_{ij}$$

The evolution law for isotropic hardening is given as:

$$\dot{Z} = m(Z_{1} - Z_{eff}) \frac{\dot{W}_{PL}}{Z_{0}}$$

where $Z_{0}$, $Z_{1}$ and $m$ are inelastic constants and the plastic work rate, $\dot{W}_{PL}$, is given by:

$$\dot{W}_{PL} = \sigma_{ij} \dot{\epsilon}_{ij}$$

$$Z_{eff} = Z_{0} + q \int_{0}^{t} \dot{Z}(\tau) d\tau + (1-q) \sum_{i,j=1}^{3} r_{ij} \int_{0}^{t} \dot{Z}(\tau) r_{ij}(\tau) d\tau$$

$$r_{ij}(t) = \sigma_{ij}(t) \left[\sigma_{kl}(t) \sigma_{kl}(t)\right]^{1/2}$$

An elastic only response may be obtained by setting the material parameter $D_{0}$ to zero.
3.3.4 Modified Bodner-Partom Model


This model represents a nonisothermal Bodner-Partom viscoplastic material with isotropic, $Z^I$, and directional hardening, $Z^D$, and can be used for an initially isotropic metallic material.

The flow law is given as:

$$
\dot{\varepsilon}_{ij}^I = \Lambda s_{ij}
$$

where

$$
\Lambda = D_0 \exp \left[ -0.5 \left( \frac{Z^2}{3J_2} \right) \right] \frac{1}{\sqrt{J_2}}
$$

$$
Z = Z^I + Z^D
$$

$$
J_2 = \frac{1}{2} S_{ij} S_{ij}
$$

$$
S_{ij} = \sigma_{ij} - \frac{1}{3} \sigma_{kk} \delta_{ij}
$$

The evolution law for the isotropic hardening, $Z^I$, is given as:

$$
\dot{Z}^I = m_1 (Z_1 - Z^I) W_{PL}^P - A_1 Z_1 \left( \frac{Z^I - Z_2}{Z_1} \right)^{r_2} + \left( \frac{Z^I - Z_2}{Z_1 - Z_2} \right) \frac{\partial Z_1}{\partial T} + \left( \frac{Z_2 - Z^I}{Z_1 - Z_2} \right) \frac{\partial Z_2}{\partial T} T
$$

where $Z_0$, $Z_1$, $Z_2$, $A_1$, $r_1$, and $m_1$ are the material parameters associated with isotropic hardening. The magnitude of the directional hardening is defined as the scalar product of a state variable, $\beta_{ij}$, and unit stress vector, $u_{ij}$, as given below.

$$
Z^D = \beta_{ij} u_{ij}
$$

$$
\dot{\beta}_{ij} = m_2 (Z_3 u_{ij} - \beta_{ij}) W_{PL}^P - A_1 Z_1 \left( \frac{\sqrt{\beta_{ij} \beta_{ij}}}{Z_1} \right)^{r_2} + \left( \frac{\beta_{ij}}{Z_3} \frac{\partial Z_3}{\partial T} \right) T
$$
\[ u_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{kl} \sigma_{kl}}} \]
\[ v_{ij} = \frac{\beta_{ij}}{\sqrt{\beta_{kl} \beta_{kl}}} \]

where, the plastic work rate is defined as; \[ W_{ij}^{PL} = \sigma_{ij} \varepsilon_{ij} \] and with, \( Z_3, A_2, r_2 \) and \( m_2 \) being the material parameters defining the directional evolution. An elastic only response may be obtained by setting the material parameter \( D_0 \) to zero.

### 3.3.5 Robinson Viscoplastic Model


This model represents a **transversely isotropic material** wherein the vector of direction cosines, \( d_i \), defines the preferred material direction. In this model the strength of anisotropy is specified by the parameters \( \omega \) and \( \eta \); where \( \omega \) is the ratio of the normal longitudinal and transverse yield stress and \( \eta \) is the ratio of longitudinal and transverse threshold shear stress.

**Flow Law:**

\[ \dot{\varepsilon}_{ij} = \frac{\langle F^m \rangle}{2\mu} \Gamma_{ij} \]

**Evolution Law:**

\[ a_{ij} = \frac{H}{G^\beta} \xi \dot{\varepsilon}_{ij} - R G^m - \beta \Pi_{ij} \]

where

\[ \Gamma_{ij} = \Sigma_{ij} - \xi (D_{kj} \Sigma_{kj} + D_{jk} \Sigma_{ki} - 2I_0 D_{ij}) - \frac{1}{2} \xi I_0 (3D_{ij} - \delta_{ij}) \]

\[ \Pi_{ij} = a_{ij} - \xi (D_{kj} a_{kj} + D_{jk} a_{ki} - 2I_0 D_{ij}) - \frac{1}{2} \xi I_0 (3D_{ij} - \delta_{ij}) \]
and

\[ F = \frac{1}{\kappa_T^2} \left[ I_1 + \frac{1}{\eta^2} I_2 + \frac{9}{4(4\omega^2 - 1)} I_3 \right] - 1 \]

\[ \hat{G} = \frac{1}{\kappa_T^2} \left[ I_1 + \frac{1}{\eta^2} I_2 + \frac{9}{4(4\omega^2 - 1)} I_3 \right] \]

\[ G = \langle \hat{G} - \hat{G}_0 \rangle H v [S_{ij} \pi_{ij}] + \hat{G}_0 \]

\[ I_1 = J_2 - I - \frac{1}{4} I_3 \quad I_2 = I - I_3 \quad I_3 = I_0^2 \]

\[ J_2 = \frac{1}{2} \Sigma_{ij} \Sigma_{ji} \quad I = D_{ij} \Sigma_{ji} \quad D_{ij} = d_i d_j \quad \Sigma_{ij} = S_{ij} - a_{ij} \]

\[ \xi = \frac{\eta^2 - 1}{\eta^2} \quad \zeta = \frac{4(\omega^2 - 1)}{4\omega - 1} \]

with \( \kappa_T, \mu, n, H, \beta, R, m, G_0, \eta \) and \( \omega \) representing the associated required inelastic material parameters and \( \langle > \) denoting a Macauley bracket, as defined below.

\[ \langle x \rangle = \begin{cases} 0 & x < 0 \\ x & x \geq 0 \end{cases} \]

The invariants \( \hat{I}_1, \hat{I}_2, \hat{I}_3 \) are the same \( I_1, I_2, I_3 \) as those given above but with \( \Sigma_{ij} \) replaced by \( a_{ij} \). Special cases involving an isotropic material and/or elastic only response can be obtained by defining \( \omega = \eta = 1 \) and/or by setting \( \kappa_T \) to an extremely large number.
3.3.6 Generalized Viscoplastic Potential Structure (GVIPS) Model


This model is a fully associative, multiaxial, nonlinear kinematic hardening viscoplastic model for use with initially isotropic metallic materials. A unique aspect of this model is the inclusion of non-linear hardening through the use of a compliance operator \( Q_{ijkl} \) in the evolution law for the back stress. This non-linear tensorial operator is significant in that it allows both the flow and evolutionary laws to be fully associative (and therefore easily integrated) and greatly influences the multiaxial response under non-proportional loading paths.

Flow Law:

\[
\dot{\varepsilon}_{ij} = \frac{3}{2} \left\| \dot{\varepsilon}_{ij} \right\| \frac{\Sigma_{ij}}{\sqrt{J_2}} \quad \text{if} \quad F \geq 0
\]

where

\[
\left\| \dot{\varepsilon}_{ij} \right\| = \sqrt{\frac{2}{3} \dot{\varepsilon}_{ij} \dot{\varepsilon}_{ij}} = \frac{\mu F^n}{\kappa}
\]

Internal constitutive rate equation:

\[
\dot{a}_{ij} = L_{ijrs} \dot{A}_{rs}
\]

Evolution Law:

\[
\dot{A}_{rs} = \dot{\varepsilon}_{rs} - \frac{3}{2} \frac{\beta \kappa}{\kappa_o} \left\| \dot{\varepsilon}_{ij} \right\| \frac{a_{rs}}{\|G\|} H[Y] - \frac{3 R_{\alpha} B_0 G^q}{\kappa_o^2} a_{rs} \quad \text{if} \quad a_{ij} \Sigma_{ij} \geq 0
\]

\[
\dot{A}_{rs} = Q_{rstlm} E_{lmnp} \left( \dot{\varepsilon}_{np} - \frac{3}{2} \frac{\beta \kappa}{\kappa_o} \left\| \dot{\varepsilon}_{ij} \right\| \frac{a_{np}}{\|G\|} H[Y] - \frac{3 R_{\alpha} B_0 G^q}{\kappa_o^2} a_{np} \right) \quad \text{if} \quad a_{ij} \Sigma_{ij} < 0
\]

where
\[
F = \left( \frac{J_2}{\kappa} - Y \right)
\]
\[
Y = \left( 1 - \beta \sqrt{G} \right)
\]
\[
G = \frac{I_2}{\kappa_o^2}
\]

\[
L_{ijrs} = [Q_{ijrs}]^{-1} = \frac{\kappa_o^2}{3B_0(1 + B_1 p G^{p-1})} \left( I_{ijrs} - \frac{3B_1 p(p-1)G^{p-2}}{\kappa_o^2(1 + B_1 p G^{p-1}(6p - 5))} a_{rs} a_{ij} \right)
\]

and

\[
I_2 = \frac{3}{2} a_{ij} a_{ij} \quad \Sigma_{ij} = S_{ij} - a_{ij}
\]
\[
J_2 = \frac{3}{2} \Sigma_{ij} \Sigma_{ij} \quad S_{ij} = \sigma_{ij} - \frac{1}{3} \sigma_{kk} \delta_{ij}
\]

with \( \kappa, \mu, n, \kappa_o, B_0, B_1, p, R_\alpha, q \) and \( \beta \) being the associated required inelastic material parameters. Typically, \( \kappa, \mu, B_0, R_\alpha \) and \( \beta \) are taken to be functions of temperature and \( \kappa_o = \kappa(T_{ref}) \) is the initial drag stress at the reference temperature. The special case of an elastic only response maybe obtained by setting \( \kappa \) to an extremely large value.
3.3.7 Transversely Isotropic GVIPS Model (TGVIPS)


This model is a fully associative, multiaxial, isothermal, nonlinear kinematic hardening viscoplastic model for use with initially transversely isotropic metallic materials. A unique aspect of this model is the inclusion of non-linear hardening through the use of a compliance operator $Q_{ijkl}$ in the evolution law for the back stress. This non-linear tensorial operator is significant in that it allows both the flow and evolutionary laws to be fully associative (and therefore easily integrated) and greatly influences the multiaxial response under non-proportional loading paths.

The flow law for the inelastic strain, $\dot{\varepsilon}$, is given by,

$$\dot{\varepsilon} = \frac{\langle F^n \rangle}{2\mu} \Gamma$$

and the evolution law for internal stress, $\alpha$, is given by,

$$\dot{\alpha} = \left[ Z_m + \frac{h'}{h(1 + 2\beta)} (\alpha \otimes \alpha) \right] \left( \frac{H}{G^\beta} \dot{\varepsilon} - R G^m \Pi \right)$$

where $h = H/G^\beta$, $h' = -\beta/G$ and $Z_m = M^{-1}$

$$F = \frac{1}{2\kappa_i^2} (\sigma - \alpha) : M : (\sigma - \alpha) - 1$$

$$G = \frac{1}{2\kappa_i^2} \alpha : M : \alpha$$

$$\Gamma = M : (\sigma - \alpha)$$

$$\Pi = M : \alpha$$

$$M = \mathbf{P} - \xi Q - \frac{1}{2} \zeta R$$
The anisotropy of the material is introduced through the $\mathbf{M}$ matrix, specifically the parameters, $\xi$, $\zeta$ which are defined as,

$$\xi = \frac{\eta^2 - 1}{\eta^2} \quad \zeta = \frac{4(\omega^2 - 1)}{4\omega^2 - 1}$$

$$\eta = \frac{K_l}{K_t} \quad \omega = \frac{Y_t}{Y_l}$$

In the above, $0 \leq \xi \leq 1$ and $0 \leq \zeta \leq 1$ are the material strength ratios, in which the constants $K_l$ ($K_t$) are the threshold strengths in longitudinal (transverse) shear, and $Y_l$ ($Y_t$) are the threshold strengths in longitudinal (transverse) tension. Note that for an isotropic material, $\omega = \eta = 1$.

In addition, the fourth-order tensors $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$ are defined as,

$$\mathbf{P} = \mathbf{I} - \frac{1}{3}(\delta \otimes \delta) \quad I_{ijkl} = \frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})$$

$$Q_{ijkl} = \frac{1}{2}(D_{ik}\delta_{jl} + D_{il}\delta_{jk} + D_{jk}\delta_{il} + D_{jl}\delta_{ik}) - 2D_{ij}D_{kl}$$

$$R_{ijkl} = 3D_{ij}D_{kl} - (D_{ij}\delta_{kl} + \delta_{ij}D_{kl}) + \frac{1}{3}(\delta_{ij}\delta_{kl})$$

In the above, the vector of direction cosines $d_i$ defines the orientation of the material fiber, which leads to the material directionality tensor $\mathbf{D}$, $(\mathbf{D} = d_i d_j)$. Also, $\delta$ is the Kronecker delta (second-order identity tensor) and $\mathbf{I}$ is the fourth-order identity tensor. Finally, the symbol $: \quad$ indicates double-contraction and $\otimes$ cross product.

**Note:** When calculating $Z_m$ for the three dimension case one needs to replace $\mathbf{P}$ with $\hat{\mathbf{P}} = \text{diag}[1,1,1,2,2,2]$ as $\mathbf{P}$ is singular for the three dimensional case.
3.4 Laminate Theory

MAC/GMC includes the capability to analyze general (symmetric and nonsymmetric) composite laminates [9], see Fig. 4. Mid-plane strains and resultant forces in the plane of the laminate may be applied. That is, the global laminate stress-strain relation that is solved within MAC/GMC is expressed as,

\[
\begin{bmatrix}
N_{XX} \\
N_{YY} \\
N_{XY}
\end{bmatrix} = \begin{bmatrix}
A_{11} & A_{12} & A_{13} \\
A_{21} & A_{22} & A_{23} \\
A_{31} & A_{32} & A_{33}
\end{bmatrix} \begin{bmatrix}
\varepsilon_{xx} \\
\varepsilon_{yy} \\
\gamma_{xy}
\end{bmatrix} + \begin{bmatrix}
B_{11} & B_{12} & B_{13} \\
B_{21} & B_{22} & B_{23} \\
B_{31} & B_{32} & B_{33}
\end{bmatrix} \begin{bmatrix}
\kappa_{xx} \\
\kappa_{yy} \\
\kappa_{xy}
\end{bmatrix} - \begin{bmatrix}
\bar{N}_{XX} \\
\bar{N}_{YY} \\
\bar{N}_{XY}
\end{bmatrix} - \begin{bmatrix}
\bar{N}_{XX} \\
\bar{N}_{YY} \\
\bar{N}_{XY}
\end{bmatrix}
\] (EQ 15)

or

\[
\begin{bmatrix}
\bar{M}_{XX} \\
\bar{M}_{YY} \\
\bar{M}_{XY}
\end{bmatrix} = \begin{bmatrix}
D_{11} & D_{12} & D_{13} \\
D_{21} & D_{22} & D_{23} \\
D_{31} & D_{32} & D_{33}
\end{bmatrix} \begin{bmatrix}
\kappa_{xx} \\
\kappa_{yy} \\
\kappa_{xy}
\end{bmatrix} - \begin{bmatrix}
\bar{M}_{XX} \\
\bar{M}_{YY} \\
\bar{M}_{XY}
\end{bmatrix} - \begin{bmatrix}
\bar{M}_{XX} \\
\bar{M}_{YY} \\
\bar{M}_{XY}
\end{bmatrix}
\] (EQ 16)

where \(N\), \(\bar{N}\), \(\bar{N}^T\), \(M\), \(\bar{M}\), \(\bar{M}^T\) are the global laminate total, inelastic and thermal force and moment resultants, respectively. The matrices \(A\), \(B\), and \(D\) are the global laminate extensional, coupling and bending stiffnesses, respectively, and \(\varepsilon\) and \(\kappa\) the global laminate mid-plane strain and mid-plane curvature, respectively.

In forming the laminate extensional stiffness \(A\) the generalized method of cells model, GMC, is utilized to calculate the individual lamina properties. In this regard, the individual laminate stiffness, in lamina coordinates, \(Q\) is given by,

\[
Q = \begin{bmatrix}
Q_{11} & Q_{12} & 0 \\
Q_{21} & Q_{22} & 0 \\
0 & 0 & Q_{66}
\end{bmatrix}
\] (EQ 18)

in which the components of \(Q\) are given as,
The $C_{ij}$ in the above are the effective macro properties for the unidirectional composite lamina obtained from GMC.

It follows, employing a Kirchhoff-Love hypothesis, that the lamina stress-strain relation in global (laminate) coordinates denoted by $x-y$ is given by the relation,

\[
\begin{bmatrix}
\sigma_x \\
\sigma_y \\
\sigma_{xy}
\end{bmatrix}_k = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\
Q_{12} & Q_{22} & Q_{23} \\
Q_{13} & Q_{23} & Q_{33}
\end{bmatrix}_k \begin{bmatrix}
\varepsilon_{xx} \\
\varepsilon_{yy} \\
\gamma_{xy}
\end{bmatrix} + \begin{bmatrix}
\kappa_{xx} \\
\kappa_{yy} \\
\kappa_{xy}
\end{bmatrix}_k - \begin{bmatrix}
\varepsilon_{xx}' \\
\varepsilon_{yy}' \\
\gamma_{xy}'
\end{bmatrix}_k - \begin{bmatrix}
\varepsilon_{xx}^T \\
\varepsilon_{yy}^T \\
\gamma_{xy}^T
\end{bmatrix}_k
\]

or

\[
\sigma_k = \overline{Q}_k (\varepsilon + \varepsilon_k^T - \varepsilon_k - \varepsilon_k^T)
\]

where

\[
\overline{Q}_k = R_k^{-1} Q_k R_k
\]

\[
R_k = \begin{bmatrix}
\cos^2 \theta & \sin^2 \theta & 2 \sin \theta \cos \theta \\
\sin^2 \theta & \cos^2 \theta & -2 \sin \theta \cos \theta \\
-\sin \theta \cos \theta & \sin \theta \cos \theta & \cos^2 \theta - \sin^2 \theta
\end{bmatrix}
\]

and $\theta$ is the orientation of the longitudinal lamina axis with respect to the global $x$-direction, see Fig. 4, thus $\overline{Q}_k$ is the transformed lamina stiffness, i.e. from local lamina to global laminate coordinates. In addition, $\sigma_k$ is the lamina stress vector in laminate coordinates. It then follows that the global laminate extensional stiffness $A$ is given by,

\[
A = \sum_{k=1}^{nly} \overline{Q}_k t_k
\]
in which \( n_{ly} \) is the total number of layers in the laminate and \( t_k \) is the thickness of the \( k^{th} \) lamina. The coupling and bending stiffnesses can be similarly developed and given by the following expressions:

\[
B = \frac{1}{2} \sum_{k=1}^{n_{ly}} \overline{Q}_k (z_k^2 - z_{k-1}^2)
\]

(EQ 25)

\[
D = \frac{1}{3} \sum_{k=1}^{n_{ly}} \overline{Q}_k (z_k^3 - z_{k-1}^3)
\]

(EQ 26)

where \( z_k \) is the distance (considering the sign) to the top of layer \( k \) from the mid-plane.

Returning to EQ. 17, the quantities \( \overline{N}^I \) and \( \overline{N}^T \) (the laminate inelastic and thermal force resultants, respectively) are calculated from the individual lamina contributions through the following relations,

\[
\overline{N}^I = \sum_{k=1}^{n_{ly}} \overline{Q}_k \int_{z_{k-1}}^{z_k} \varepsilon_k^I dz \]

\[
\overline{M}^I = \sum_{k=1}^{n_{ly}} \overline{Q}_k \int_{z_{k-1}}^{z_k} \varepsilon_k^I z dz
\]

(EQ 27)

and

\[
\overline{N}^T = \sum_{k=1}^{n_{ly}} \overline{Q}_k \varepsilon_k^T t_k \]

\[
\overline{M}^T = \frac{1}{2} \sum_{k=1}^{n_{ly}} \overline{Q}_k \varepsilon_k^T (z_k^2 - z_{k-1}^2)
\]

(EQ 28)

where the integrals in EQ. 27 are performed using second order gauss quadrature which requires two integration points per layer. Thus all field quantities are tracked at the two gauss quadrature points in each layer of the laminate in MAC/GMC.
Figure 4: Laminate Coordinate System
3.5 Fatigue Damage Analysis

The fatigue damage calculations utilize a recently developed multiaxial, isothermal, continuum damage mechanics model for the fatigue of unidirectional metal matrix composites [10]. The model is phenomenological, stress based, and assumes a single scalar internal damage variable, \( D \). Note that for an initially anisotropic material, the evolution of the damage, although a scalar, is directionally dependent. As will be shown, this directional dependence is accounted for in the terms, \( \hat{F}_m \), \( \Phi_{fl} \), and \( \Phi_u \). The present multiaxial, isothermal, continuum damage model for \textbf{initially transversely isotropic materials} (e.g., unidirectional metal matrix composites) may be expressed as, [10]

\[
D_k = \int_{D_{k-1}}^N \left[ (1 - (1 - D))^{\beta + 1} \right]^{\alpha} \left[ \frac{\hat{F}_m(t)}{1 - D} \right]^\beta dN
\]

(EQ 29)

where \( N \) is the number of cycles at the current stress state, \( (\sigma_k) \), and \( D_k \) and \( D_{k-1} \) are the amount of damage at the current and previous increments, respectively. The quantity \( \alpha \), which is a function of the current stress state is defined as,

\[
\alpha = 1 - \frac{\langle \Phi_{fl} \rangle}{\langle \Phi_u \rangle}
\]

(EQ 30)

where \( \langle \ \rangle \) are the Macauley brackets. In the above, the fatigue limit surface, \( \Phi_{fl} \), and the static fracture surface, \( \Phi_u \), are defined as

\[
\Phi_{fl} = \frac{1}{2} \max_{i} f_{(\sigma_{ij})}(\sigma_{ij}(t) - \sigma_{ij}(t_0)) - 1
\]

(EQ 31)

\[
\Phi_u = 1 - \max_{i} \frac{f_{(\sigma_{ij})}(\sigma_{ij}(t))}{\sigma_{ij}(t)}
\]

(EQ 32)

and the quantity \( \hat{F}_m \), used in EQ. 29 is the normalized stress amplitude, and is defined as,

\[
\hat{F}_m = \frac{1}{2} \max_{i} \frac{f_{(M)}(\sigma_{ij}(t) - \sigma_{ij}(t_0))}{\sigma_{ij}(t_0)}
\]

(EQ 33)

Note, the case \( \langle \Phi_{fl} \rangle = 0 \) indicates static fracture, which is failure, making it unnecessary to perform the fatigue calculations as in this case the subcell is considered to have failed completely. Thus, having to consider the possibility of \( \alpha \) being undefined is unnecessary. The case \( \langle \Phi_{fl} \rangle = 0 \) indicates that the current stress state is below the fatigue limit and thus \( \alpha \) is set equal to 1. This presents a special case when integrating the fatigue damage expression, EQ. 29, and will be considered later in this section.
In the above equations, \( t_0 \) is the time at the beginning of the current load cycle, and \( t \), is some time during the load cycle. The general form for \( F(\sigma_{ij}), (\sigma_n), \) or \( (M) \) may be expressed as,

\[
F(\cdot) = \frac{1}{N(\cdot)^2} \left\{ \left( 4\omega_{(\cdot)}^2 - 1 \right) I_1 + \frac{4\omega_{(\cdot)}^2 - 1}{\eta_{(\cdot)}^2} I_2 + \frac{9}{4} I_3 \right\} 
\]

(EQ 34)

It is in the above expression, in which the evolution of the damage becomes directionally dependent. This simply amounts to the assumption of partial anisotropy, where the “extent” (magnitude) of damage is affected by the directionality of the stress state. Specifically, the directional dependence enters through the quantities, \( I_1, I_2, I_3, \omega_{(\cdot)}, \) and \( \eta_{(\cdot)} \). The quantities, \( I_1, I_2, I_3 \) are invariants having the form,

\[
I_1 = \frac{1}{2} S_{ij} S_{ij} - d_i d_j S_{jk} S_{ki} + \frac{1}{4} (d_i d_j S_{ij})^2
\]

\[
I_2 = d_i d_j S_{jk} S_{ki} - (d_i d_j S_{ij})^2
\]

\[
I_3 = (d_i d_j S_{ij})^2
\]

(EQ 35)

which are a function of the current deviatoric stress state, \( s_{ij}^k = \sigma_{ij}^k - \frac{1}{3} \sigma_{mm}^k \delta_{ij} \), as well as the vector \( d_i \) which defines the materials’ preferred direction (e.g., fiber orientation in a composite). In addition, the terms \( \omega_{(\cdot)} \) and \( \eta_{(\cdot)} \) represent the ratios of longitudinal to transverse normal and shear stresses, respectively. Note, the longitudinal direction is parallel to the preferred direction and transverse is perpendicular to the preferred direction. For initially transversely isotropic materials, \( \omega_{(\cdot)} \) and \( \eta_{(\cdot)} \) are > 1 and for isotropic materials \( \omega_{(\cdot)} \) and \( \eta_{(\cdot)} \) are = 1.

In the context of micromechanics analysis within MAC/GMC, the isotropic simplification of the above representation will be predominately used for the various constituent phases [11]. This isotropic representation is the previously validated NonLinear Cumulative Damage Rule (NLCDR) developed at ONERA (Office Nationale d'Etudes et de Recherches Aerospatiales) for isotropic monolithic metals. However, it maybe desirable to use the transverse isotropic form when dealing with fiber tows in woven composites systems.

### 3.5.1 Above Initial Fatigue Limit

Given a current state of stress, \( \sigma_{ij} \), above the fatigue limit, i.e. \( \alpha \neq 1 \) and integrating EQ. 29 results in an expression for the number of cycles, \( N \), i.e.,
\[ N = \frac{\left( [1 - (1 - D_k)^{\beta + 1}]^{1-\alpha} - [1 - (1 - D_{k-1})^{\beta + 1}]^{1-\alpha} \right)}{F_m(1 - \alpha)(\beta + 1)} \]  

(EQ 36)

Note that \( D_{k-1} \) is the total amount of damage at the beginning of the load block and \( D_k \) is the total amount of damage at the end of this load block. Alternatively, rewriting EQ 36 an expression for the damage, \( D_k \), in terms of the number of cycles and previous damage can be obtained, i.e.,

\[ D_k = 1 - \left( 1 - \left[ (1 - D_{k-1})^{\beta + 1} \right]^{1-\alpha} + (1 - \alpha)(\beta + 1)F_mN \right)^{\frac{1}{1-\alpha}} \]  

(EQ 37)

In the present computational scheme, since the damage increment is controlled, both \( D_k \) and \( D_{k-1} \) are known. That is, \( D_k = D_{k-1} + \Delta D \) where \( \Delta D \) is the user specified increment in damage. Thus EQ 36 is used to predict the increment in the number of cycles for each subcell, \( N^e \), due to the imposed increment in damage.

To calculate the number of cycles to failure, for an initial damage amount, \( D_{k-1} \), let \( D_k = 1 \), which results in the following,

\[ N_F = \frac{(1 - [1 - (1 - D_{k-1})^{\beta + 1}]^{1-\alpha})}{F_m(1 - \alpha)(\beta + 1)} \]  

(EQ 38)

3.5.2 Below Initial Fatigue Limit

Now consider the case in which the current stress state is below the initial fatigue limit, i.e. \( \langle \Phi_f \rangle = 0 \), which leads to \( \alpha_k = 1 \). Thus, EQ. 29 takes the form,

\[ \frac{D_k}{D_{k-1}} \int_{1 - (1 - D_{k-1})^{\beta + 1}}^N dD = \int_{0}^{N} F_m dN \]  

(EQ 39)

Upon integrating the above equation, the increment in cycles, \( N \), with initial damage, \( D_{k-1} \), may be expressed as,

\[ N = \left( \frac{\log[1 - (1 - D_k)^{\beta + 1}] - \log[1 - (1 - D_{k-1})^{\beta + 1}]}{F_m(\beta + 1)} \right) \]  

(EQ 40)

Alternatively, the following expression for the damage, \( D_k \), may be expressed as:
\[ D_k = 1 - \left\{ 1 - \left[ 1 - (1 - D_{k-1})^{\beta + 1} \right] \exp((\beta + 1)\hat{\sigma}_m N) \right\}^{\frac{1}{\beta + 1}} \]  
(EQ 41)

For the number of cycles to failure, let \( D_k = 1 \),
\[ N_F = \frac{-\log[1 - (1 - D_{k-1})^{\beta + 1}]}{\hat{\sigma}_m^{\beta}(\beta + 1)} \]  
(EQ 42)

The effect of damage is included in the present micromechanics analysis utilizing the concept of effective stress and the hypothesis of strain-equivalence [12].
\[ \hat{\sigma} = \frac{\sigma}{(1 - D_k)} \]  
(EQ 43)
3.6 User Defined Inelastic Material Model

MAC/GMC, has the option for a user to implement their own inelastic constitutive model. This is accomplished by using the subroutine USRMAT into which the user writes the necessary FORTRAN code for the particular constitutive model being implemented. The USRMAT subroutine is as shown below and is always called whenever ncmd=99 (see section 4.2.12).

```fortran
SUBROUTINE USRMAT(DSA, SA, PE, PV, D, LOCTISO, TIME, TSTEP,
                   & CTEMP, DTEMPR, NIO, NE, NV, NS, MN, CDUM, DMGF,
                   & NEP, NVP, NSASIZE)

purpose: user material constitutive model for determination of
the inelastic strain and state variable rates
(used when ncmd = 99)

IMPLICIT DOUBLE PRECISION (A - H, O - Z)

CHARACTER*2 CDUM
DIMENSION SS(6), S(6), R(6)
DIMENSION DSA(NSASIZE), SA(NSASIZE)
DIMENSION PV(NVP), PE(NEP), D(3)

note: 1) in this subroutine, [SA] and [DSA] contain the
micro (subcell) quantities for aboudi's micromechanics model
2) arrangement of [dsa] & [sa] arrays:
variable location

| strain rate (1-6) (contains ENGINEERING shears)
| stress rate (7-12)
| inelastic strain rate (13-18) (contains ENGINEERING shears)
| 12 "slots" (19-30)
| for state variables
| thermal strain rate (31-36)

NOTE: quantities in [SA] and [DSA] are SUBCELL quantities - the
values on entry are for the first subcell containing material # MN - the values on exit of this subroutine will be applied to ALL SUBCELLS containing material # MN. It is thus recommended that, if using the field variables, you assign the appropriate material # to ONE SUBCELL ONLY. Use of [SA] and [DSA] in this context in conjunction with bending in laminate theory will result in erroneous results as field variables become dependent.
A description of the input and output required for the USRMAT subroutine is as follows:

Data supplied to USRMAT:

- **SA** array containing current total quantities for all of the state variables
- **PE** array containing elastic constants
PV
D(3)
LOCTISO

array containing inelastic constants
vector of direction cosines (for models 3, 7, & 9)
flag indicating if ANY material exhibits local transverse isotropy (and global anisotropy)

= 0 - all materials are at most globally transversely isotropic (D not used)
= 1 - at least one material is locally transversely isotropic (D used)

TIME
TSTEP
CTEMP
DTEMPR
NE
NV
NIO
NCE
NS
MN
CDUM
DMGF

current time
current time increment (step)
current temperature
current temperature rate
total number of elastic constants
total number of inelastic constants
output file unit number
current subcell number
subcell number
material number of current constituent
Character string identifying current constituent
damage factor - if damage is included the user should multiply material stiffness terms by DMGF when using such terms in their inelastic model.

Output expected from **USRMAT**:

**DSA**

current increments in all state variables

The state variables are arranged in SA and DSA in the following order:

<table>
<thead>
<tr>
<th>Position</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>strain</td>
</tr>
<tr>
<td>7-12</td>
<td>stress</td>
</tr>
<tr>
<td>13-18</td>
<td>inelastic strain</td>
</tr>
<tr>
<td>19-30</td>
<td>available space for 2 6x1 vectors for model dependent internal state variables</td>
</tr>
<tr>
<td>31-36</td>
<td>thermal strain</td>
</tr>
<tr>
<td>37-42</td>
<td>debonding parameters</td>
</tr>
</tbody>
</table>

Again, SA contains the total quantities and DSA contains the rates.

**Note:** Example K contains a sample input file and the **USRMAT** subroutine containing an implementation of the Bodner-Partom viscoplastic material model as well as two elastic models.

**Note:** GMC utilizes engineering strains, in SA and DSA consequently it is the users responsibility to convert tensorial strain quantities to engineering before exiting the **USRMAT** routine.
3.7 User Defined Functional Form For Material Properties

An addition to this version of MAC/GMC is the option for a user to implement functionally dependent material properties. With this the elastic and/or inelastic material properties may be defined as a function of any variable contained in SA and DSA (e.g., stress, strain, time, temperature, etc.) In certain cases, the user must also provide code for determining the stiffness matrix (USRFORMDE) or the time derivative of the stiffness matrix (USRCPEVAL). This is accomplished by using the subroutines USRFUN, USRFORMDE, and USRCPEVAL into which the user writes the necessary FORTRAN code for the particular constitutive model being implemented. All three routines are shown below.

SUBROUTINE USRFUN(MN, TIME, TSTEP, CTEMP, DTEMPR, SA, DSA,
& DOLD, PEM, PVM, D, LOCTISO, ALPA, ALPT,
& NE, NV, NMTS, NEP, NVP, NSASIZE)

* Purpose: user subroutine to allow elastic and viscoplastic
  material properties to be functions of TEMP or
  field variables. Used for user defined functional
  form material properties, that is, when:
  (mat .eq. 'U') and (ifm .eq. 2)

* Note: can be used in conjunction with a provided material
  constitutive model, or a constitutive model input
  by the user in USRMAT

INCLUDE 'parm.inc'
IMPLICIT DOUBLE PRECISION (A - H, O - Z)
DIMENSION DOLD(6, 6), D(3)
DIMENSION PEM(NEP, NMTS), PVM(NVP, NMTS)
DIMENSION ALPA(NMTS), ALPT(NMTS)
DIMENSION DSA(NSASIZE), SA(NSASIZE)

* Note: 1) in this subroutine, [SA] and [DSA] contain the
  micro (subcell) quantities for aboudi's micromechanics model

* 2) arrangement of [dsa] & [sa] arrays:
  variable   location
  +-------------------
  | strain rate       (1-6) (contains ENGINEERING shears)
  +-------------------
  | stress rate       (7-12)
  +-------------------
  | inelastic
  | strain rate       (13-18) (contains ENGINEERING shears)
  +-------------------
  | 12 "slots"        (19-30)
  | for state variables
  +-------------------
  | thermal strain rate (31-36)
  +-------------------
NOTE: quantities in [SA] and [DSA] are SUBCELL quantities - the values on entry are for the first subcell containing material # MN - the values on exit of this subroutine will be applied to ALL SUBCELLS containing material # MN. It is thus recommended that, if using the field variables, you assign the appropriate material # to ONE SUBCELL ONLY. Use of [SA] and [DSA] in this context in conjunction with bending in laminate theory will result in erroneous results as field variables become dependent on through-thickness position.

on entry:
- MN - material number
- TIME - current time
- TSTEP - current time step
- CTEMP - current temperature
- DTEMPR - time rate of change of temperature
- SA - vector of total (integrated) quantities (see above)
- DSA - vector of rate quantities (see above)
- DOLD(6, 6) - previous elastic material stiffness matrix
- PEM(NE, MN) - vector of previous elastic constants for material # MN (where NE = # of elastic constants --> 9 MAX)
- PVM(NV, MN) - vector of previous viscoplastic constants for material # MN (where NV = # of viscoplastic constants --> 19 MAX)

expected on exit:
- PEM(NE, MN) - vector of current elastic constants for material MN
- PVM(NV, MN) - vector of current viscoplastic constants for material MN
- D(3) - vector of direction cosines (required for models 3, 7, & 9)
- LOCTISO - flag indicating if ANY material exhibits local transverse isotropy (and global anisotropy)
- = 0 - all materials are at most globally transversely isotropic (D not used)
- = 1 - at least one material is locally transversely isotropic (D used)
- ALPA(MN) - longitudinal cte for material MN
- ALPT(MN) - transverse cte for material MN

BEGIN USER EDITS
place code here
END USER EDITS
RETURN
END
SUBROUTINE USRFORMDE(MN, PEM, PVM, D, LOCTISO, DNEW, 
& NE, NV, NEP, NVP, NMTS)

purpose: user subroutine to allow formation of material stiffness
matrices based on a user constitutive model (used when
ncmd = 99)

INCLUDE 'parm.inc'

IMPLICIT DOUBLE PRECISION (A - H, O - Z)

DIMENSION DNEW(6, 6)
DIMENSION PEM(NEP, NMTS), PVM(NVP, NMTS)
DIMENSION D(3)

on entry:
MN - material number
PEM(NE, MN) - vector of elastic constants for material
# MN (where NE = # of elastic constants --> 9 MAX)
PVM(NV, MN) - vector of viscoplastic constants for material # MN
(where NV = # of viscoplastic constants --> 19 MAX)
D(3) - vector of direction cosines
(required for models 3, 7, & 9)
LOCTISO - flag indicating if ANY material exhibits
local transverse isotropy (and global anisotropy)
  = 0 - all materials are at most globally transversely
  isotropic (D not used)
  = 1 - at least one material is locally transversely
  isotropic (D used)

expected on exit:
DNEW(6, 6) - current elastic material stiffness matrix

*BEGIN USER EDITS*
place code here
*END USER EDITS*

RETURN
END
SUBROUTINE USRCPEVAL(DSA, SA, MN, TIME, TSTEP, CTEMP, DTEMPR, & DNEW, DOLD, PEM, PVM, D, LOCTISO, ALPA, ALPT, DDOT, & NE, NV, NMTS, NEP, NVP, NSASIZE)

purpose: user subroutine to allow formation of the TIME
derivative of the material stiffness matrix.
this subroutine is used when:
a) material properties are user defined and functional
form. That is: (mat .eq. 'U') .and. (ifm .eq. 2)
b) the constitutive model is user-defined, and the
material properties are not functional form, and
the material properties are temperature-dependent
that is: (ncmd .eq. 99) .and. (ifm .eq. 2) .and.
(ndpt .eq. 2)

IMPLICIT DOUBLE PRECISION (A - H, O - Z)

DIMENSION DNEW(6, 6), DOLD(6, 6)
DIMENSION DDOT(6, 6)

DIMENSION PEM(NEP, NMTS), PVM(NVP, NMTS)
DIMENSION ALPA(NMTS), ALPT(NMTS)
DIMENSION DSA(NSASIZE), SA(NSASIZE)

note: 1) in this subroutine, [SA] and [DSA] contain the
micro (subcell) quantities for aboudi's micromechanics model

2) arrangement of [dsa] & [sa] arrays:
variable location
---
strain rate (1-6) (contains ENGINEERING shears)
---
stress rate (7-12)
---
inelastic
strain rate (13-18) (contains ENGINEERING shears)
---
12 "slots" (19-30)
for state variables
---
thermal strain rate (31-36)

NOTE: quantities in [SA] and [DSA] are SUBCELL quantities - the
values on entry are for the first subcell containing material
# MN - the values on exit of this subroutine will be applied to
ALL SUBCELLS containing material # MN. It is thus recommended
that, if using the field variables, you assign the appropriate
material # to ONE SUBCELL ONLY. Use of [SA] and [DSA] in this
context in conjunction with bending in laminate theory will
result in erroneous results as field variables become dependent
on through-thickness position.
A description of the input and output required for each subroutine is given in the comment statements at the top of each subroutine.

**Note:** MAC/GMC assumes the longitudinal direction for the material is oriented in the $x_1$-coordinate direction, see Fig. 6.

**Note:** Example K illustrates how one could use each of these routines to define material properties as well as the associated elastic stiffness matrix.
3.8 Interface Modeling

Interfaces in composite materials play a major role in the determination of their mechanical and thermal properties. Consequently, it is important to have the ability to model interface behavior accurately. This is accomplished in MAC/GMC in one of two ways. The first is to define an actual interface region with its own constitutive behavior. In this way the influence of initial imperfections (flaws, voids, improper wetting, etc.) and induced interfacial damage (due to stress, environment, chemical reactions, etc.) may be incorporated into the micromechanical analysis of the overall behavior of the composite. The development of proper interfacial constitutive models is an active area of research, and MAC/GMC, through the use of its USRMAT routine, provides the researcher with a convenient tool for testing new and existing interfacial constitutive models.

The second approach to modeling the effect of imperfect (weak) bonding between two phases (e.g. a fiber and a matrix) is to assume that a jump in the displacement field at an interface may occur given certain conditions, while still maintaining continuity of the traction vector. In the spirit of Jones and Whittier [13] and Achenbach and Zhu [14] we have assumed the following flexible interface model.

\[
\begin{align*}
(u_n^I &= R_n \cdot \sigma_n^I) \\
(u_t^I &= R_t \cdot \sigma_t^I)
\end{align*}
\]

if

\[
\begin{align*}
\sigma_n^I &\geq \sigma_{DBn} \\
\sigma_t^I &\geq \sigma_{DBt}
\end{align*}
\]

(EQ 44)

\[
\begin{align*}
(u_n^I &= 0) \\
(u_t^I &= 0)
\end{align*}
\]

if

\[
\begin{align*}
\sigma_n^I &< \sigma_{DBn} \\
\sigma_t^I &< \sigma_{DBt}
\end{align*}
\]

(EQ 45)

where \(R_n, R_t, \sigma_{DBn}\) and \(\sigma_{DBt}\) are the interfacial normal and shear, compliance and debond stresses, respectively. Note, the implementation of the various forms for \(R_n\) and \(R_t\) described below will impact the definition of the concentration matrices, \(A^{(a\beta\gamma)}\) in the original GMC formulation or its counterpart in the reformulated version, see section 3.1.

This approach to debonding has been implemented in MAC/GMC in two forms. In the first, \(R_n\) and \(R_t\) are assumed to be constants that are independent of time. Therefore, when the time derivative of EQ. 44 is taken (as it is for implementation in the incremental formulation of MAC/GMC) the expression becomes,
\[
\begin{align*}
\begin{cases}
\dot{u}_n' = R_n \cdot \dot{\sigma}_n' \\
\dot{u}_t' = R_t \cdot \dot{\sigma}_t'
\end{cases}
\quad \text{if} \quad \begin{cases}
\sigma_n' \geq \sigma_{DBn} \\
\sigma_t' \geq \sigma_{DBt}
\end{cases}
\tag{EQ 46}
\end{align*}
\]

Hence, the debonding is instantaneous, and immediately reaches its full extent. If \( R_n \) and \( R_t \) are chosen to be sufficiently large (as is customary), the stress at the interface will remain constant, with a value of \( \sigma_{DB} \) after debonding occurs.

In the second form, \( R_n \) and \( R_t \) are assumed to be functions of time. Thus we obtain

\[
\begin{align*}
\begin{cases}
\dot{u}_n' = R_n \cdot \dot{\sigma}_n' + \dot{R}_n \cdot \sigma_n' \\
\dot{u}_t' = R_t \cdot \dot{\sigma}_t' + \dot{R}_t \cdot \sigma_t'
\end{cases}
\quad \text{if} \quad \begin{cases}
\sigma_n' \geq \sigma_{DBn} \\
\sigma_t' \geq \sigma_{DBt}
\end{cases}
\tag{EQ 47}
\end{align*}
\]

in lieu of EQ 46. The additional term present in EQ. 47 for both normal and tangential debonding is significant. If the time-dependence of \( R_n \) and \( R_t \) is chosen wisely, these additional terms will enable local unloading in the composite. For implementation in \textit{MAC/GMC}, the following functional form of the time-dependence has been employed,

\[
R(t) = \Lambda \left[ \exp \left( \frac{\dot{\tau}}{B} \right) - 1 \right]
\tag{EQ 48}
\]

where \( \dot{\tau} \) is the time since debonding, and \( \Lambda \) and \( B \) are functional parameters which characterize how the interface unloads.

Figure 5 shows a simple example to illustrate the differences between the two implementations of this debond model. The repeating unit cell used to generate the results shown in this figure, is IDP = 1 as illustrated in Fig. 9. As Fig. 5 shows, the results generated using both implementations of the debond model are the same until debonding occurs at an interfacial stress of approximately 15 ksi. At this point, the stress at the interface in the first implementation becomes constant, while, in the case of the second implementation, the slope of the interfacial stress verses applied strain curve decreases and then the interface begins to unload. The effect of the difference between the two implementations on the predicted composite stress-strain response is shown clearly in Fig. 5, where the softer composite response is a result of additional inelasticity in the remaining matrix subcells due to the local stress redistribution from the debonded subcells.
Figure 5. Simulated transverse behavior using the two implementations of the debond model in MAC/GMC.
3.9 References


4.0 Running MAC/GMC

MAC/GMC can be executed in one of two ways; 1) as a stand-alone code, and 2) through the UMAT facilities of the nonlinear finite element code, ABAQUS. Upon linking the associated object modules that comprise MAC/GMC, an executable version of MAC/GMC is generated as well as the file "feamac.o". The file "feamac.o" is one of the required files to be provided to ABAQUS for finite element access to MAC.

4.0.1 MAC/GMC - stand-alone:

The execution of MAC/GMC as a stand-alone code utilizes I/O redirection. Execution of MAC/GMC is initiated by typing a line at the command prompt such as;

MAC < infilename > outfilename

4.0.2 FEAMAC - Accessing MAC through ABAQUS:

To execute MAC/GMC through ABAQUS several files are required, these are summarized in the next section. The execution of ABAQUS for MAC/GMC access is as follows;

abaqus job=abaqus_input_deck user=feamac.o

Note: Please check the "readme" file included with your version for any last minute changes to this manual!

OR you can check our web page at
www.lerc.nasa.gov/WWW/LPB/mac/index.html
4.1 Input and Output Files

The input and output files required for the stand-alone version of MAC/GMC and FEAMAC are specified in different ways. The differences between these two methods of running MAC/GMC are detailed below.

4.1.1 MAC stand-alone

Since MAC/GMC reads from standard in and writes to standard out, the name of the files used as input and output are left to the user's discretion, i.e., will have no impact on the performance of the MAC/GMC code. As will be explained in the following sections, the user will also need to specify the names of the PATRAN output files and the files containing the X-Y data for plotting.

Note that a “debug” file may be generated depending on the “PRINT” level the user specifies in the input file (see 4.2.2). This file will have the name “mac_debug”.

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>infilename</td>
<td>outfilename</td>
</tr>
<tr>
<td></td>
<td>debug file (mac_debug)</td>
</tr>
<tr>
<td></td>
<td>PATRAN files (optional, see section 4.2.9)</td>
</tr>
<tr>
<td></td>
<td>plot data files (see section 4.2.16 and 4.2.17)</td>
</tr>
<tr>
<td></td>
<td>damage files (optional, see section 4.2.6)</td>
</tr>
<tr>
<td></td>
<td>yield surface files (optional, see section 4.2.7)</td>
</tr>
</tbody>
</table>

4.1.2 FEAMAC

Almost all of the capabilities of MAC/GMC are available through the user defined material facilities of the ABAQUS finite element code. Access to MAC/GMC from within ABAQUS is outlined here. Currently, MAC/GMC, can only be access using 3D elements; however, FEAMAC has only been specifically tested using the C3D8 element within ABAQUS. Note any options not available or modified for finite element implementation are denoted subsequently by the following notation, for example:

FEAMAC Note: Not required by FEAMAC

An example containing an ABAQUS input deck, required FORTRAN subroutines, and FEAMAC input files are given in Example O.

ABAQUS Input Deck:

To utilize FEAMAC, the ABAQUS input deck must contain the "**USER
SUBROUTINE", "*USER DEFINED FIELD", "*INITIAL CONDITIONS, TYPE=FIELD", "*EXPANSION" (if conducting nonisothermal analysis) and "*USER MATERIAL" card. Where the USER SUBROUTINE card identifies the name of the file containing the FORTRAN subroutines required to access MAC/GMC and the "USER MATERIAL" card instructs ABAQUS to obtain the materials stress/strain behavior from the supplied subroutines. The name of the user material (defined in the ABAQUS input deck), converted to upper case, will be appended with the file extension that has been defined in the user supplied FORTRAN routines, consequently this name must be used as the MAC/GMC input filename for the given material. Note: the material name is limited by ABAQUS to be 8 characters or less. See the ABAQUS Users' Manual for a complete descriptions of these cards.

**ABAQUS FORTRAN Subroutine Files:**

The name of this file is to be provided on the "*USER SUBROUTINE" card inside the ABAQUS input deck. These subroutines will be executed for every integration point of every element in the finite element model associated with a given material group. The required subroutine are listed in Example O. Only a portion of the FEAMAC_INIT routine should be edited by the user. This portion is bounded by the comments;

```
C *** BEGIN USER EDITS ***
and
C *** END OF USER EDITS ***
```

The seven variables within this section that can be edited are as follows:

- **PATH:** CHARACTER*80
  Path to working directory

- **EXTENSION:** CHARACTER*80
  File name extension to be appended to material name to form the name of the MAC/GMC input deck.

- **DFNAME:** CHARACTER*80
  Name of diagnostic file.

- **NPEL:** INTEGER
  Number of integration points whose data is to be plotted.

- **N_PEN(N):** INTEGER ARRAY (N:1->NPEL)
  Element numbers whose data is to be plotted.

- **N_PIN(N):** INTEGER ARRAY (N:1->NPEL)
  Integration numbers whose data is to be plotted.
**Note:** N_PEN() and N_PIN() pairs define locations of data extraction to be utilized with the *MACRO or *MICRO MAC/GMC options.

AB_PRINT: INTEGER
Diagnostic print level for FEAMAC.

0 - NONE
1 - UMAT: STRESS, STRAIN, INCREMENTAL AND EXECUTION TRACING INFORMATION
3 - FEAMAC_PRE: STRESS, STRAIN, INCREMENTAL AND EXECUTION TRACING INFORMATION
4 - TEMPERATURE INFORMATION
5 - STATE INFORMATION BEFORE AND AFTER CALL TO FEAMAC
25 - CONSTITUTIVE MATERIAL PROPERITES

**Note:** Higher values of AB_PRINT provide all information provided by lower values plus additional information as described above.

**FEAMAC Input Deck:**

The FEAMAC input deck can be identical to the stand-alone MAC/GMC input deck as discussed in section 4.2. The associated file name is obtained by converting the material property name, defined within the ABAQUS input deck, to uppercase and appending the file extension specified in the FEAMAC_INIT subroutine described previously. Since the load history definition and time integration are dictated by the associated ABAQUS, input deck, information in the MAC/GMC deck that relates to these issues will be ignored. All entries in section 4.2 that describe MAC/GMC options should be considered to be required by FEAMAC unless denoted otherwise.

**Note:** Multiple FEAMAC input decks can be specific per ABAQUS run, so that multiple material systems can be analyzed within a given structure. Utilization of this option will allow the most efficient execution of large problems. For example, if one knows that only a small portion of the problem will be inelastic then define the zone of elements that may go inelastic by a given material group and the remainder by another. In this way maximum speed can be achieved, as a purely isothermal, elastic, problem with no micro fields being printed executes the fastest.
FEAMAC Output Files:

Most diagnostic messages from FEAMAC appear in the ABAQUS msg file. Files containing X-Y plotting information are generated for each N_PEN/N_PIN pair. The names for these files correspond to those defined in the MAC/GMC deck except that the N_PEN and N_PIN number are appended to the name.

Example:
If in the FEAMAC_INIT subroutine the user specified:

\[
\begin{align*}
NPEL &= 1 \\
N\_PEN(1) &= 18 \\
N\_PIN(1) &= 7
\end{align*}
\]

and in the FEAMAC deck one specified:

\[
\begin{align*}
*\text{CURVE} \\
\text{NP} &= 2 \\
*\text{MACRO} \\
\text{NT} &= 1 \\
\text{NC} &= 1 \quad X = 1 \quad Y = 7 \quad \text{NAM} = \text{plot\_file\_sve} \%
\end{align*}
\]

FEAMAC would provide the stress and strain data in the 11-direction for integration point 7 of element number 18 in a file named

"PLOT\_FILE\_SVE.18.7.\feamacro.data".

\textbf{Note:} see section 4.2.17 regarding curve data input.
4.2 Input Requirements (for both MAC/GMC and FEAMAC)

This section describes how the data should appear in the input file, `infilename`, where each line in the input file is limited to 80 characters. In the following, each block of input data will have its own subsection and will typically contain the following information:

1) statement of purpose
2) declaration of input data block
3) example(s)
4) notes

A data block has the following general format:

1) **KEYWORD**
   beginning of data block
2) input line(s)
3) %
   end of data block

where

1) **KEYWORD**: (denotes beginning of data block)

   Each input data block begins with the appropriate keyword, which starts with the * symbol. For example, the keyword for selecting the method of integration is **SOLVER**. The input routine scans the input file and locates the appropriate keyword and then reads the corresponding input data.

2) Input line(s):

   The input lines contain the necessary input data. The specific format of these input lines will be given in the following subsections. The capital letters denote actual variable names and lower case letters represent the possible input choices. Multiple inputs can be on one line, just as shown in the following. However, at least one space must be used to separate **data sets** on a single line.

   **Two special characters** (& and #) are provided for entering the input.

The "&" (continuation) symbol:

For input data that is too long to fit on a **single 80 character line**, the "&" symbol is used as the continuation character. Thus, a line of data may be divided into a series of lines. For example, when entering the material properties all of the data cannot fit on a single line, thus the continuation character is required:

```plaintext
EL=11700.,11700.,0.365,0.365,4287.5,1.,1. &
VI=0.8E-8,0.1,0.1E-5,0.,0.85E-3,0.05,1.,1.,3.3,1.8,1.35,1.,0.01
```
The "#" (comment) symbol:

The input file may also contain "comments" for the users convenience. The "#" symbol is used to mark a comment line, with the requirement that the "#" symbol appears in the first column.

3) % (denotes end of data block)

Each input block ends with the "%" symbol. The % symbol should be included as it signals the input routine that a particular data block has been completed.

A sample data block is as follows,

```
*SOLVER
#  NTF=2 ISTM=0.0001 ERR=0.1E-3
  NTF=2 ISTM=0.0000024 ERR=0.1E-2 
```

Note how the # symbol is used to comment out a line of input data. Thus, the user can change various parameters by simply commenting out input lines containing different input data. Also, note how the data block is terminated with the % symbol and how it need not appear on a separate line.

**Note:** It is suggested that the order of keywords in a given input file follow that given in this manual as some keyword ordering may give rise to problems, since in certain cases, MAC/GMC uses input from previous keywords to determine which input to read from later keywords.
4.2.1 Header Line:

Purpose: Define the title of this particular job (80A format).

**Note:** the **Header Line** is limited to one 80 character line, and is always taken as the first line in the input file.

**FEAMAC Note:** Required by FEAMAC

* problem title

Example: Transverse tensile response of 35% SCS6/Ti-6-4

4.2.2 Output Print Level:

Purpose: To control the output generated.

**FEAMAC Note:** Required by FEAMAC

*PRINT

NPL=nplvl

%

Where:

*nplvl:*

-1 = print out macro stiffness matrix, engineering constants and macro-thermal expansion coefficients and stop
0 = Input echo, minimal output (most commonly used)
1 = time and iteration information
3 = Material properties/Stiffness matrix output
5 = Lamination information
7 = Damage information
9 = Ancillary data
10 = program execution trace and all array data

**CAUTION:** the choice 10 generates a very large output file

**Note:** Print levels are inclusive, such that the current level information as well as all lower level information is printed.

Example: (minimal print out)

*PRINT

NPL=0 

53
4.2.3 Load Type:
Purpose: To select load type

FEAMAC Note: Only LCON is required for FEAMAC, LOP, and LSS are Ignored by FEAMAC.

*LOAD
  LCON=nset  LOP=lop  LSS=iopt
%

Where

nset:
  1 = Thermal Load
  2 = Mechanical Load
  3 = Thermomechanical Load

lop: (for 2-D and 3-D GMC)
  1 = axial load in 1-direction
  2 = axial load in 2-direction
  3 = axial load in 3-direction
  4 = shear load 23-direction
  5 = shear load 13-direction
  6 = shear load 12-direction
  99 = generalized loading in potentially all 6 directions

For 2-D GMC:
  7 = axial loads in 1 and 2 directions
  8 = axial loads in 2 and 3 directions
  9 = axial load in 1-direction and shear load in 23-direction
 10 = axial load in 2-direction and shear load in 13-direction

For 3-D GMC:
  7 = axial loads in 1 and 2 directions
  8 = axial loads in 2 and 3 directions
  9 = axial loads in 1 and 3 directions
 10 = axial load in 1-direction and shear load in 23-direction
 11 = axial load in 2-direction and shear load in 13-direction
 12 = axial load in 3-direction and shear load in 12-direction

iopt:
  1 = Strain control
  2 = Stress control

Note: If lop = 99 one must now enter six iopt values, for example:
    LCON= nset LOP=lop LSS = iopt1, iopt2, iopt3, iopt4, iopt5, iopt6

Note: see Fig. 6 for the definitions of the directions mentioned above

Note: If using laminate option; lop = 1, 2 or 6 are the only valid selections
Example: General loading option

*LOAD
  LCON=1 LOP=99 LSS=2,1,2,2,2,2 %

**Note:** This would correspond to specifying mixed stress and strain control, i.e.,:

\[ \sigma_{11}, \varepsilon_{22}, \sigma_{33}, \tau_{23}, \tau_{31}, \tau_{12} \]

If however, the loads in the *MECH section were set to 0, 0.01, 0,0,0,0 this would correspond to a pure strain control problem in the 22 direction.
Double Periodicity

Triple Periodicity

Figure 6: Coordinate Systems
4.2.4 Mechanical Load Control:

Purpose: Select type of load control for mechanical load

Note: This block is only required if LCON = 2 or 3

FEAMAC Note: *All the data on the MECH card is ignored by FEAMAC*

\[
*\text{MECH} \quad \text{NPTW}=nptw \quad Tl=t_1, t_2, ..., t_{nptw} \quad LO=l_1, l_2, ..., l_{nptw}
\%
\]

Where:

\[nptw: \text{number of points on load curve}\]
\[t_1, t_2, ..., t_{nptw}: \text{time values}\]
\[l_1, l_2, ..., l_{nptw}: \text{load curve values}\]

Example:

\[
*\text{MECH} \quad \text{NPTW}=3 \quad Tl=0, 1, 2 \quad LO=0, 20, 30
\%
\]

Note: For LOP=7,8,...etc (i.e., two load/displacement components) the following format is required *(See Example J for an example)*:

\[
*\text{MECH} \quad \text{NPTW}=2 \quad Tl=0, 1.5 \quad LO=0, 0.015 \quad \text{component 1 curve}
\quad \text{NPTW}=2 \quad Tl=0, 1.5 \quad LO=0, 0.01 \% \quad \text{component 2 curve}
\]

For LOP=99, six load/displacement components are required.
4.2.5 Temperature Control:

Purpose: Select control for temperature.

Note: This block is only required if LCON = 1 or 3

FEAMAC Note: *All the data on the THERM card is ignored by FEAMAC*

\[ \text{*THERM} \]
\[ \text{NPTT}=nptt \quad \text{TI}=t_1, t_2, ..., t_{nptt} \quad \text{TE}=te_1, te_2, ..., te_{nptt} \%
\]

Where:

- \( nptt \): number of points on temperature curve
- \( t_1, t_2, ..., t_{nptt} \): time values
- \( te_1, te_2, ..., te_{nptt} \): temperature curve values

Example:

\[ \text{*THERM} \]
\[ \text{NPTT}=3 \quad \text{TI}=0., 1., 2. \quad \text{TE}=100., 200., 150. \%
\]
**Note:** For the thermomechanical load $t_{nptw} = t_{nptt}$, and both curves must have $t_1 = 0$. But the number and time value of the data points in-between maybe different, see figure 7.

**Temperature**

$te_1$ $te_2$ $te_3$ $te_4$ $te_5$

$t_1$ $t_2$ $t_3$ $t_4$ $t_5$

Time

**Stress or Strain**

$l_1$ $l_2$ $l_3$ $l_4$ $l_5$ $l_6$

$t_1$ $t_2$ $t_3$ $t_4$ $t_5$ $t_6$

Time

**Figure 7: Load History Specification**
4.2.6 Fatigue Damage Option:

Purpose: Activate the Fatigue Damage analysis option

Note: Currently this option is **only available** for continuous reinforcements, modid=1 see section 4.2.8

**FEAMAC Note:** This option is currently **not available** from within **FEAMAC**

**DAMAGE**

\[ NCY=ncycle \quad D=deld \quad DMAX=dmax \quad FG=fg \quad FL=fl \]

where:

- \( ncycle \): number of "blocks" of cycles to be run (see Fig. 5)
- \( deld \): damage increment
- \( dmax \): maximum amount of damage allowed \((0 \leq dmax \leq 1)\)

\( fg \):

- 0 - no macro (global) failure check
- 1 - macro failure check

\( fl \):

- 0 - no micro failure check
- 1 - micro failure check

Note: For \( FG=1 \), the following data is required on a new line;

\[ T=ss \quad IC=comp \quad V=val \]

Where

- \( ss \):
  - 1 - stress
  - 2 - strain

- \( comp \):
  - 1 - component 11
  - 2 - component 22
  - 3 - component 33
  - 4 - component 23
  - 5 - component 13
  - 6 - component 12

- \( val \): value of failure stress or strain (depending on value of \( ss \))

Note: For \( FL=1 \), see description of additional required data in section 4.2.12
Note: the actual time vs. load data for the "block" of cycles is specified using the previously defined *MECH input data (section 4.2.4).

![Figure 8](image)

**Note:** When the fatigue damage analysis option is activated the following additional files will be generated:

- **dam1.data** intermediate damage calculations
- **dam2.data** print-out of cycles to failure at the end of each load block
- **dam3.data** print-out of current damage in each subcell at the end of each load block

**Example:**

* DAMAGE  
NCY=2 D=0.25 DMAX=0.95 FG=1 FL=0  
T=1 IC=2 V=100.

**Note:** In the above example:

- 2 "blocks" of cycles have been specified via NCY
- the damage increment is 0.25 (25%) with a maximum allowable amount of damage set to 0.95 (95%)
- a Global failure criteria is used: stress component -22 with a failure value of 100.
4.2.7 Yield and Damage Surface Generation Analysis

Purpose: Initiate probing history so as to generate flow/damage surface plots

Note: Currently this option is only available for continuous reinforcements, modid=1 see section 4.2.8

FEAMAC Note: All the data on the SURF card is ignored by FEAMAC; option is meaningless in the context of finite element.

*SURF

NPRE=npre ISP=isp IAN=ian C1=c1 C2=c2 C3=c3 C4=c4

where:

npre - the number of preloading steps before probing for the yield surface begins (e.g., this is allows one to represent stress-free cooling)

isp - stress space
  = 1 transverse-axial (σ_{22} - σ_{11})
  = 2 transverse-transverse (σ_{22} - σ_{33})
  = 3 shear-axial (σ_{12} - σ_{11})

ian - probe angle increment in degrees

“yield” criteria:

  c1 - equivalent plastic strain, \( \sqrt{\frac{2}{3} \Delta \epsilon_{ij} \Delta \epsilon_{ij}} \)
  c2 - Surface of Constant Dissipation Rate, SCDR, \( \Sigma_{ij} \cdot \dot{\epsilon}_{ij} \)
  c3 - Surface of Constant Inelastic Strain Rate, SCISR, \( \dot{\epsilon}_{ij} \cdot \dot{\epsilon}_{ij} \)
  c4 - Surface of Constant Inelastic Power, SCIP, \( \bar{\sigma} \cdot \Delta \epsilon_{ij} \)

Note: Results from application of this option are described in: Lissenden C. J. and Arnold, S. M.; “Theoretical and Experimental Considerations in Representing Macroscale Flow/Damage Surfaces For Metal Matrix Composites", Int. Jnl. of Plasticity, Vol. 13, No. 4, pp. 327-358, 1997.

Note: Upon fulfillment of each criteria (i.e., c1, c2, c3, and c4) the probe angle and stress vector are written to output files surf1.dat, surf2.dat, surf3.dat, and surf4.dat, respectively, for post processing by the user.

Note: Probing continues until all four criteria are satisfied or the specified load history is completed for a given probe angle. Thus, to render a particular criterion inactive, use a value of 0 for the criterion so it is fulfilled immediately, thereby rendering the associated output meaningless. Also, it is suggested that a large load history be imposed to ensure yielding along a given probe angle occurs.
4.2.8 Micromechanics Model Identification:

Purpose: Select desired GMC micromechanics model

*MODEL
MOD=modid

☞ Note: If modid=3 (laminate option) you must enter the following line of data,
MATSYS=matsys NLY=nly THK=thk1, thk2, ..., thknly CON=c1,
c2, ..., cnly SYS=s1, s2, ..., snly
ANG=a1, a2, ..., anly

☞ FEAMAC Note: Currently laminate option (modid=3) is not available within
the finite element implementation.

% Where

modid: - defines micromechanics analysis
1 = double periodicity
2 = triple periodicity
3 = laminate option
matsys: - number of different material systems in the analysis
nly: - number of layers in the laminate

thk1, thk2, ..., thknly: - thickness of each layer

c1, c2, ..., cnly: - layer material type id: 1 = isotropic 2 = anisotropic
s1, s2, ..., snly: - material system id number

Example 1: double periodicity model

*MODEL
MOD=1 %

Example 2: laminate option with 1 material system

*MODEL
MOD=3 MATSYS=1 NLY=4 THK=0.25,0.25,0.25,0.25 CON=2,2,2,2 &
SYS=1,1,1,1 ANG=45.,-45.,-45.,45. %

63
4.2.9 PATRAN Output: (Optional)

Purpose: To enable generation of additional output files for use within the MSC/PATRAN graphical pre- and post-processing environment - when running the MACPOST program. For details on using MACPOST, see NASA TM 209062

**Note:** This is an optional input line. If output for use in the MACPOST program is not desired, do NOT include this data block. Inclusion of this data block will increase execution time significantly, particularly for complex RVE's, as microfield data for each subcell are written to additional output files.

*PATRAN

    FN=prefix  TPRE=tpre  STP=npstp

%  

Where:

* prefix: - filename prefix that will be assigned to the geometry file for use in MACPOST.  
* tpre: - preloading time after which output to data files begins  
* npstp: - time step interval for output to the MACPOST data files (output occurs every npstp time steps).  

** CAUTION:** a small value for npstp will cause large output files to be generated, which can result in significantly longer execution times.

Example 1: enable the generation of MACPOST output files using prefix "run1"

*PATRAN

    fn=run1  TPRE=57600  STP=300  %

** Note:** If this option is chosen, MAC/GMC generates 14 additional output files for use in the MACPOST program within MSC/PATRAN. After the time reaches 57600 units, data is written every 300 time steps. The following 14 output files are generated:

* prefix.macgeo: Contains unit cell (RVE) geometry data.  
* prefix.total_pat.data: Contains number of time steps and number of subcells data
**prefix.macro1_pat.data:** Macroscopic strain data at different times
**prefix.macro2_pat.data:** Macroscopic stress data at different times
**prefix.macro3_pat.data:** Macroscopic inelastic strain data at different times
**prefix.macro4_pat.data:** Macroscopic thermal strain, creep time, temperature, and stress invariant data at different times

**prefix.micro1_pat.data:** Microscopic (subcell) strain data at different times
**prefix.micro2_pat.data:** Microscopic (subcell) stress data at different times
**prefix.micro3_pat.data:** Microscopic (subcell) inelastic strain data at different times
**prefix.micro4_pat.data:** Microscopic (subcell) thermal strain, creep time, temperature, and stress invariant data at different times

**prefix.micro1_pat.contour:** Microscopic (subcell) strain data for contour plots
**prefix.micro2_pat.contour:** Microscopic (subcell) stress data for contour plots
**prefix.micro3_pat.contour:** Microscopic (subcell) inelastic strain data for contour plots
**prefix.micro4_pat.contour:** Microscopic (subcell) thermal strain, creep time, temperature, and stress invariant data for contour plots

See Example S for an example illustrating how to construct the required input deck that utilizes the PATRAN option and typical results.

---

**Note:** All 14 files listed above must be present in the active directory used in MSC/PATRAN in order to use the MACPOST program. For more details on the content of the various MACPOST files, see NASA TM 209062.

**FEAMAC Note:** Invoking this option within FEAMAC, will cause all "_pat" extensions in the above output files to be changed to "_feapat"
4.2.10 Integrator Identification

Purpose: Select type of integration scheme

FEAMAC Note: *All the data on the *SOLVER card is ignored by FEAMAC*

*SOLVER*

NTF=ntf ... (additional data as specified below) %

If NTF=1, enter the following:

NTF = 1 NPTS=npts TIM = t1,t2,...tnpts STP=st1,st2,...stnpts-1 %

If NTF=2, enter the following:

NTF = 2 ISTM=istpm ISTT=istpt ERR=errtol %

Where:

ntf:

1 = Forward Euler method (specified time step)
2 = Predictor/Corrector method (self-adaptive time step)

npts: = number of time points

Ti: = start and end times for load increments

Sti: = time step for that load increment

istpm: = initial mechanical load time step

istpt: = initial thermal load time step

errtol: = error tolerance for predictor/corrector

Note: only istpm is required for a mechanical load (nset=-2, see section 4.2.3), only istpt is required for a thermal load (nset=-1) and both istpm and istpt are required for a thermomechanical load (nset=-3)

Note: errtol is only required when using predictor/corrector (ntf = 2)
Suggested errtol = 0.1, if one sees oscillations in predicted response curve the errtol should be lowered.

Note: Laminate option and cases using the second debonding implementation must use forward euler.
**Example 1:** for a mechanical load, select predictor/corrector, with initial time step = 0.001 and error tolerance = 0.01

* **SOLVER**
  NTF=2 ISTM=0.001 ERR=0.01%

**Example 2:** select Forward Euler, time step = 0.01 and 0.02

* **SOLVER**
  NTF=1 NPTS=3 TIM=0.,1.,2. STP=0.01,0.02%

\[ \begin{align*}
  s_{t1} &= 0.01 \\
  s_{t2} &= 0.02 \\
  t_1 &= 0. \\
  t_2 &= 1. \\
  t_3 &= 2.
\end{align*} \]

---

**4.2.11 Thermal Conductivity Calculation:** (Optional)

**Purpose:** Calculate the coefficients of thermal conductivity for the composite

* **COND**

**Note:** If the **COND** keyword is not used, the thermal conductivity calculations are not performed.

**Note:** To use this option, all materials must be user input material properties (mat=U, see section 4.2.12) and must be at the same temperatures. **MAC/GMC** will then calculate the global thermal conductivities at the material property input temperatures, and write them to the output file, see Example T.
**4.2.12 Constituent Material Model Identification:**

Purpose: To select the model for the fiber and matrix constituents.

**FIBER**

NFIBS=nfibs

The following new line is to be repeated for each fiber (nfibs):

\[
\begin{align*}
NF=nf_f & \quad MS=ms_f & \quad MF=ncmd_f & \quad NDPT=dpt & \quad TEMP=mtemp & \quad MAT=mat_f & \quad IFM=ifm \\
D=d_1,d_2,d_3
\end{align*}
\]

**MATRIX**

NMATX=nmatx

The following new line is to be repeated for each matrix (nmatx):

\[
\begin{align*}
NM=nm_m & \quad MS=ms_m & \quad MM=ncmd_m & \quad NDPT=dpt & \quad TEMP=mtemp & \quad MAT=mat_m & \quad IFM=ifm \\
D=d_1,d_2,d_3
\end{align*}
\]

**MONOL (available for laminate option only)**

NMON=nmon

The following new line is to be repeated for each monolithic material (nmon):

\[
\begin{align*}
NMO=nmo_{iso} & \quad MS=ms_{iso} & \quad MMO=ncmd_{iso} & \quad NDPT=dpt & \quad TEMP=mtemp & \quad MAT=mat_{iso} & \quad IFM=ifm
\end{align*}
\]

%  

Where:

- **nfibs**: - number of different fibers
- **nmatx**: - number of different matrices
- **nmon**: - number of different monolithic layers
- **nf**: - fiber material designation number running from 1 to nfibs, sequentially.
- **nm_m**: - matrix material designation number running from 1 to nmatx, sequentially.
- **nmo_{iso}**: - monolithic material designation number running from 1 to nmon, sequentially.
- **msf**: - fiber material system ID (**required only when using laminate option**)
- **ms_m**: - matrix material system ID. (**required only when using laminate option**)
- **ms_{iso}**: - monolithic material system ID
ncmd: - material model identifier for either fiber, matrix or monolithic layer:
   1 = Bodner-Partom Model
   2 = Modified Bodner-Partom Model
   3 = Robinson Viscoplastic Model
   4 = Generalized Viscoplastic Potential Structure (GVIPS) Model
   6 = Transversely Isotropic Elastic Model (2-3 isotropic plane)
   7 = Transversely Isotropic GVIPS Model (TGVIPS)
   9 = Local Transversely Isotropic Elastic Model
   99 = User defined model (see note at the end of section for special format
   instructions)

mat: - material identification letter for either fiber or matrix, selected from mate-
   rial database, see Table II.

**Note:** By specifying MAT=U allows the user to specify the material constants
according to the formats specified below.

dpt: - flag indicating whether material constants should be temperature inde-
   pendent or temperature dependent
   1 = Temperature Independent
   2 = Temperature Dependent

mtemp: - the constant temperature at which material properties are to be taken
(only required for dpt=1 and when using database properties)

ifm - flag indicating whether material properties will be read from input file or
taken from a user defined function (provided in the USRFUN subroutine) ONLY NEEDED when MAT=U.
   1 = read from input file
   2 = functional form taken from USRFUN routine

di: - direction vector defining the normal to the plane of local isotropy
(only required for ncmd = 3, 7 or 9)

**Note:** If modid=1 (Double periodicity) and one desires transverse isotropy using
ncmd=3,7 or 9 then the strong material direction must be specified in the 1-
direction.

**Note:** Additional Input is required if the *DAMAGE* (section 4.2.6) option is invoked

The following additional data is entered on a new line:

ANG=θ  BN=b  BP=b′  OMU=ω_u  OMFL=ω_μ  OMM=ω_m  ETU= flux &
ETFL= flux  ETM= flux  BE=β  A=a  SFL=σ_f  XML=M  SU=σ_u  SK=sk %

where:
\textit{sk:}

\begin{itemize}
  \item 1 skip fatigue damage calculations for this material
  \item 0 perform fatigue damage calculations
\end{itemize}

The remaining constants are described in section 3.5

\textbf{Note:} In addition, for FL=1 (micro failure criteria) in section 4.2.6, the additional data is also required:

\begin{itemize}
  \item T=ss IC=icomp V=val
\end{itemize}

\begin{itemize}
  \item \textit{ss:}
    \begin{itemize}
      \item 1 - stress
      \item 2 - strain
    \end{itemize}
  \item \textit{comp:}
    \begin{itemize}
      \item 1 - component 11
      \item 2 - component 22
      \item 3 - component 33
      \item 4 - component 23
      \item 5 - component 13
      \item 6 - component 12
    \end{itemize}
  \item \textit{val} - value of failure stress or strain (depending on value of \textit{ss})
\end{itemize}

\textbf{Example 1:} select 1 fiber, SCS-6, and 1 matrix material, TIMETAL 21S, both read from database; i.e., SCS-6/TIMETAL 21S composite system

\begin{verbatim}
*FIBER
  NFIB=1
  NF=1 MF=6 NDPT=1 TEMP = 23 MAT=D %

*MATRIX
  NMATX=1
  NM=1 MM=4 NDPT=1 TEMP = 23 MAT=A %
\end{verbatim}

\textbf{Note:} See Section 3.3 for a mathematical description of each material model.

\textbf{Example 2:} (select 2 matrix materials; material 1: Boron, read from database, material 2: user supplied properties)

\begin{verbatim}
*MATRIX
  NMATX=2
  NM=1 MM=6 NDPT=1 MAT=U IFM=1 EL=E_L, E_T, v_A, \nu_T, &
  G_A, \alpha_A, \alpha_T \ Vi= D_0, Z_0, Z_1, m, n,q \ K= \kappa_A, \kappa_T
  NM=2 MM=1 NDPT=1 MAT=U IFM=1 EL=E_L, E_T, v_A, \nu_T, &
  G_A, \alpha_A, \alpha_T \ Vi= D_0, Z_0, Z_1, m, n,q \ K= \kappa_A, \kappa_T %
\end{verbatim}
**Note:** The \( \kappa_A, \kappa_T \) data is only required if the \(*\text{COND}\) keyword has been used; where \( \kappa_A, \kappa_T \) are the axial and transverse thermal conductivities, respectively.

**Note:** Required Material constants for each model is as follows:

<table>
<thead>
<tr>
<th>Model</th>
<th>( ncmd )</th>
<th>Elastic</th>
<th>Inelastic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bodner-Partom</td>
<td>1</td>
<td>( \text{EL}=E_L, E_T, \nu_A, \nu_T, G_A, \alpha_A, \alpha_T )</td>
<td>( \text{VL}=D_0, Z_0, Z_1, m, n, q )</td>
</tr>
<tr>
<td>Modified Bodner-Partom</td>
<td>2</td>
<td>( \text{EL}=E_L, E_T, \nu_A, \nu_T, G_A, \alpha_A, \alpha_T )</td>
<td>( \text{VL}=D_0, Z_0, Z_1, Z_2, Z_3, m_1, m_2, n, a_1, a_2, r_1, r_2, D_{m1}, D_{m2} )</td>
</tr>
<tr>
<td>Robinson Viscoplastic</td>
<td>3</td>
<td>( \text{EL}=E_L, E_T, \nu_A, \nu_T, G_A, \alpha_A, \alpha_T )</td>
<td>( \text{VL}=n, m, \mu, \kappa_T, \beta, R, H, \hat{G}_0, \eta, \omega )</td>
</tr>
<tr>
<td>GVIPS</td>
<td>4</td>
<td>( \text{EL}=E, \nu, \alpha )</td>
<td>( \text{VL}=\mu, \kappa, R_\alpha, R_\kappa, B_0, B_1, L_0, L_1, m, n, p, q, w, z_0 )</td>
</tr>
<tr>
<td>Elastic Model</td>
<td>6</td>
<td>( \text{EL}=E_L, E_T, \nu_A, \nu_T, G_A, \alpha_A, \alpha_T )</td>
<td>( \text{VL}=\kappa, n, \mu, m, \beta, R, H, G_0, \omega, \eta )</td>
</tr>
<tr>
<td>TGVIPS</td>
<td>7</td>
<td>( \text{EL}=E_L, E_T, \nu_A, \nu_T, G_A, \alpha_A, \alpha_T )</td>
<td>( \text{VL}=\kappa, n, \mu, m, \beta, R, H, G_0, \omega, \eta )</td>
</tr>
<tr>
<td>Transversely Isotropic Elastic Model</td>
<td>9</td>
<td>( \text{EL}=E_L, E_T, \nu_A, \nu_T, G_A, \alpha_A, \alpha_T )</td>
<td>( \text{D}=d_1, d_2, d_3 )</td>
</tr>
</tbody>
</table>
Table II. MAC/GMC Material Constant Database

<table>
<thead>
<tr>
<th>Model</th>
<th>Material</th>
<th>Temperature Dependent?</th>
<th>Units</th>
<th>mat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bodner-Partom</td>
<td>Aluminum (2024-T4)</td>
<td>Yes</td>
<td>Pa, sec, °C</td>
<td>A</td>
</tr>
<tr>
<td>ncmd = 1</td>
<td>Aluminum (2024-0)</td>
<td>Yes</td>
<td>Pa, sec, °C</td>
<td>B</td>
</tr>
<tr>
<td>all properties taken from ref [1]</td>
<td>Aluminum (6061-0a)</td>
<td>Yes</td>
<td>Pa, sec, °C</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>Aluminum (6061-0b)</td>
<td>Yes</td>
<td>Pa, sec, °C</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td>Aluminum (pure)</td>
<td>No</td>
<td>Pa, sec, °C</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>Titanium (pure)</td>
<td>No</td>
<td>Pa, sec, °C</td>
<td>F</td>
</tr>
<tr>
<td></td>
<td>Copper (pure)</td>
<td>No</td>
<td>Pa, sec, °C</td>
<td>G</td>
</tr>
<tr>
<td>Modified Bodner-Partom</td>
<td>TIMETAL 21S</td>
<td>Yes</td>
<td>Pa, sec, °C</td>
<td>A</td>
</tr>
<tr>
<td>ncmd = 2</td>
<td>TIMETAL 21S</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Robinson Visco-plastic</td>
<td>Kanthal</td>
<td>No, 600°C</td>
<td>ksi, hr, °C</td>
<td>A</td>
</tr>
<tr>
<td>ncmd = 3</td>
<td>FeCrAlY</td>
<td>Yes</td>
<td>ksi, hr, °C</td>
<td>B</td>
</tr>
<tr>
<td>W/Kanalthal (vf=35%)</td>
<td>No, 600°C</td>
<td>Yes</td>
<td>ksi, hr, °C</td>
<td>C</td>
</tr>
<tr>
<td>GVIPS</td>
<td>TIMETAL 21S</td>
<td>Yes</td>
<td>ksi, sec, °C</td>
<td>A</td>
</tr>
<tr>
<td>ncmd = 4</td>
<td>TIMETAL 21S</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Linear Elastic</td>
<td>Boron</td>
<td>No</td>
<td>Pa, °C</td>
<td>A</td>
</tr>
<tr>
<td>ncmd = 6</td>
<td>SCS-6</td>
<td>Yes</td>
<td>Pa, °C</td>
<td>B</td>
</tr>
<tr>
<td>All properties are</td>
<td>Tungsten (W)</td>
<td>No</td>
<td>Pa, °C</td>
<td>C</td>
</tr>
<tr>
<td>assumed isotropic</td>
<td>Boron</td>
<td>No</td>
<td>ksi, °C</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td>SCS-6</td>
<td>Yes</td>
<td>ksi, °C</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>Tungsten (W)</td>
<td>No</td>
<td>ksi, °C</td>
<td>F</td>
</tr>
<tr>
<td>TGVIPS</td>
<td>Ti-6-4</td>
<td>Yes</td>
<td>ksi, sec, °C</td>
<td>A</td>
</tr>
<tr>
<td>ncmd = 7</td>
<td>Ti-6-4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Linear Elastic</td>
<td>T50 Graphite</td>
<td>No</td>
<td>Pa, °C</td>
<td>A</td>
</tr>
<tr>
<td>ncmd = 9</td>
<td>T300 Graphite</td>
<td>No</td>
<td>Pa, °C</td>
<td>B</td>
</tr>
<tr>
<td>Transversely Isotropic</td>
<td>P100 Graphite</td>
<td>Yes</td>
<td>Pa, °C</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>T50 Graphite</td>
<td>No</td>
<td>ksi, °C</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td>T300 Graphite</td>
<td>No</td>
<td>ksi, °C</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>P100 Graphite</td>
<td>Yes</td>
<td>ksi, °C</td>
<td>F</td>
</tr>
</tbody>
</table>

**Note: Warning:** It is the user's responsibility to ensure that consistent material property units are being employed within a given problem. Particularly, when mixing database and user supplied material properties.
Note: Even if a material model is temperature independent, it can still be used in a nonisothermal analysis (ndpt=2). Its properties will just not vary with temperature.

Note: Required Format for User Supplied Non-Isothermal Material Constants: each of the following data statements are on separate lines.

NTP= ntpts
TEM= T1, T2, ..., Tntpts
EA= E_AT1, E_AT2, ... E_ATntpts
ET= E_TT1, E_TT2, ... E_TTntpts
NUA= v_AT1, v_AT2, ... v_ATntpts
NUT= v_TT1, v_TT2, ... v_TTntpts
GA= G_AT1, G_AT2, ... G_ATntpts
ALPA= α_AT1, α_AT2, ... α_ATntpts
ALPT= α_TT1, α_TT2, ... α_TTntpts
V1= V1T1, V1T2, ... V1Tntpts
V2= V2T1, V2T2, ... V2Tntpts
...
VN= VN_T1, VN_T2, ... VN_Tntpts
D = d1, d2, d3

Note: The total number of viscoplastic constants (V1, V2, V3, ... VN) required for each model are described on the bottom of page 71. For ncmd=4, three extra lines (V15= κo V16= Bo V17= βo) must be added. Also, D= d1, d2, d3 is only required when ncmd= 3, 7 or 9.

Note: Format for User defined material model (ncmd=99):

Given User Supplied Isothermal Material Constants, the following special format is required:

*FIBER
NFIBS=nfibs

The following line is to be repeated for each fiber (nfibs):
NF=nff MS=msf MF=99 NDPT=dpt NPE=npe EL=e1, e2, ..., enpe &
ALP= α_A, α_T NPV=npv VI=v1, v2, ..., vnpv K=κ_A, κ_T %
*MATRIX
NMATX=nmatx

The following line is to be repeated for each matrix (nmatx):
NM=nm m MS=ms_m MM=99 NDPT=dpt NPE=npe EL=e_1,e_2,...e_npe & ALP= _A, _t NPV=npv VI=v_1,v_2,...v_npv K=κ_A, κ_T %

Where:
npe: - total number of elastic constants (maximum of 9)
npv: - total number of inelastic constants (model specific, max of 19)
e_1,e_2,... - elastic constants
v_1,v_2,... - inelastic constants (model specific)
α_A,α_T,... - longitudinal and transverse thermal expansion coefficients
κ_A,κ_T,... - thermal conductivities (if *COND only)

OR given User Supplied Non-Isothermal Material Constants:

*FIBER
NFIBS=nfibs

The following line is to be repeated for each fiber (nfibs):
NF=nf M=ms_f MF=99 NDPT=2 MAT= U IFM= 1
NPE=npe NPV=npv

with the following data statements immediately following each material declaration on a separate line.

NTP= ntpts
TEM= T_1, T_2, ..., T_{ntpts}
E1= E_1T_1, E_1T_2, ..., E_1T_{ntpts}
E2= E_2T_1, E_2T_2, ..., E_2T
... Enpe= E_{npeT_1}, E_{npeT_2}, ..., E_{npeT_{ntpts}}
ALPA= α_AT_1, α_AT_2, ..., α_AT_{ntpts}
ALPT= α_TT_1, α_TT_2, ..., α_TT_{ntpts}
V1= V_1T_1, V_1T_2, ..., V_1T_{ntpts}
V2 = \( V_{2T_1}, V_{2T_2}, \ldots V_{2T_n} \)

\( \vdots \)

Vnpv = \( V_{npvT_1}, V_{npvT_2}, \ldots V_{npvT_{npts}} \)

KA = \( \kappa_{AT_1}, \kappa_{AT_2}, \ldots \kappa_{AT_{npts}} \)

KT = \( \kappa_{TT_1}, \kappa_{TT_2}, \ldots \kappa_{TT_{npts}} \)

*MATRIX

NMATX = nmatx

The following line is to be repeated for each matrix (nmatx):

NM = nm_m MS = ms_m MM = 99 NDPT = 2 MAT = U IFM = 1

NPE = npe NPV = npv

with the following data statements immediately following each material declaration on a separate line.

NTP = ntpts

TEM = \( T_{T_1}, T_{T_2}, \ldots, T_{ntpts} \)

E1 = \( E_{1T_1}, E_{1T_2}, \ldots E_{1T_{npts}} \)

E2 = \( E_{2T_1}, E_{2T_2}, \ldots E_{2T_{npts}} \)

\( \vdots \)

Enpe = \( E_{npeT_1}, E_{npeT_2}, \ldots E_{npeT_{npts}} \)

ALPA = \( \alpha_{AT_1}, \alpha_{AT_2}, \ldots \alpha_{AT_{npts}} \)

ALPT = \( \alpha_{TT_1}, \alpha_{TT_2}, \ldots \alpha_{TT_{npts}} \)

V1 = \( V_{1T_1}, V_{1T_2}, \ldots V_{1T_{npts}} \)

V2 = \( V_{2T_1}, V_{2T_2}, \ldots V_{2T_{npts}} \)

\( \vdots \)

Vnpv = \( V_{npvT_1}, V_{npvT_2}, \ldots V_{npvT_{npts}} \)

KA = \( \kappa_{AT_1}, \kappa_{AT_2}, \ldots \kappa_{AT_{npts}} \)

KT = \( \kappa_{TT_1}, \kappa_{TT_2}, \ldots \kappa_{TT_{npts}} \)
**Note:** Required material user subroutines; see Example K for details on how they integrate together

**USRFUN:** - Routine that defines functional form for material properties
  Required when: ifm=2 and mat=U

**USRFORMDE:** - Routine that forms the stiffness matrix. Required when ncmd = 99

**USRCPEVAL:** - Routine that defines the time derivative of the stiffness matrix. Required when:
  a) ifm=2 and mat=U
  b) ncmd=99 AND ifm=1 AND ndpt=2
4.2.13 RVE Data:

Purpose: Select RVE representing desired fiber packing arrangement/architecture.

*MRVE
IDP=idp

For Double Periodic RVE (i.e., MOD=1):

**Note:** The following data is entered on the same line as IDP (except where noted)

Without Interface:

- For IDP = 0
  no further data required (only used for a monolithic layer in a laminate)
- For IDP = 1, 2, or 3
  VF=vf
- For IDP = 4
  VF=vf  XA=xa
- For IDP = 6
  VF=vf  R=R
- For IDP = 7
  VF=vf  R=R
- For IDP = 9
  VF1=vf1  RAD1=rad1  VF2=vf2  RAD2=rad2  R=R
- For IDP = 11
  VF=vf  RAD=rad1  R=R
- For IDP = 13
  VF=vf  R=R

With Interface:

- For IDP = 1, 2, or 3
  VF=vf  RAD=rad1  CPER=cper
- For IDP = 4
  **Currently NOT Available**
- For IDP = 6
  VF=vf  R=R  CPER=cper
- For IDP = 7
  **Currently NOT Available**
- For IDP = 9
  VF1=vl1 RAD1=rad1 CPER1=cper1 VF2=vl2 &
  RAD2=rad2 CPER2=cper2 R=R
- IDP = 11
  VF=vl RAD=rad1 R=R CPER=cper
- For IDP = 13
  * Currently NOT Available
%

where:

vf, vl, v,f2 = the fiber volume ratios
rad, rad1, rad2 = fiber radii
cper, cperl, cper2 = ratios of interface thickness to fiber radius
xa = length of the cross, see Fig. 10.

R = X/Y which defines the ratio of distances between fiber centers within a
"ply" and those between a "ply" (see Fig. 11)

* Note: For the laminate option the RVE data has the following input format:

*MRVE
  IDP=idp11, idp12, ... idp nly
  L=1 VF= ... (rve data for layer 1)
  L=2 VF= ... (rve data for layer 2)
  ...
  L=nly VF= ... (rve data for layer nly)
%

Where:

idp:- unique identifying number of each Double Periodic RVE as given below

idp                      Description
- 1 = Square Fiber, Square Pack (original 4-cell model)
    \[ V_f \leq 1/(1 + \Delta)^2 \]
    RVE shown in Fig. 9
<table>
<thead>
<tr>
<th>idp</th>
<th>Description</th>
<th>$V_f \leq \frac{0.86602}{(1 + \Delta)^2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Square Fiber, Triangular (hexagonal) Pack; RVE shown in Fig. 9</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Square Fiber, Square Diagonal Pack; RVE shown in Fig. 9</td>
<td></td>
</tr>
</tbody>
</table>

Figure 9: RVE's Available in MAC/GMC
Figure 9 cont: RVE's Available in MAC/GMC
idp | Description
--- | ---
4 | Cross Shaped Fiber, Square Pack
\[ V_f \leq 1 - 4(xa)^2 \]
RVE shown in Fig. 9
6 | 7x7 Circular Fiber Approximation Rectangular or Square Pack
\[ V_f \leq \frac{0.8125}{R(1 + \Delta)^2} \quad \text{if} \quad R > 1.0 \]
\[ V_f \leq \frac{R(0.8125)}{(1 + \Delta)^2} \quad \text{if} \quad R < 1.0 \]
RVE shown in Fig. 9
7 | 14x14 Circular Fiber Approximation Rectangular or Square Pack
\[ V_f \leq \frac{0.8148}{R} \quad \text{if} \quad R > 1.0 \]
\[ V_f \leq R(0.8148) \quad \text{if} \quad R < 1.0 \]
RVE shown in Fig. 9
9 | Two Different Size Square Fibers, Rectangular or Square Pack
\[ V_{f_2} \leq \frac{1}{R \left( (1 + \Delta_2) + (1 + \Delta_1) \frac{R_{f_1}}{R_{f_2}} \right)^2} \]
\[ V_{f_1} \leq \frac{2 \sqrt{R V_{f_2}}}{\left[ \left( \frac{R_{f_2}}{R_{f_1}} \right)^2 + \left( \frac{R_{f_2}}{R_{f_1}} \right)^2 (1 + \Delta_2) + \left( \frac{R_{f_2}}{R_{f_1}} \right)^2 (1 + \Delta_1) \right]} \]
\[ \frac{V_{f_1}}{V_{f_2}} \sqrt{\frac{R_{f_1}}{R_{f_2}}} \geq 1 \]
RVE shown in Fig. 9

Note: Two fibers and two interfaces must be defined in *Fiber and *Interface.

Note: \( V_{f_1} \) will be altered, if necessary, such that an integer number of small fibers fit within the RVE.
idp | Description
---|---
11 | Square Fiber, Rectangular Pack
\[
V_f \leq \frac{1}{R(1 + \Delta)^2} \quad \text{if} \quad R > 1.0
\]
\[
V_f \leq \frac{R}{(1 + \Delta)^2} \quad \text{if} \quad R < 1.0
\]
RVE shown in Fig. 9

13 | 26 x 26 Circular Fiber, Rectangular or Square Pack
\[
V_f \leq \frac{0.80613}{R} \quad \text{if} \quad R > 1.0
\]
\[
V_f \leq 0.80613R \quad \text{if} \quad R < 1.0
\]
RVE shown in Fig. 9

*Note: \( R_{f1} = \text{rad1}, R_{f2} = \text{rad2} \) and if no interface is present, \( \Delta = \text{cper}, \Delta_1 = \text{cper1} \) and \( \Delta_2 = \text{cper2} \) should be taken to be zero, in the above formulas.

99 | User Defined RVE
Example of RVE representing random packing shown in Fig. 12. Required input shown in Example I.

Example: Triangular packing with interface thickness 1% of fiber radius.

*MRVE
IDP=2 VF=35 RAD=0.07 CPER=0.01 %
Figure 10: Cross Shaped Fiber Distance xa

Figure 11: Hybrid Composite RVE; IDP = 9. Large Fiber Spacing Ratio, R=\(X/Y\)
Figure 12: User defined RVE (see Example I)
For Triple Periodic RVE (i.e., MOD=2):

**Note:** The following data is entered on the same line as the IDP data, except where noted.

Thus for

- For IDP = 1
  
  VF=vf
  
  ASP=asp

- For IDP = 2
  
  VF=vf
  
  ASP1=asp1
  
  ASP2=asp2

- For IDP = 3
  
  VF=vf
  
  ASP1=asp1
  
  ASP2=asp2
  
  DR=dr

where,

- *asp, asp1* - are the aspect ratio of the short fiber (i.e., fiber length/fiber diameter; e.g. in IDP=3 this is \((d_1 + d_2 + d_3)/h_1)\)
- *asp2* - is the aspect ratio of the unit cell, \(d/h\) (see Fig. 2)
- *dr* - is the ratio \(d_1/d_3\) which quantifies the fiber off-set (see Fig. 13)

- For IDP = 4
  
  One of the following lines should be entered after the IDP data depending upon the option desired

  - OPT=1
    
    A=a
    
    B=b
    
    C=c
    
    D=d
    
    H=h
    
    L=l
    
    %

  - OPT=2
    
    VF=vf
    
    A=a
    
    B=b
    
    C=c
    
    RD=rd
    
    RL=rl
    
    %

  - OPT=3
    
    VF=vf
    
    RA=ra
    
    RC=rc
    
    D=d
    
    H=h
    
    L=l
    
    %

  - OPT=4
    
    VF=vf
    
    A=a
    
    B=b
    
    C=c
    
    RD=rd
    
    RLC=rlc
    
    %

where,

- *a, b, and c* are the ellipsoid's semi-major axes (see Fig. 13)
- *d, h, and l* are the unit cell dimensions (see Fig. 13)
- *rd* = aspect ratio of \(d/h\)
- *rl* = aspect ratio of \(l/h\)
- *ra* = aspect ratio of \(a/b\)
- *rc* = aspect ratio of \(c/b\)
- *rlc* = aspect ratio of \(l/(2c)\)

- For IDP = 10
  
  VF=vf

Where

\(idp\):- unique identifying number of each **Triple Periodic** RVE given Fig. 13
**Description**

- **1 =** Short fibers in Square array, with equal spacing in all directions (i.e., \( d_2 = h_2 = l_2 \) see Fig. 13)
  
  \[ 0 \leq V_f \leq 1 \]
  
  RVE shown in Fig. 13

- **2 =** Short fibers in Diagonal array, with variable inclusion spacing in the \( x_1 \) - direction
  
  \[ V_f < \frac{1}{2} \left( \frac{Asp_1}{Asp_2} \right) \quad \text{if} \quad Asp_1 < Asp_2 \]
  
  \[ V_f > \frac{1}{2} \left( \frac{Asp_2}{Asp_1} \right) \quad \text{if} \quad Asp_1 > Asp_2 \]
  
  RVE shown in Fig. 13

- **3 =** Off-set short fibers in square array, variable fiber spacing in the \( x_1 \) - direction --- NOTE \( Asp_1 > Asp_2 \)
  
  \[ V_f < 4 \left( \frac{Asp_1}{Asp_2} \right)^2 \quad \text{if} \quad \frac{Asp_1}{Asp_2} > 2 \]
  
  \[ V_f < \frac{1}{2} \left( \frac{Asp_2}{Asp_1} \right) \quad \text{if} \quad \frac{Asp_1}{Asp_2} < 2 \]
  
  \[ V_f > \frac{1}{2} \left( \frac{Asp_2}{Asp_1} \right)^2 \]
  
  RVE shown in Fig. 13

- **4 =** Ellipsoidal Inclusion

**Note:** Analytical expressions for \( V_f \) limits are unavailable as forming this unit cell geometry requires solution of non-linear equations.

**Note:** Use of non-physical ellipsoid unit cell dimensions (e.g., \( 2b > h \)) will cause execution of MAC/GMC to stop.

- **10 =** Open cell
  
  \[ 0 \leq V_f \leq 1 \]

**Note:** Solid material (see Fig. 13) is the matrix material specified in *MATRIX. whereas, the "open" material is the fiber material specified in *FIBER. To simulate a truly "open cell" material, specify a fiber material with the properties of air, thus fiber volume fraction becomes the void fraction

- **99 =** User defined RVE

86
Note: The following is the format for the user defined RVE's, IDP = 99 (see Fig. 12 for an example).

1) For 2-D RVE: (Each line of data must be on a separate line)

NB = nb  NG = ng
H = h₁, h₂, ..., hng
L = l₁, l₂, ..., lnb
CM = ss₁,₁, ss₁,₂, ..., ss₁,ng

repeat data for the (nb x ng) 2-D RVE

CM = ss₁,₁, ss₁,₂, ..., ss₁,ng

2) For 3-D RVE: (Each line of data must be on a separate line)

NA = na  NB = nb  NG = ng
D = d₁, d₂, ..., dna
H = h₁, h₂, ..., hng
L = l₁, l₂, ..., lnb
CM = ss₁,₁, ss₁,₂, ..., ss₁,ng
CM = ss₁,₁, ss₁,₂, ..., ss₁,ng

repeat data for the (na x nb x ng) 3-D RVE

CM = ss₁,₁, ngss₁,₂, ng..., ss₁, nb, ng

Where:

ss₁,₁ and ss₁,₁, k - are the identifying material labels which are given in the following format:

first character: F - for a fiber or M - for matrix
second character: 1, 2, 3, ... for fiber/matrix number 1, 2, 3 ...

na - number of subcells in the alpha direction
nb - number of subcells in the beta direction
ng - number of subcells in the gamma direction
h - the height of each subcell (x₂-dimension)
l - the length of each subcell (x₃-dimension)
d - the depth of each subcell (x₁-dimension)
Figure 13 Available 3-D RVE's
Ellipsoidal Inclusion RVE (IDP = 4)

Figure 13 con't: Available 3-D RVE's
IDP=10 - Open cell

Figure 13 con't: Available 3-D RVE's
**Note:** A 2-D (continuous reinforcement) and 3-D (discontinuous reinforcement) example are described next with a detailed example of the user defined inputs being found in Example I, 2-D case, and in Examples H and N, for the 3-D case.

![Figure 14](image)

**Example:** 2-D user defined RVE (as shown above)

```
*MRVE
IDP=99
NB=3 NG=3
H=1.,1.,1.
L=1.,1.,1.
CM=M1,M1,M1
CM=M1,M1,F1
CM=F1,M1,M1 %
```

**Note:** the fiber ID is F1 and the matrix ID is M1
Example: A 3-D User defined RVE (as shown above), representing a [0/90] continuous reinforced laminate composite.

Note: The fiber is material number F1 and the matrix is material number M1.
**MRVE**

IDP=99  ASP=1.
NA=4  NB=2  NG=2
D=1.,1.,1.,1.
H=1.,1.
L=1.,1.
CM=M1,M1
CM=M1,M1
CM=M1,M1
CM=F1,M1
CM=M1,M1
CM=F1,F1
CM=M1,M1
CM=F1,M1  

rows 1-4 are for cross-section 1 (γ=1)

rows 5-8 are for cross-section 2 (γ=2)
4.2.14 Interface Data: (Optional, for IDP=1, 2, 3, 6 or 11)

Purpose: Specify interface layer properties

*INTERFACE

NINT=nint

The following line is to be repeated for each interface(nint):
NL=nm_i MS=ms_i ML=ncmd_i NDPT=dpt TEMP=mtemp MAT=mat_i IFM=ifm_i D=d_1,d_2,d_3

% Note: If MAT= U, see section 4.2.12 for format of additional input required.

Where

nint: - number of different interfaces

nm_i: - i_th interface number

ms_i: - matrix material system ID. (required only when using laminate option)

ncmd_i: - material model identifier for the i_th interface

mat_i: - material id letter for the i_th interface

dpt: - flag indicating whether material constants should be temperature independent or temperature dependent
1 = Temperature Independent
2 = Temperature Dependent

mtemp: - the constant temperature at which material properties are to be taken
(only required for dpt= 1 and when using database properties)

ifm - flag indicating whether material properties will be read from input file or taken from a user defined function (provided in the USRFUN subroutine)

ONLY NEEDED when MAT=U.
1 = read from input file
2 = functional form taken from USRFUN routine

d_i: - direction vector defining the normal to the plane of local isotropy
(only required for ncmd = 3, 7 or 9)

Note: Please refer to section 4.2.12 for ncmd and mat values

Example: (Assuming one interface with user supplied material constants)

*INTERFACE

NINT=1
NL=1 ML=4 NDPT=1 MAT=U IFM=1 &
EL=11700.,0.365,1.0E-06 &
VI=0.8E-8,0.1,0.1E-5,0.,0.85E-3,0.05,1.,1.,1.,3.3,1.8,1.35,1.,0.01
4.2.15 Debond Data: (Optional)

Purpose: Specify the subcell faces where debonding can take place (see section 3.8)

2-D GMC Model Format

*DEBOND

# Option 1

DBCH= 1 NBl=nbi NGl=ngi FACE=nfc RN=rn BDN=bdstrn RS=rs & BDS=bdstrs TI=tmd

OR

# Option 2

DBCH= 2 NBl=nbi NGl=ngi FACE=nfc BDN=bdstrn GCN=gamman & BCN= betan TOLN=toln BDS= bdstrs GCS=gammas BCS=betas TI=tmd

: repeat nii times

%

3-D GMC Model Format

*DEBOND

# Option 1

DBCH= 1 NAl=nai NBl=nbi NGI=ngi FACE=nfc RN=rn BDN=bdstrn & RS=rs BDS=bdstrs TI=tmd

OR

# Option 2

DBCH= 2 NAl=nai NBl=nbi NGI=ngi FACE=nfc BDN=bdstrn & GCN=gamman BCN= betan TOLN=toln BDS= bdstrs GCS=gammas & BCS=betas TI=tmd

%

Laminate Model Format

* Note: Currently this is only available for continuous reinforcements, modid=1 see section 4.2.8

*DEBOND

L=1 NII=nii (data for layer 1)
see 2-D GMC format above --- for required input here
L=2 NII=nii (data for layer 2)
see 2-D GMC format above --- for required input here
:
L=nly NII=nii (data for layer nly)
see 2-D GMC format above --- for required input here
%

95
Where:

- \( n_{ii} \): number of subcell interfaces with debonding
- \( dbch \): debond model implementation choice, =1 (for first) or = 2 (for second), see section 3.8
- \( nai, nbi \) and \( ngi \): subcell indices (\( \alpha, \beta, \gamma \))
- \( nfc \): used specify which "face" of the subcell is to be debonded (See Fig. 17 and 18)

For 2-D GMC
- = 1 Top Face
- = 2 Right Face
- = 3 Both Faces

For 3-D GMC
- = 1 Top Face
- = 2 Right Face
- = 3 Back Face
- = 4 Top & Right Face
- = 5 Top & Back Face
- = 6 Right & Back Face
- = 7 ALL Faces

- \( r_n \): \( R_n \) value (see section 3.8)
- \( bdstrm \): debond stress, normal component
- \( rs \): \( R_s \) value (see section 3.8)
- \( bdstrs \): debond stress, shear component
- \( tmd \): time at which debond criteria becomes active (defaults to \( tmd=0 \), if not specified)
- \( gamman \): \( \Lambda_n \) value (see section 3.8)
- \( gammas \): \( \Lambda_s \) value (see section 3.8)
- \( betan \): \( B_n \) value (see section 3.8)
- \( betas \): \( B_s \) value (see section 3.8)
- \( toln \): Normal stress reversal tolerance for unloading (i.e., stress value that is low enough to treat as zero stress - interface has closed)

**Note:** For the Laminate Model Only: \( n_{ii}=0 \) specifies NO debonding for that layer and no additional debond data is required.

**Note:** All debond parameters must be calibrated for a given composite system. **Example M** provides some initial estimates for an SCS-6/Ti system.
Example 1:
allow debonding of subcell 1,1 top face normal component starting at
time = 11
NII=1
DBCH= 1 NBI=1 NGI=1 FACE=1 RN=1. BDN=100. RS=0. BDS=0. TI= 11.

Example 2:
allow debonding of subcell 1,1 right face shear component starting at
time = 15
NII=1
DBCH= 1 NBI=1 NGI=1 FACE=2 RN=0. BDN=0. RS=1. BDS=100. TI= 15.

Example 3: Three Layer Laminate Analysis
debond subcell 1,1 right face shear component for layer 2 only
L=1 NII=0 (no debonding specified)
L=2 NII=1
DBCH=1 NBI=1 NGI=1 FACE=2 RS=1. BDN=100.
L=3 NII=0 (no debonding specified)

Figure 17: Subcell Faces For 2-D GMC
Figure 18: Subcell Faces For 3-D GMC
4.2.16 Plot Point Information:

Purpose: Specify the frequency at which data will be written to output files for both X-Y data

*CURVE
   NP=nmax
%

Where:

nmax: - plot point increment

Example: (print out every fifth data point)

*CURVE
   NP=5   %
4.2.17 Curve Data:

**Note:** Currently a maximum of 5 macro and 5 micro curves may be specified per problem.

**Specifying Curve Data For Macro (composite) Quantities:**

```
*MACRO
  NT=nucuv

Repeat the following line nucuv times:
  NC=nocu X=maidx Y=maidy NAM=tname
%
```

Where:

- `nucuv` - total number of curves
- `nocu` - curve number

`maidx` and `maidy` variable options are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>e_{11}</code></td>
</tr>
<tr>
<td>2</td>
<td><code>e_{22}</code></td>
</tr>
<tr>
<td>3</td>
<td><code>e_{33}</code></td>
</tr>
<tr>
<td>4</td>
<td><code>\gamma_{23}</code></td>
</tr>
<tr>
<td>5</td>
<td><code>\gamma_{13}</code></td>
</tr>
<tr>
<td>6</td>
<td><code>\gamma_{12}</code></td>
</tr>
<tr>
<td>7</td>
<td><code>\sigma_{11}</code></td>
</tr>
<tr>
<td>8</td>
<td><code>\sigma_{22}</code></td>
</tr>
<tr>
<td>9</td>
<td><code>\sigma_{33}</code></td>
</tr>
<tr>
<td>10</td>
<td><code>\tau_{23}</code></td>
</tr>
<tr>
<td>11</td>
<td><code>\tau_{13}</code></td>
</tr>
<tr>
<td>12</td>
<td><code>\tau_{12}</code></td>
</tr>
<tr>
<td>13</td>
<td><code>\varepsilon^{i}_{11}</code></td>
</tr>
<tr>
<td>14</td>
<td><code>\varepsilon^{i}_{22}</code></td>
</tr>
<tr>
<td>15</td>
<td><code>\varepsilon^{i}_{33}</code></td>
</tr>
<tr>
<td>16</td>
<td><code>\varepsilon^{i}_{23}</code></td>
</tr>
<tr>
<td>17</td>
<td><code>\varepsilon^{i}_{13}</code></td>
</tr>
<tr>
<td>18</td>
<td><code>\varepsilon^{i}_{12}</code></td>
</tr>
<tr>
<td>19</td>
<td>n.a.</td>
</tr>
<tr>
<td>30</td>
<td><code>\gamma_{13}^{th}</code></td>
</tr>
<tr>
<td>31</td>
<td><code>\gamma_{22}^{th}</code></td>
</tr>
<tr>
<td>32</td>
<td><code>\gamma_{33}^{th}</code></td>
</tr>
<tr>
<td>33</td>
<td><code>\gamma_{23}^{th}</code></td>
</tr>
<tr>
<td>34</td>
<td><code>\gamma_{13}^{th}</code></td>
</tr>
<tr>
<td>35</td>
<td><code>\gamma_{12}^{th}</code></td>
</tr>
<tr>
<td>36</td>
<td><code>\gamma_{13}</code></td>
</tr>
<tr>
<td>37</td>
<td>Total Time</td>
</tr>
<tr>
<td>38</td>
<td>Creep Time</td>
</tr>
<tr>
<td>39</td>
<td>Temperature</td>
</tr>
</tbody>
</table>

- `tname`: - name of plot file

**Note:** The file(s) generated will be of the form `tname_macro.data`. If the user desires to use the same `tname` for files 2-5, a double quote, "", is entered for `tname`. Those files then will have the form:

```
tname2_macro.data
tname3_macro.data
e.tc.
```

(see the example on the next page for more details)

**FEAMAC Note:** The file(s) generated when running FEAMAC will be of the form `tname.element#.integrationpt#.feamacro.data`, see section 4.1.2.
EXAMPLE:

*MACRO
NT=3
NC=1  X=1  Y=7  NAM=stress  file created: stress_macro.data
NC=2  X=37 Y=13  NAM=""  file created: stress2_macro.data
NC=3  X=38 Y=13  NAM= plot  %  file created: plot_macro.data

**Note:** For the Laminate option, the \textit{maidx} and \textit{maidy} variable options are modified as follows:

\begin{align*}
1 \ - \ &\bar{\varepsilon}_{xx} \quad 7 \ - \ \bar{N}_{XX} \quad 15 \ - \ \bar{N}_{XY} \quad 21 \ - \ 23 \ n.a. \quad 31 \ - \ \bar{N}_{XX}^T \quad 37 \ - \ \text{Total Time} \\
2 \ - \ &\bar{\varepsilon}_{yy} \quad 8 \ - \ \bar{N}_{YY} \quad 16 \ - \ \bar{M}_{XX} \quad 24 \ - \ \bar{\kappa}_{xy} \quad 32 \ - \ \bar{N}_{YY}^T \quad 38 \ - \ \text{Creep Time} \\
3 \ - \ &\bar{\varepsilon}_{33} \quad 9 \ - \ 11 \ n.a. \quad 17 \ - \ \bar{M}_{YY} \quad 25 \ - \ \bar{M}_{XX} \quad 33 \ - \ \bar{N}_{XY}^T \quad 39 \ - \ \text{Temperature} \\
4 \ - \ &\text{n.a.} \quad 12 \ - \ \bar{N}_{XY} \quad 18 \ - \ \bar{M}_{XY}^T \quad 26 \ - \ \bar{M}_{YY} \quad 34 \ - \ \bar{M}_{XX}^T \\
5 \ - \ &\text{n.a.} \quad 13 \ - \ \bar{N}_{XX}^T \quad 19 \ - \ \bar{\kappa}_{xx} \quad 27 \ - \ 29 \ n.a. \quad 35 \ - \ \bar{M}_{YY}^T \\
6 \ - \ &\bar{\gamma}_{xy} \quad 14 \ - \ \bar{N}_{YY}^T \quad 20 \ - \ \bar{\kappa}_{yy} \quad 30 \ - \ \bar{M}_{XY} \quad 36 \ - \ \bar{M}_{XY}^T
\end{align*}

**Note:** When using laminate option, resultant forces (\(\bar{N}_{XX}, \bar{N}_{YY}, \text{or} \ \bar{N}_{XY}\)) are output in place of stress components. To obtain associated stresses one must merely divide resultant force by overall laminate thickness. Also, the strains are \textit{mid-plane strains} while \(\bar{\varepsilon}_{33}\) represents the average strain through the thickness.

**Specifying Curve Data for Micro (subcell) Quantities:**

*MICRO
NT=\text{nucuv}

Repeat the following line \text{nucuv} times:
\begin{verbatim}
  NC=\text{nucu}  LYR=\text{lyr}  CELL=\text{nsse}  X=\text{maidx}  Y=\text{maidy}  NAM=\text{tname2}  \\
% 
\end{verbatim}

Where:
\begin{verbatim}
  nucuv \ - \ total \ number \ of \ curves
\end{verbatim}
nucu: - curve number

lyr: - layer number (only required when using laminate option)

nsset: - subcell number (see next page for details on numbering)

tname2: - name of plot file

maidx and maidy variable options are:

1 - $e_{11}$  7 - $\sigma_{11}$  13 - $\varepsilon_{11}^i$  19 - $\Phi_{11}$  25 - $\Psi_{11}$  31 - $\varepsilon_{11}^{th}$  37 - Total Time
2 - $e_{22}$  8 - $\sigma_{22}$  14 - $\varepsilon_{22}^i$  20 - $\Phi_{22}$  26 - $\Psi_{22}$  32 - $\varepsilon_{22}^{th}$  38 - Creep Time
3 - $e_{33}$  9 - $\sigma_{33}$  15 - $\varepsilon_{33}^i$  21 - $\Phi_{33}$  27 - $\Psi_{33}$  33 - $\varepsilon_{33}^{th}$  39 - Temperature
4 - $\gamma_{23}$  10 - $\tau_{23}$  16 - $\gamma_{23}^i$  22 - $\Phi_{23}$  28 - $\Psi_{23}$  34 - $\gamma_{23}^{th}$
5 - $\gamma_{13}$  11 - $\tau_{13}$  17 - $\gamma_{13}^i$  23 - $\Phi_{13}$  29 - $\Psi_{13}$  35 - $\gamma_{13}^{th}$
6 - $\gamma_{12}$  12 - $\tau_{12}$  18 - $\gamma_{12}^i$  24 - $\Phi_{12}$  30 - $\Psi_{12}$  36 - $\gamma_{12}^{th}$

Note: The quantities $\Phi$ and $\Psi$ are the possible internal state variables (constitutive model dependent).

Note: For the Laminate option, the data will be output for both integration points within a layer. Thus 4 rather than 2 columns of data will appear in the associated output file(s), with the first 2 columns representing the x and y quantities at the lower quadrature point and the last 2 columns those associated with the upper quadrature point (see Example P).

Note: The file(s) generated will be of the form tname2_micro.data. If the user desires to use the same tname2 for files 2-5, a double quote, ", is entered for tname2. Those files then will have the form:

```
tname22_micro.data
ntname23_micro.data
```

Note: The file(s) generated when running the laminate option will be of the form tname.l#.micro.data, where # indicates the layer number, see example below

FEAMAC Note: The file(s) generated when running FEAMAC will be of the form tname.element#.integrationpt#.feamicro.data, again see section 4.1.2.
Note: Subcell numbering is assigned according to the following algorithm.

2-D case

Do * IB=1,NB
Do * IG=1,NG
subcell number = NG*(IB-1)+IG
continue

Note: Refer to numbers in upper left corners of subcells in Fig. 14 for example of 2-D subcell numbering scheme.

3-D case

Do * IA=1,NA
Do * IB=1,NB
Do * IG=1,NG
subcell number = IA+(IB-1)*NA+(IG-1)*NA*NB
continue

Note: Refer to numbers in upper left corners of subcells in Fig. 16 for example of 3-D subcell numbering scheme.

EXAMPLE:

*MICRO
NT = 1
NC=1  CELL=1  X=1  Y=7  NAM=cell  %  file created:cell_micro.data

EXAMPLE: Assuming laminate comprised of 2 layers

*MICRO
NT = 2
NC=1  LYR=1  CELL=2  X=3  Y=2  NAM=lam-x2  %
  file created:lam-x2.11.micro.data

NC=2  LYR=2  CELL=2  X=3  Y=1  NAM=lam-x3  %
  file created:lam-x3.12.micro.data
5.0 Conclusion/Future Modifications

A computationally efficient, user-friendly, comprehensive, micromechanics analysis tool, MAC/GMC, has been presented that admits physically based viscoplastic deformation and life models, can analyze continuous or discontinuous multi-phased materials of interest in advanced propulsion systems, and can assist both the materials scientist and structural analyst in developing, designing and analyzing strategic materials. MAC/GMC’s most outstanding feature is its ability to accurately model various laminated fiber architectures (including both shape and packing arrangements) at minimal expense both from a computational and required user input standpoint. However, the development of this tool is far from complete. A number of possible future enhancements that are planned or currently underway include:

- Provision of an implicit integration algorithm to improve computational efficiency.
- The incorporation of woven composite IDP architectures.
- The ability to analyze smart composites, i.e. piezoelectric, etc.
- The addition of debond unit cells, so as to minimize user input.
- The incorporation of additional damage evolution laws and failure criteria (e.g., ultimate longitudinal and transverse tensile strength) so as to automate life estimates.
- The incorporation of viscoelastic behavior models for the analysis of elevated temperature polymeric composites.
- The ability to use the triple-periodicity model in conjunction with laminate theory allowing individual lamina to be reinforced with particulates or weaves.

5.1 Acknowledgment

The authors would like to thank Mr. Robert Goldberg and Mrs. Cheryl Bowman for their diligent efforts and assistance in checking the accuracy and consistency of this manual by running numerous test cases. Also, we would like to thank Prof. J. Aboudi for many fruitful discussions and technical assistance.
6.0 EXAMPLE PROBLEMS

It is recommended that a new user construct a MAC/GMC input file using the data given in these Examples and then check to see if the same result plots and/or files are obtained.

6.1 Example A: Pure Mechanical Load

Sample Input File For A Mechanical Load Problem

The following example is used to explain the control blocks in more detail.

Problem Summary:

Load Type: Mechanical
Load Component: 22-direction (transverse to fiber)
Load History: Monotonic
Load Control: Strain
Load History Data:
\[ \dot{\varepsilon} = 8.333 \times 10^{-4} \text{/sec}, \]
\[ \varepsilon_{\text{max}} = 0.015, \]
\[ \varepsilon_{\text{min}} = 0. \]
\[ \Delta t_{\text{initial}} = 0.0000024 \text{ sec} \]
Micromechanics model: Double Periodicity
Fiber Packing Arrangement: Hexagonal Pack at 35% fiber volume ratio
Integration Algorithm: Predictor/Corrector
Constituent Material Model: GVIPS - isotropic form
Constituents:
Fiber: SCS-6
Matrix: TIMETAL21S
Interface: fictitious weak interface for TIMETAL21S

*Note:* This example will take an extremely long time to run due to the elastic-perfectly viscoplastic definition of the fictitious weak interface, as the overall time step is limited to that of the allowable interface time step. This slow computational response is NOT indicative of GMC.
Mechanical load, no residual, perfectly viscoplastic interface, takes long time

*PRINT
   NPL=0 %
*LOAD
   LCON=2 LOP=2 LSS=1 %
*Mech
   NPTW=2 TI=0.,18. LO=0.,0.015 %
*MODEL
   MOD=1 %
*SOLVER
   NTF=2 ISTM=0.0000024 ERR=0.1E-2 %
*FIBER
   NFIBS=1
   NF=1 MF=6 NDPT=1 MAT=D TEMP=650 %
*MATRIX
   NMATX=1
   NM=1 MM=4 NDPT=1 MAT=A TEMP=650 %
*MRVE
   IDP=2 VF=0.35 RAD=0.07 CPER=0.1 %
*INTERFACE
   NINT=1
   NI=1 MI=4 NDPT=1 MAT=U IFM=1 &
   EL=11700.,0.365,1. &
   VI=0.8E-8,0.1,0.1E-5,0.,0.85E-3,0.05,1.,1.,3.3,1.8,1.35,1.,0.01 %
*CURVE
   NP=1 %
*MACRO
   NT=1
   NC=1 X=2 Y=8 NAM=apdxa %
*END
The following figure was obtained from the x-y plot data file produced by the present example.

**Note:** Using this fictitious interface model the qualitative feature of a weak bond (reduction below that of the matrix only) is obtained, however an accurate fit of the experimental data is still lacking. For a more accurate prediction of transverse behavior see Examples C and D.
6.2 Example B: Pure Thermal Problem

Sample Input File For A Thermal Load Problem

The following example is used to explain the control blocks in more detail.

Problem Summary:

Load Type: Thermal

Load History Data:

$\dot{T} = 0.01 \, ^\circ\text{C/sec}$

$T_{max} = 371.1 ^\circ\text{C}$

$T_{min} = 21.1 ^\circ\text{C}$

$\Delta t_{initial} = 17.505 \, \text{sec}$

Micromechanics model: Double Periodicity

Fiber Packing Arrangement: Square Pack at 35% fiber volume ratio

Integration Algorithm: Predictor/Corrector

 Constituent Material Model: Bodner-Partom

Constituents:

Fiber: T50 Graphite
Matrix: Aluminum (2024-T4)

*Note: This problem is taken from reference 1., pg. .238
cooldown heatup bodner model aboudi prob. pre/cor gmc2d

*NPRINT
  NPL=0  

*LOAD
  LCON=1  

*THERM
  NPTT=3  TI=0.,35010.,70020. &
  TE=371.1,21.1,371.1  

*MODEL
  MOD=1  

*SOLVER
  NTF=2  ISTT=17.505  ERR=0.001  

*FIBER
  NFIBS=1
  NF=1  MF=9  NDPT=2  MAT=A  D=1.,0.,0.

*MATRIX
  NMATX=1
  NM=1  MM=1  NDPT=2  MAT=A  

*MRVE
  IDP=1  VF=0.30  

*CURVE
  NP=5  

*MACRO
  NT=1
  NC=1  X=39  Y=1  NAM=apdxb  

*END
The following figure was obtained from the x-y plot data file produced by the present example.

\[\begin{align*}
\text{Temperature} & \quad 0 \quad 50 \quad 100 \quad 150 \quad 200 \quad 250 \quad 300 \quad 350 \quad 400 \\
\varepsilon & \quad 0.01 \quad 0.02 \quad 0.03 \quad 0.04 \quad 0.05 \quad 0.06 \quad 0.07 \quad 0.08 \quad 0.09 \\
\end{align*}\]

\[\text{Forward Euler} \quad \circ \quad \text{Predictor/Corrector}\]

**Note:** It is recommended that a new user construct a mac input file using the data given in this Example and then check to see if the same result plot is obtained.
6.3 Example C: Thermomechanical Load Problem

Sample Input File For A Thermomechanical Load Problem

The following example is used to explain how to impose thermal residual stresses due to manufacturing into mechanical analysis.

Problem Summary:

Load Type: Thermomechanical
Load Component: 33-direction (transverse to fiber)
Load History: Cyclic
Load Control: Strain
Load History Data: Cool-down from 900 °C to 23 °C then hold temperature constant during mechanical loading

\[ \dot{T} = 1.52 \times 10^{-2} \degree C/sec \]

\[ \dot{\varepsilon} = 1.666 \times 10^{-4} /sec, \quad \varepsilon_{max} = 0.015, \quad \varepsilon_{min} = 0. \]

\[ \Delta t_{thermal} = 100. \text{ sec}, \quad \Delta t_{mech} = 0.1 \text{ sec} \]

Micromechanics model: Double Periodicity
Fiber Packing Arrangement: Rectangular Pack at 33% fiber volume ratio
Integration Algorithm: Forward Euler
Constituent Material Model: GVIPS - isotropic form
Constituents: Fiber: SCS-6 (properties input manually)
Matrix: TIMETAL21S
Debonding: First implementation of debond model

Note: This example demonstrates how one would included the effects of residual stresses on the composite behavior. Also, although strain control is prescribe, during a pure cool down MAC/GMC allows Poisson's strains to be incurred.
test of residual stress idp=11 23C apply transverse loading
*PRINT
  NPL=0 %
*LOAD
  LCON=3 LOP=3 LSS=1 %
#  LCON=3 LOP=1 LSS=1 %
*MECH
  NPTW=3 TI=0.,57600.,57690. LO=0.,0.,0.015 %
#  NPTW=3 TI=0.,57600.,57750. LO=0.,0.,0.015 %
*THERM
  NPTT=3 TI=0.,57600.,57690. TE=900.,23.,23. %
#  NPTT=3 TI=0.,57600.,57750. TE=900.,23.,23. %
*MODEL
  MOD=1 %
*SOLVER
  NTF=1 NPTS=3 TIM=0.,57600.,57690. STP=100.,0.1 %
#  NTF=1 NPTS=3 TIM=0.,57600.,57750. STP=100.,0.1 %
*FIBER
  NFIBS=1
#  Use GVIPS but make it elastic k=1E20, so it represents SCS-6 fiber
  NF=1 MF=4 NDPT=1 MAT=U IFM=1 &
    EL=58.3E3, 0.32, 3.5E-06 &
    VI=0.8E-9,1.E20,0.1E-5,0.,0.85E-4,0.05,1.,1.,3.3,1.8,1.35,1.,0.01 %
*MATRIX
  NMATX=1
  NM=1 MM=4 NDPT=2 MAT=A %
*MRVE
  IDP=11 VF=0.33 RAD=0.07 R=1.1111 %
*DEBOND
  NII=2
  DBCH=1 NBI=1 NGI=1 FACE=2 RN=0.1 BDN=14. TI=57600.
#  DBCH=1 NBI=1 NGI=2 FACE=2 RN=0.1 BDN=14. TI=57600. %
*CURVE
  NP=2 %
*MACRO
  NT=2
  NC=1 X=3 Y=9 NAM=apdxc-t
  NC=2 X=1 Y=7 NAM=apdxc-l
*END
The following figure was obtained from the x-y plot data file produced by the present example.

![Graph showing stress-strain relationship for different orientations and temperatures.](image)

**Note:** In order to produce the longitudinal response curve one must change the direction of loading (LOP=1 instead of 3) and the mechanical strain rate imposed, so as to agree with the imposed experimental history by commenting and uncommenting the appropriate lines under *LOAD, *MECH, *THERM, and *SOLVER.

**Note:** It is recommended that a new user construct a MAC/GMC input file using the data given in this Example and then check to see if the same resulting plot can be obtained.
6.4 Example D: Transverse Debonding

Sample Input File for Transverse Debonding

Problem Summary

Load Type: Thermomechanical

Load History Data: 33-direction (transverse to fiber)

Load History: Cyclic

Load Control: Strain

Load History Data: Cool-down from 900°C to 534°C to 23°C, heat-up to 650°C, then hold temperature during mechanical loading

\[ \dot{T} = 0.0152 \, ^\circ\text{C/sec}, \, 0.0871 \, ^\circ\text{C/sec} \]

\[ \varepsilon = 1.666 \times 10^{-4} \, /\text{sec}, \, \varepsilon_{max} = 0.018, \, \varepsilon_{min} = 0. \]

\[ \Delta t_{thermal} = 250, \, 40. \, \text{sec}; \, \Delta t_{mech} = 0.4 \, \text{sec} \]

Micromechanics Model: Double Periodicity

Fiber Packing Arrangement: Rectangular Pack, \( R = 1.1111 \), 35% fiber volume ratio

Repeating Unit Cell: 26x26 circular fiber cross-section approximation

Integration Algorithm: Forward Euler

Constituent Material Model:

Fiber: Elastic, transversely isotropic
Matrix: GVIPS - isotropic form

Constituents:

Fiber: SCS-6 (temp. dep. properties input manually)
Matrix: TIMETAL21S

Debonding: Second implementation of debond model

Debond Parameters:

\[ \sigma_{DB} = \sigma_{DB}^0 + C_1/|\text{slope}| \], \( \sigma_{DB}^0 = 7.\text{ksi}, C_1 = 15.\text{ksi} \)

\[ \Lambda = \Lambda_o - C_2/|\text{slope}| \], \( \Lambda_o = 0.06 \frac{\text{in}^3}{\text{kip}} \)

\[ C_2 = 0.03997 \frac{\text{in}^3}{\text{kip}} \quad B = 16.5 \, \text{1/sec} \]
**Note:** The debond parameters $\sigma_{DB}$ and $\Lambda$ have been weighted with respect to the local slope (i.e., $x_2/x_3$) of the representative circular fiber at each interface (see IDP=13 Figure 9). This helps correct for the fact that the representative circle has interfaces normal to the loading direction (due to the necessarily rectangular-shaped subcells), while an actual circular fiber does not. For an actual circular fiber, the fiber matrix interface is only normal to the loading direction at two points, and as we trace the fiber's perimeter away from these points, the interface becomes increasingly parallel to the loading direction. In our representative fiber, we thus provide interfaces that are more parallel to the loading direction with larger debond strengths since the component of stress normal to the actual fiber interface (at this point) is lower. Similarly, we provide these more parallel interfaces with lower values of $\Lambda$ so that, once debonding occurs, the stress at the simulated interface (which is in the global loading direction) unloads more slowly. The weighting has been chosen as linear (with respect to the local interfacial slope) for simplicity and because this introduces only one additional parameter for each of $\sigma_{DB}$ and $\Lambda$. Note that modeling of interfacial debonding is an area of active research using **MAC/GMC**, and thus the techniques employed in this example are still developing.

The results of this example shown in the following figure have been plotted only to an applied strain level of 0.016 because at this point the slope of the predicted global stress-strain curve becomes negative. We are treating this as a criterion for simulated global failure of the composite.

The interfacial curves shown in the figure are **NOT** simple plots of the **MICRO** output files. Rather, the microlevel stress from these output files is plotted versus the macro level strain applied to the composite from the first **MACRO** output file.
test of refined subcell transverse debonding
*PRINT
   NPL=1  
*LOAD
   LCON=3 LOP=3 LSS=1  
*MECH
   NPTW=5 TI=0.,24000.,57600.,64800.,64908. LO=0.,0.,0.,0.,0.018  
*THERM
   NPTT=5 TI=0.,24000.,57600.,64800.,64908. TE=900.,534.583,23.,650.,650.  
*MODEL
   MOD=1  
*SOLVER
   NTF=1 NPTS=5 TIM=0.,24000.,57600.,64800.,64908. STP=250.,40.,40.,0.4  
*FIBER
   NFIBS=1
   NF=1 MF=6 NDPT=2 MAT=U IFM=1
   NTP=6
   TEM=21.1,204.44,315.56,426.67,537.78,871.11
   EA=57.0E3,55.98E3,55.4E3,54.82E3,54.24E3,53.36E3
   ET=57.0E3,55.98E3,55.4E3,54.82E3,54.24E3,53.36E3
   NUA=0.25,0.25,0.25,0.25,0.25,0.25
   NUT=0.25,0.25,0.25,0.25,0.25,0.25
   ALPA=3.564E-6,3.618E-6,3.726E-6,3.906E-6,4.068E-6,4.572E-6
   ALPT=3.564E-6,3.618E-6,3.726E-6,3.906E-6,4.068E-6,4.572E-6
*MATRIX
   NMATX=1
   NM=1 MM=4 NDPT=2 MAT=A  
*MRVE
   IDP=13 VF=0.35 R=1.111111111
*DEBOND
   NIl=48
   DBCH=2 NBl=13 NGl= 1 FACE= 2 BDN=7. GCN=0.060 BCN=16.5 TOLN=1  
   BDS= 40 GCS=0.1 BCS=100 Ti=64800
   DBCH=2 NBl=1 2 NGl= 2 FACE= 2 BDN=11.3 GCN=0.0486 BCN=16.5 TOLN=1  
   BDS= 40 GCS=0.1 BCS=100 Ti=64800
   DBCH=2 NBl=11 NGl= 3 FACE= 2 BDN=12. GCN=0.0467 BCN=16.5 TOLN=1  
   BDS= 40 GCS=0.1 BCS=100 Ti=64800
   DBCH=2 NBl=10 NGl= 4 FACE= 2 BDN=17. GCN=0.0334 BCN=16.5 TOLN=1  
   BDS= 40 GCS=0.1 BCS=100 Ti=64800
   DBCH=2 NBl= 9 NGl= 5 FACE= 2 BDN=18.3 GCN=0.0300 BCN=16.5 TOLN=1  
   BDS= 40 GCS=0.1 BCS=100 Ti=64800
   DBCH=2 NBl= 8 NGl= 6 FACE= 2 BDN=19. GCN=0.0280 BCN=16.5 TOLN=1  
   BDS= 40 GCS=0.1 BCS=100 Ti=64800
   DBCH=2 NBl= 7 NGl= 7 FACE= 2 BDN=22. GCN=0.0200 BCN=16.5 TOLN=1  
   BDS= 40 GCS=0.1 BCS=100 Ti=64800
DBCH=2, NBI= 6, NGI= 8, FACE= 2, BDN=25.8, GCN=0.0100, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 5, NGI= 9, FACE= 2, BDN=27, GCN=0.00671, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 4, NGI=10, FACE= 2, BDN=29.5, GCN=0.000045, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 3, NGI=11, FACE= 2, BDN=52, GCN=0, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 2, NGI=12, FACE= 2, BDN=59.5, GCN=0, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 2, NGI=14, FACE= 2, BDN=59.5, GCN=0, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 3, NGI=15, FACE= 2, BDN=52, GCN=0, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 4, NGI=16, FACE= 2, BDN=29.5, GCN=0.000045, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 5, NGI=17, FACE= 2, BDN=27, GCN=0.00671, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 6, NGI=18, FACE= 2, BDN=25.8, GCN=0.0100, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 7, NGI=19, FACE= 2, BDN=22, GCN=0.0200, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 8, NGI=20, FACE= 2, BDN=19, GCN=0.0280, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI= 9, NGI=21, FACE= 2, BDN=18.3, GCN=0.0300, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=10, NGI=22, FACE= 2, BDN=17, GCN=0.0334, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=11, NGI=23, FACE= 2, BDN=12, GCN=0.0467, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=12, NGI=24, FACE= 2, BDN=11.3, GCN=0.0486, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=13, NGI=25, FACE= 2, BDN=7, GCN=0.060, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=14, NGI=25, FACE= 2, BDN=7, GCN=0.060, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=15, NGI=24, FACE= 2, BDN=11.3, GCN=0.0486, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=16, NGI=23, FACE= 2, BDN=12, GCN=0.0467, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=17, NGI=22, FACE= 2, BDN=17, GCN=0.0334, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=18, NGI=21, FACE= 2, BDN=18.3, GCN=0.0300, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800

DBCH=2, NBI=19, NGI=20, FACE= 2, BDN=19, GCN=0.0280, BCN=16.5, TOLN=1 & BDS= 40, GCS=0.1, BCS=100, TI=64800
DBCH=2 NBI=20 NGI=19 FACE=2 BDN=22. GCN=0.0200 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=21 NGI=18 FACE=2 BDN=25.8 GCN=0.0100 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=22 NGI=17 FACE=2 BDN=27. GCN=0.00671 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=23 NGI=16 FACE=2 BDN=29.5 GCN=0.000045 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=24 NGI=15 FACE=2 BDN=52. GCN=0.0 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=25 NGI=14 FACE=2 BDN=59.5 GCN=0.0 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=25 NGI=12 FACE=2 BDN=59.5 GCN=0.0 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=24 NGI=11 FACE=2 BDN=52. GCN=0.0 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=23 NGI=10 FACE=2 BDN=29.5 GCN=0.000045 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=22 NGI=9 FACE=2 BDN=27. GCN=0.00671 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=21 NGI=8 FACE=2 BDN=25.8 GCN=0.0100 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=20 NGI=7 FACE=2 BDN=22. GCN=0.0200 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=19 NGI=6 FACE=2 BDN=19. GCN=0.0280 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=18 NGI=5 FACE=2 BDN=18.3 GCN=0.0300 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=17 NGI=4 FACE=2 BDN=17. GCN=0.0334 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=16 NGI=3 FACE=2 BDN=12. GCN=0.0467 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=15 NGI=2 FACE=2 BDN=11.3 GCN=0.0486 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800
DBCH=2 NBI=14 NGI=1 FACE=2 BDN=7. GCN=0.060 BCN=16.5 TOLN=1 & BDS=40 GCS=0.1 BCS=100 T1=64800

*CURVE
NP=3 %

*MICRO
NT=2
NC=1 X=3 Y=9 NAM=TRAN-M
NC=2 X=24 Y=3 NAM=TRAN-T

*MACRO
NT=2
NC=1 CELL=313 X=3 Y=9 NAM=TRAN-M
The following figure was obtained from the x-y plot data file produced by the present example.
6.5 Example E: General Loading

Sample Input File for General Loading

Problem Summary

Load Type: Thermo-Mechanical

Load History Data: Combined:
- 33 - normal strain (transverse to fiber)
- 13 - shear strain
- 12 - shear strain
- 11, 22, and 23 component stress-free
- Heat-up from 23 °C to 223 °C

Load History: Monotonic

Load Control: Combined stress and strain

Load History Data:

\[ T = 1.85 \, ^\circ\text{C}/\text{sec} \]
\[ \varepsilon_{33} = 9.26 \times 10^{-5}/\text{sec}, \varepsilon_{33_{\text{max}}} = 0.01, \varepsilon_{33_{\text{min}}} = 0 \]
\[ \varepsilon_{13} = 5.56 \times 10^{-5}/\text{sec}, \varepsilon_{13_{\text{max}}} = 0.006, \varepsilon_{13_{\text{min}}} = 0 \]
\[ \varepsilon_{12} = 1.11 \times 10^{-4}/\text{sec}, \varepsilon_{12_{\text{max}}} = 0.012, \varepsilon_{12_{\text{min}}} = 0 \]
\[ \Delta t = 0.4 \, \text{sec} \]

Micromechanics Model: Double Periodicity

Fiber Packing Arrangement: Square Pack, R = 1., 45% fiber volume ratio

Repeating Unit Cell: 2x2 subcells, square pack using IDP=11

Integration Algorithm: Forward Euler

Constituent Material Model:
- Fiber: Elastic, isotropic
- Matrix: GVIPS - isotropic form

Constituents:
- Fiber: SCS-6 (temp. dep. properties input manually)
- Matrix: TIMETAL21S

120
example of general loading

*PRINT
   NPL=1 %
*LOAD
   LCON=3 LOP=99 LSS=2,2,1,2,1,1 %
*MECH
   NPTW=2 T1=0.,108. LO=0.,0.
   NPTW=2 T1=0.,108. LO=0.,0.
   NPTW=2 T1=0.,108. LO=0.,0.01
   NPTW=2 T1=0.,108. LO=0.,0.
   NPTW=2 T1=0.,108. LO=0.,0.006
   NPTW=2 T1=0.,108. LO=0.,0.012
*THERM
   NPTT=2 T1=0.,108. TE=23.,223.
*MODEL
   MOD=1 %
*SOLVER
   NTF=1 NPTS=2 TIM=0.,108. STP=0.4 %
*FIBER
   NFIBS=1
   NF=1 MF=9 NDPT=2 IFM= 1 MAT=U
   NTP=6
   TEM=21.1,204.44,315.56,426.67,537.78,871.11
   EA=57.0E3,55.98E3,55.4E3,54.82E3,54.24E3,53.36E3
   ET=57.0E3,55.98E3,55.4E3,54.82E3,54.24E3,53.36E3
   NUA=0.25,0.25,0.25,0.25,0.25,0.25
   NUT=0.25,0.25,0.25,0.25,0.25,0.25
   ALPA=3.564E-6,3.618E-6,3.726E-6,3.906E-6,4.068E-6,4.572E-6
   ALPT=3.564E-6,3.618E-6,3.726E-6,3.906E-6,4.068E-6,4.572E-6
   D=1.,0.,0.
*MATRIX
   NMATX=1
   NM=1 MM=4 NDPT=2 MAT=A %
*MRVE
   IDP=11 VF=0.45 RAD=0.07 R=1.0
*CURVE
   NP=5 %
*MACRO
   NT=4
   NC=1 X=3 Y=9 NAM=GENLD-33
   NC=2 X=5 Y=11 NAM=GENLD-13
   NC=3 X=6 Y=12 NAM=GENLD-12
   NC=4 X=39 Y=1 NAM=GENLD-T %
*END
The following figures were obtained from the x-y plot data file produced by the present example.

![Graph 1: Strain vs. Temperature](image1)

![Graph 2: Stress vs. Strain](image2)
6.6 Example F: Unidirectional Laminate Problem

Sample Input File For A Laminate Problem

The following example is used to explain the control blocks in more detail.

Problem Summary:

Load Type: Mechanical
Load History: Monotonic
Load Control: Strain
Load History Data: \[ \varepsilon = 0.01 / \text{sec}, \]
\[ \varepsilon_{\text{max}} = 0.01, \]
\[ \Delta t_{\text{constant}} = 0.00025 \text{ sec} \]
Micromechanics model: Laminate Theory
Fiber Packing Arrangement: Square Pack at 46% fiber volume ratio
Integration Algorithm: Forward Euler
Constituent Material Model: Bodner-Partom
Constituents:
Fiber: Boron
Matrix: Aluminum (6061-0a)
test of laminate strain control
*PRINT
   NPL=0 %
*LOAD
   LCON=2 LOP=1 LSS=1 %
*MECH
   NPTW=2 TI=0.,1. LO=0.,0.01 %
*MODEL
   MOD=3 MATSYS=1 NLY=1 THK=1. CON=2 SYS=1 ANG=45. %
   # MOD=3 MATSYS=1 NLY=1 THK=1. CON=2 SYS=1 ANG=0. %
   # MOD=3 MATSYS=1 NLY=1 THK=1. CON=2 SYS=1 ANG=90. %
   # MOD=3 MATSYS=1 NLY=1 THK=1. CON=2 SYS=1 ANG=10. %
*SOLVER
   NTF=1 NPTS=2 TIM=0.,1. STP=0.00025 %
*FIBER
   NFIBS=1
   NF=1 MS=1 MF=6 NDPT=1 TEMP=21. MAT=A %
*MATRIX
   NMATX=1
   NM=1 MS=1 MM=1 NDPT=1 TEMP=21. MAT=C %
*MRVE
   IDP=1
   L=1 VF=0.46 %
*CURVE
   NP=10 %
*MACRO
   NT=1
   NC=1 X=1 Y=7 NAM=apdx %
*END

124
The following figures were obtained from the x-y plot data file produced by the present example. The validity of these predictions were assessed by comparing these results to those previously obtain from:

reference 1 Fig. 8-5, pg. 235

reference 1 Fig. 8-7, pg. 237
### 6.7 Example G: A Cross-Ply Laminate Problem

#### Sample Input File For A Laminate Problem

The following example is used to explain the control blocks in more detail.

**Problem Summary:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Type</td>
<td>Mechanical</td>
</tr>
<tr>
<td>Load History</td>
<td>Monotonic</td>
</tr>
<tr>
<td>Load Control</td>
<td>Stress</td>
</tr>
<tr>
<td>Load History Data</td>
<td>$\dot{\sigma} = 0.01 \text{ ksi/sec}, \sigma_{\text{max}} = 45. \text{ ksi}, \Delta t_{\text{constant}} = 1.125 \text{ sec}$</td>
</tr>
<tr>
<td>Micromechanics model</td>
<td>$[\pm 45]$, Laminate with Double Periodicity</td>
</tr>
<tr>
<td>Fiber Packing Arrangement</td>
<td>Square Pack at 45% fiber volume ratio</td>
</tr>
<tr>
<td>Integration Algorithm</td>
<td>Forward Euler</td>
</tr>
<tr>
<td>Constituent Material Model</td>
<td>Bodner-Partom</td>
</tr>
<tr>
<td>Constituents</td>
<td>Fiber: Boron (properties input manually)</td>
</tr>
<tr>
<td></td>
<td>Matrix: Aluminum (properties input manually)</td>
</tr>
</tbody>
</table>

test of [±45]2s laminate stress control

*PRINT
  NPL=0 %

*LOAD
  LCON=2 LOP=1 LSS=2 %

*MECH
  NPTW=2 TI=0.,4500. LO=0.,45. %

*MODEL
  MOD=3 MATSYS=1 NLY=4 &
  THK=0.25,0.25,0.25,0.25 &
  CON= 2, 2, 2, 2 &
  SYS= 1, 1, 1, 1 &
  ANG=45.,-45.,-45.,45. %

*SOLVER
  NTF=1 NPTS=2 TIM=0.,4500. STP=1.125 %

*FIBER
  NFIBS=1
  NF=1 MS=1 MF=6 NDPT=1 MAT=U IFM=1 &
  EL=58.E3,58.E3,0.20,0.20,24.17E3,6.3E-6,8.28E-6 %

*MATRIX
  NMATX=1
  NM=1 MS=1 MM=1 NDPT=1 MAT=U IFM=1 &
  EL=9.53E3,9.53E3,0.33,0.33,3.58E3,21.06E-6,21.06E-6 &
  VI=1.E4,49.,63.,300.,4.,1. %

*MRVE
  IDP=1,1,1,1
  L=1 VF=0.45 %
  L=2 VF=0.45 %
  L=3 VF=0.45 %
  L=4 VF=0.45 %

*CURVE
  NP=10 %

*MACRO
  NT=1
  NC=1 X=1 Y=7 NAM=apdxg %

*END
The following figure was obtained from the x-y plot data file produced by the present example and for validation was compared to Arenburg and Reddy, Fig. 16, pg. 1382.
6.8 Example H: Triple Periodic GMC; [0/90]

Sample Input File For Triple Periodic GMC

Problem Summary:

Loading: Mechanical, Strain control
Load History Data:  \( \dot{\varepsilon} = 0.01/\text{sec}, \)
\( \varepsilon_{\text{max}} = 0.01, \)
\( \Delta t_{\text{constant}} = 0.00025 \text{ sec} \)

Micromechanics model: Triple Periodic GMC
Fiber Packing Arrangement: Rectangular Pack at 46% fiber volume ratio
(input manually, see figure below)
Integration Algorithm: Forward Euler
Constituent Material Model: Bodner-Partom
Constituents:
Fiber: Boron
Matrix: Aluminum (6061-0a)

This problem uses the 3-D GMC model to simulate a [0/90] laminate as shown below.
test of gmc3d model 0/90 laminate simulation

*PRINT
  NPL=10 %

*LOAD
  LCON=2 LOP=3 LSS=1 %

*MECH
  NPTW=2 TI=0.,1. LO=0.,0.01 %

*MODEL
  MOD=2 %

*SOLVER
  NTF=1 NPTS=2 TIM=0.,1. STP=0.00025 %

*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=1 TEMP=21. MAT=A %

*MATRIX
  NMATX=1
  NM=1 MM=1 NDPT=1 TEMP=21. MAT=C %

*MRVE
  IDP=99
  NA=4 NB=2 NG=2
  D=0.67823,0.32177,0.67823,0.32177
  H=0.67823,0.32177
  L=0.32177,0.67823
  CM=M1,M1
  CM=M1,M1
  CM=M1,M1
  CM=F1,M1
  CM=M1,M1
  CM=F1,F1
  CM=M1,M1
  CM=F1,M1 %

*CURVE
  NP=10 %

*MACRO
  NT=1
  NC=1 X=3 Y=9 NAM=apdxh %

*END
The following figure was obtained from the x-y plot data file produced by the present example.

**Note:** It is recommended that a new user construct a MAC/GMC input file using the data given in this Example and then check to see if the same result plot is obtained.
6.9 Example I: A User Defined RVE

Sample Input File For A User Defined RVE

The following example is used to explain how to enter a user defined architecture.

**Problem Summary:**

<table>
<thead>
<tr>
<th>Load Type:</th>
<th>Mechanical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load History:</td>
<td>Monotonic</td>
</tr>
<tr>
<td>Load Control:</td>
<td>Strain</td>
</tr>
<tr>
<td>Load History Data:</td>
<td>( \dot{\varepsilon} = 0.1 ) /sec,</td>
</tr>
<tr>
<td></td>
<td>( \varepsilon_{max} = 0.01 ) ,</td>
</tr>
<tr>
<td></td>
<td>( \Delta t_{constant} = 0.1 ) sec</td>
</tr>
<tr>
<td>Micromechanics model:</td>
<td>Double periodic</td>
</tr>
<tr>
<td>Fiber Packing Arrangement:</td>
<td>Random composite (see figure 13)</td>
</tr>
<tr>
<td>Integration Algorithm:</td>
<td>Forward Euler</td>
</tr>
<tr>
<td>Constituent Material Model:</td>
<td>Bodner-Partom</td>
</tr>
<tr>
<td>Constituents:</td>
<td>Fiber: Boron</td>
</tr>
<tr>
<td></td>
<td>Matrix: Aluminum (6061-0a)</td>
</tr>
</tbody>
</table>
random composite rve using 2-d gmc

*PRINT
  NPL=-1 %

*LOAD
  LCON=2 LOP=2 LSS=1 %

*MECH
  NPTW=2 TI=0.,0.1 LO=0.,0.01 %

*MODEL
  MOD=1 %

*SOLVER
  NTF=1 NPTS=2 TIM=0.,0.1 STP=0.00005 %

*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=1 TEMP=21. MAT=A %

*MATRIX
  NMATX=1
  NM=1 MM=1 NDPT=1 TEMP=21. MAT=C %

*MRVE
  IDP=99
  NB=14 NG=14
  H=1,,1,,1,,1,,1,,1,,1,,1,,1,,1,,1,,1,,1,
  L=1,,1,,1,,1,,1,,1,,1,,1,,1,,1,,1,,1,,1,,1.
  CM=F1,M1,M1,M1,M1,F1,F1,M1,M1,F1,F1,M1,F1
  CM=F1,M1,F1,F1,M1,M1,F1,F1,M1,M1,F1,M1,F1
  CM=M1,M1,F1,F1,M1,M1,M1,F1,F1,M1,M1,F1,M1
  CM=M1,M1,F1,F1,M1,M1,M1,M1,F1,F1,M1,M1
  CM=M1,M1,F1,M1,F1,M1,M1,F1,F1,M1,M1,F1,M1
  CM=M1,M1,F1,M1,F1,M1,M1,F1,F1,M1,M1,F1,M1
  CM=M1,M1,F1,F1,M1,M1,F1,F1,M1,M1,F1,M1
  CM=M1,F1,F1,M1,F1,F1,M1,F1,F1,M1,F1,M1
  CM=M1,F1,F1,M1,M1,F1,F1,M1,F1,M1,F1,F1,M1
  CM=M1,F1,F1,M1,M1,F1,F1,M1,F1,M1,F1,F1,M1
  CM=M1,F1,F1,M1,F1,F1,M1,M1,F1,F1,M1,F1,F1,M1
  CM=M1,F1,F1,M1,F1,F1,M1,F1,F1,M1,F1,F1,M1
  CM=M1,F1,F1,M1,F1,F1,M1,F1,F1,M1,F1,F1,M1
  CM=M1,F1,F1,F1,M1,F1,F1,M1,F1,F1,M1,F1,F1,M1
  CM=M1,F1,F1,F1,F1,M1,F1,F1,M1,F1,F1,M1,F1,F1,M1
  CM=M1,F1,F1,F1,F1,M1,F1,F1,M1,F1,F1,M1,F1,F1,M1

*CURVE
  NP=1 %

*MACRO
  NT=1
  NC=1 X=2 Y=8 NAM=apdxi %

*END
The corresponding output produced:

MATERIAL NO. = 1 VOLUME RATIO = 0.347E+00
MATERIAL NO. = 2 VOLUME RATIO = 0.653E+00
CG - Effective/Macro Stiffness Matrix
0.221E+12 0.610E+11 0.608E+11
0.610E+11 0.148E+12 0.675E+11
0.608E+11 0.675E+11 0.147E+12
.            0.384E+11
.            0.392E+11
.            0.398E+11
CI - Inverse of Effective/Macro Stiffness
0.537E-11 -0.152E-11 -0.153E-11
-0.152E-11 0.897E-11 -0.350E-11
-0.153E-11 -0.350E-11 0.906E-11
.            0.260E-10
.            0.255E-10
.            0.251E-10
Effective Engineering Moduli
E11S = 0.186E+12
N12S = 0.282E+00
E22S = 0.111E+12
N23S = 0.390E+00
E33S = 0.110E+12
G23S = 0.384E+11
G13S = 0.392E+11
G12S = 0.398E+11
Effective Thermal Expansion Coefficients
0.120E-04 0.192E-04 0.196E-04
STopping: NPLVL == -1
6.10 Example J: A Biaxial Load

Sample Input File For A Biaxial Load

The following example is used to explain how to impose a biaxial load state.

Problem Summary:

Load Type: Mechanical
Load History: Monotonic
Load Control: Strain
Load History Data:
\[ \varepsilon_{11} = 0.01 / \text{sec}, \varepsilon_{22} = 0.0067 / \text{sec} \]
\[ \varepsilon_{11_{\text{max}}} = 0.015 \quad \varepsilon_{22_{\text{max}}} = 0.01, \]
\[ \Delta t_{\text{constant}} = 0.015 \text{ sec} \]
Micromechanics model: Double Periodic
Fiber Packing Arrangement: Square Pack at 35% fiber volume ratio
Integration Algorithm: Forward Euler
Constituent Material Model: GVIPS
Constituents:
Fiber: SCS-6 (GVIPS forced to be elastic)
Matrix: TIMETAL 21S
test of biaxial load
*PRINT
  NPL=7 %
*LOAD
  LCON=2 LOP=7 LSS=1 %
*MECH
  NPTW=2 TI=0.,1.5 LO=0.,0.015
  NPTW=2 TI=0.,1.5 LO=0.,0.010 %
*MODEL
  MOD=1 %
*SOLVER
  NTF=1 NPTS=2 TIM=0.,1.5 STP=0.015 %
*FIBER
  NFIBS=1
  NF=1 MF=4 NDPT=1 MAT=U IFM=1 &
  EL=58.E3,0.32,3.5E-06 &
  VI=0.8E-9,1.E20,0.1E-5,0.,0.85E-4,0.05,1.,1.,1.,3.3,1.8,1.35,1.,0.01 %
*MATRIX
  NMATX=1
  NM=1 MM=4 NDPT=1 TEMP=650. MAT=A %
*MRVE
  IDP=1 VF=0.35 %
*CURVE
  NP=1 %
*MACRO
  NT=2
  NC=1 X=1 Y=7 NAM=biaxl
  NC=2 X=2 Y=8 NAM=biaxt %
*END
The following figure was obtained from the x-y plot data file produced by the present example.
6.11 Example K: User Defined Inelastic Material Model

**Sample Input File For A User Defined Material Model**

**Problem Summary:**

- **Load Type:** Thermomechanical
- **Load History:** Cyclic
- **Load Control:** Strain
- **Load History Data:** Cool-down from 400°C to 23°C, hold temperature during mechanical loading
  - \(\dot{T} = 0.419 \, ^\circ\text{C}/\text{sec}\)
  - \(\dot{\varepsilon} = 1.667 \times 10^{-4} /\text{sec}, \varepsilon_{\text{max}} = 0.015, \varepsilon_{\text{min}} = 0\)
  - \(\Delta t_{\text{thermal}} = 0.5 \, \text{sec.}; \quad \Delta t_{\text{mech}} = 0.05 \, \text{sec.}\)

- **Micromechanics model:** Double Periodicity
- **Fiber Packing Arrangement:** User Input 4x2 RVE with 2 fibers
- **Integration Algorithm:** Forward Euler
- **Constituent Material Model:**
  - Fiber 1: User Model (elastic)
  - Fiber 2: User Model (elastic)
  - Matrix: User Model (Bodner-Partom via USRMAT)

- **Constituents:**
  - Fiber 1: Imaginary material, user function material properties. Elastic modulus is a function of longitudinal strain:
    \[ E_{\text{new}} = 700 - 80.(\varepsilon_{11})(E_{\text{previous}}) \]
    with units of MPa
  - Fiber 2: Imaginary material, temperature dependent properties input manually.
  - Matrix: Imaginary material, user function temperature dependent material properties. (see USRFUN subroutine)

**Note:** The purpose here is to demonstrate how to use the various user definable subroutines, i.e., USRMAT, USRFUN, USRFORMDE, USRCPEVAL see section 4.2.12 for more information.
test of user subroutines/input

*PRINT
  NPL=10 %

*LOAD
  LCON=3 LOP=1 LSS=1 %

*MECH
  NPTW=3 TI=0.,900.,990. LO=0.,0.,0.015 %

*THERM
  NPTT=3 TI=0.,900.,990. TE=400.,23.,23. %

*MODEL
  MOD=1 %

*SOLVER
  NTF=1 NPTS=3 TIM=0.,900.,990 STP=0.5,0.05 %

*FIBER
  NFIBS=2
  NF=1 MF=99 NDPT=2 MAT=U IFM=2
  NF=2 MF=99 NDPT=2 MAT=U IFM=1 NPE=2 NPV=0
  NTP=3
  TEM=21.,200.,400.
  E1=314.1E9,293.2E9,253.0E9
  E2=0.33,0.33,0.33
  ALPA=4.5E-6,5.3E-6,6.1E-6
  ALPT=4.5E-6,5.3E-6,6.1E-6

*MATRIX
  NMATX=1
  NM=1 MM=99 NDPT=2 MAT=U IFM=2 %

*MRVE
  IDP=99
  NB=4 NG=2
  H=1.,1.,1.,1.
  L=1.,1.
  CM=M1,M1
  CM=F2,M1
  CM=M1,M1
  CM=F1,M1

*CURVE
  NP=2 %

*MACRO
  NT=1
  NC=1 X=1 Y=7 NAM=apdx-k

*END
Figure was obtained from the x-y plot data file produced by the present example.
The USRMAT subroutine

The USRMAT subroutine is used here to implement the Bodner-Partom Visco-plastic model currently available in MAC/GMC

c*****************************************************************************
SUBROUTINE USRMAT(DSA, SA, PE, PV, D, LOCTISO, TIME, TSTEP,
& CTEMP, DTTEMPR, NIO, NE, NV, NS, MN, CDUM, DMGF,
& NEP, NV, NSASIZE)
c
purpose: user material constitutive model for determination of 
the inelastic strain and state variable rates
(used when ncmd = 99)

IMPLICIT DOUBLE PRECISION (A - H, O - Z)

CHARACTER*2 CDUM
DIMENSION SS(6), S(6), R(6)
DIMENSION DSA(NSASIZE), SA(NSASIZE)
DIMENSION PV(NVP), PE(NEP), D(3)

note: 1) in this subroutine, [SA] and [DSA] contain the 
 micro (subcell) quantities for aboudi’s micromechanics model 

2) arrangement of [dsa] & [sa] arrays: 
 variable location

| strain rate (1-6) (contains ENGINEERING shears)

| stress rate (7-12)

| inelastic strain rate (13-18) (contains ENGINEERING shears)

| 12 "slots" (19-30)

| for state variables

| thermal strain rate (31-36)

on entry:
SA - vector of total (integrated) quantities (see above)
PE(NE) - vector of elastic constants for material MN
(PV(NV) - vector of viscoplastic constants for material MN
D(3) - vector of direction cosines (for models 3, 7, & 9)
LOCTISO - flag indicating if ANY material exhibits local 
transverse isotropy (and global anisotropy)
= 0 - all materials are at most globally transversely isotropic (D not used)
MAC/GMC: Micromechanics Analysis Code

**TIME** - current time
**TSTEP** - current time step
**CTEMP** - current temperature
**DTEMPR** - time rate of change of temperature
**NIO** - unit number of .out file
**NE** - # of elastic constants --> 9 MAX
**NV** - # of viscoplastic constants --> 19 MAX
**NS** - subcell number
**MN** - material number
**CDUM** - material character/number designation
  (i.e. F1 = fiber #1)
**DMGF** - damage factor - if damage is included the user should multiply material stiffness terms by DMGF when using such terms in his inelastic model.

expected on exit:
**DSA** - vector of rate quantities (see above)

```
IF( NE .GT. NEP ) THEN
  CALL FATALERROR(NIO)
  WRITE(NIO, *) 'TO MANY ELASTIC PROPERTIES FOR MATERIAL # ', MN
  WRITE(NIO, *) '# USED = ', NE
  WRITE(NIO, *) '# ALLOCATED = ', NEP
  STOP
ENDIF

IF( NV .GT. NVP ) THEN
  CALL FATALERROR(NIO)
  WRITE(NIO, *) 'TO MANY NON-LINEAR PROPERTIES FOR MATERIAL # ', MN
  WRITE(NIO, *) '# USED = ', NV
  WRITE(NIO, *) '# ALLOCATED = ', NVP
  STOP
ENDIF
```

* BEGIN USER EDITS *

```
C WRITE(6, *) ' PV=', PV
C WRITE(6, *) ' PE=', PE
```

---

```
C MATERIALS #1 & #2 ARE ELASTIC
C (set inelastic strain rates to zero)

   IF ((MN .EQ. 1) .OR. (MN .EQ. 2)) THEN
      DSA(13) = 0
      DSA(14) = 0
      DSA(15) = 0
   ENDIF
```
DSA(16) = 0
DSA(17) = 0
DSA(18) = 0
RETURN
ENDIF

C ........................................................
c MATERIAL #3 --> USE BODNER-PARTOM
C ...........-- ...........................
C-- -- .....................
c copy appropriate viscoplastic material constants
C ...

IF( NV .LT. 6 ) THEN
   CALL FATALERROR(NIO)
   WRITE(NIO, '*') ' NOT ENOUGH PV SPACE: NV =', NV
   STOP
ENDIF
D0 = PV(1)
Z0 = PV(2)
Z1 = PV(3)
BM = PV(4)
AN = PV(5)
Q = PV(6)

C ........................................................
c copy stress from [sa] to [ss]
C ..........................................
SS(1) = SA(7)
SS(2) = SA(8)
SS(3) = SA(9)
SS(4) = SA(10)
SS(5) = SA(11)
SS(6) = SA(12)

C ........................................................
c compute the deviatoric stress [s] in the subcell
C ........................................................
TEMP = (SS(1) + SS(2) + SS(3)) / 3.0
S(1) = SS(1) - TEMP
S(2) = SS(2) - TEMP
S(3) = SS(3) - TEMP
S(4) = SS(4)
S(5) = SS(5)
S(6) = SS(6)

C ..................................................
AJ2 = 0.5 * (S(1)**2 + S(2)**2 + S(3)**2) + S(4)**2 + S(5)**2 +
&  S(6)**2
SQ3AJ = DSQRT( SS(1)**2 + SS(2)**2 + SS(3)**2 + 2 * (SS(4)**2 +
&  SS(5)**2 + SS(6)**2) )
SQ2 = 1.414215
IF (SQ3AJ .EQ. 0.0) THEN
   CALL ZEROR(R, 6)
ELSE
R(1) = SS(1) / SQ3AJ
R(2) = SS(2) / SQ3AJ
R(3) = SS(3) / SQ3AJ
R(4) = SQ2 * SS(4) / SQ3AJ
R(5) = SQ2 * SS(5) / SQ3AJ
R(6) = SQ2 * SS(6) / SQ3AJ

ENDIF

C if d0=0 then assume elastic and zero-out
C [dsa(13-30)] (inelastic strain rate and internal variable rates), then return
C

IF (DO .EQ. 0) THEN
  DO 100 JJ = 13, 30
    DSA(JJ) = 0.0
  100 CONTINUE
  RETURN
C
C inelastic
C
ELSE

ZEF = Z0 + Q * SA(20) + (1 - Q) * (R(1) * SA(21) + R(2) * SA(22) + R(3) * SA(23) + R(4) * SA(24) + R(5) * SA(25) + R(6) * SA(26))

IF (AJ2 .EQ. 0.0) THEN
  AL = 0.0
ELSE
  ARG1 = ZEF**2.0 / (3.0 * AJ2)

  IF (ARG1 .GT. 1E6) ARG1 = 1E6
  CON = .5 * (AN + 1.0) / AN
  ARG = CON * (ARG1)**AN.

  IF (ARG .GT. 50.0) ARG = 50.0
  AL = DO / (DEXP(ARG) * DSQRT(AJ2))
  ENDIF
C
C inelastic strain rates
C
DSA(13) = AL * S(1)
DSA(14) = AL * S(2)
DSA(15) = AL * S(3)
DSA(16) = 2 * AL * S(4)
DSA(17) = 2 * AL * S(5)
DSA(18) = 2 * AL * S(6)

C plastic work rate
C

C state variable rates
The USRFUN subroutine

The USRFUN subroutine is used here to allow material properties to be entered in functional form instead of having to be linearly interpolated within MAC/GMC.

```fortran
SUBROUTINE USRFUN(MN, TIME, TSTEP, CTEMP, DTEMPR, SA, DSA, DOLD, PEM, PVM, D, LOCTISO, ALPA, ALPT, NE, NV, NMTS, NEP, NVP, NSASIZE)

IMPLICIT DOUBLE PRECISION (A - H, O - Z)

DIMENSION DOLD(6, 6), D(3)
DIMENSION PEM(NEP, NMTS), PVM(NVP, NMTS)
DIMENSION ALPA(NMTS), ALPT(NMTS)
DIMENSION DSA(NSASIZE), SA(NSASIZE)
```
in this subroutine, [SA] and [DSA] contain the micro (subcell) quantities for aboudi's micromechanics model

2) arrangement of [dsa] & [sa] arrays:

<table>
<thead>
<tr>
<th>variable location</th>
</tr>
</thead>
<tbody>
<tr>
<td>strain rate (1-6) (contains ENGINEERING shears)</td>
</tr>
<tr>
<td>stress rate (7-12)</td>
</tr>
<tr>
<td>inelastic strain rate (13-18) (contains ENGINEERING shears)</td>
</tr>
<tr>
<td>12 &quot;slots&quot; (19-30)</td>
</tr>
<tr>
<td>for state variables</td>
</tr>
<tr>
<td>thermal strain rate (31-36)</td>
</tr>
</tbody>
</table>

NOTE: quantities in [SA] and [DSA] are SUBCELL quantities - the values on entry are for the first subcell containing material # MN - the values on exit of this subroutine will be applied to ALL SUBCELLS containing material # MN. It is thus recommended that, if using the field variables, you assign the appropriate material # to ONE SUBCELL ONLY. Use of [SA] and [DSA] in this context in conjunction with bending in laminate theory will result in erroneous results as field variables become dependent on through-thickness position (while the material # does not).

on entry:
- material number
- current time
- current time step
- current temperature
- time rate of change of temperature
- vector of total (integrated) quantities (see above)
- vector of rate quantities (see above)
- previous elastic material stiffness matrix
- vector of previous elastic constants for material # MN (where NE = # of elastic constants --> 9 MAX)
- vector of previous viscoplastic constants for material # MN (where NV = # of viscoplastic constants --> 19 MAX)
- vector of direction cosines (required for models 3, 7, & 9)
- flag indicating if ANY material exhibits local transverse isotropy (and global anisotropy) = 0 - all materials are at most globally transversely
isotropic (D not used)
= 1 - at least one material is locally transversely isotropic (D used)
ALPA(MN) - longitudinal cte for material MN
ALPT(MN) - transverse cte for material MN
NE - NUMBER OF ELASTIC PROPERTIES USED
NV - NUMBER OF VISCOPLASTIC PROPERTIES USED

* BEGIN USER EDITS *

C WRITE(6, *) 'TOP OF USRFUN'

C MATERIAL # 1: E = FUNCTION OF STRAIN & PREVIOUS E FOR USE WITH USER CONSTITUTIVE MODEL

IF (MN .EQ. 1) THEN

IF (SA(1) .GT. 0) THEN
    EAOLD = PEM(I, MN)
    EA = 700.E9 - EAOLD * SA(1) * 80.0
ELSE
    EA = 700.E9
ENDIF

ET = EA
FNA = 0.41
FNT = 0.41
GA = EA / (2.0 * (1.0 + FNA))
ALPA(MN) = 4.5E-6
ALPT(MN) = 4.5E-6

NE = 5
PEM(1, MN) = EA
PEM(2, MN) = ET
PEM(3, MN) = FNA
PEM(4, MN) = FNT
PEM(5, MN) = GA

C MATERIAL # 2: PROPERTIES = INTERPOLATED AT INPUT TEMPERATURES
THIS SUBROUTINE IS NOT CALLED FOR MN #2 SINCE THE MATERIAL PROPERTIES ARE NOT FUNCTIONAL FORM

C MATERIAL # 3: PROPERTIES = LINEAR FUNCTION OF TEMPERATURE
FOR USE WITH USER CONSTITUTIVE MODEL
ELSEIF (MN .EQ. 3) THEN

    IF (CTEMP .LT. 21.0) CTEMP = 21.0
    IF (CTEMP .GT. 400.) CTEMP = 400.

C -- ELASTIC
    E = 72.4E9 - 81.53E6 * (CTEMP - 21.)
    FN = 0.33 + 7.916E-5 * (CTEMP - 21.)
    ALP = 22.5E-6 + 3.958E-9 * (CTEMP - 21.)

    NE = 5
    PEM(1, MN) = E
    PEM(2, MN) = E
    PEM(3, MN) = FN
    PEM(4, MN) = FN
    PEM(5, MN) = E / (2. * (1. + FN))
    ALPA(MN) = ALP
    ALPT(MN) = ALP

C -- VISCOPLASTIC
    D0 = 1.E4
    Z0 = 340.E6
    Z1 = 435.E6
    BM = 300.0
    AN = 10.0 - 0.02493 * (CTEMP - 21.)
    Q = 1.0

    NV = 6
    PVM(1, MN) = D0
    PVM(2, MN) = Z0
    PVM(3, MN) = Z1
    PVM(4, MN) = BM
    PVM(5, MN) = AN
    PVM(6, MN) = Q

ENDIF

*****************************************************************************
*                                      END USER EDITS                      *
*****************************************************************************

RETURN
END
The USRFORMDE subroutine

The USRFORMDE subroutine is used here to form the elastic material stiffness matrix in MAC/GMC when a user defined constitutive model is used.

SUBROUTINE USRFORMDE(MN, PEM, PVM, D, LOCTISO, DNEW, NE, NV, NEP, NVP, NMTS)

IMPLICIT DOUBLE PRECISION (A - H, O - Z)
DIMENSION DNEW(6, 6)
DIMENSION PEM(NEP, NMTS), PVM(NVP, NMTS)
DIMENSION D(3)

* BEGIN USER EDITS *

C

WRITE(6, *) 'TOP OF USRFORMDE'

C NOTE: In the examples shown here, standard engineering material
C elastic constants (E, nu) are used. However, the user is
C free to employ any material elastic constants with his
C constitutive model (i.e., bulk modulus, etc.). Thus, the
C user must provide the equations to determine the stiffness
C components required by GMC from his elastic constants.

C MATERIAL # 1: E = FUNCTION OF STRAIN & PREVIOUS E
c FOR USE WITH USER CONSTITUTIVE MODEL

IF (MN .EQ. 1) THEN

EA = PEM(1, MN)
ET = PEM(2, MN)
FNA = PEM(3, MN)
FNT = PEM(4, MN)
GA = PEM(5, MN)

DO 100 I = 1, 6
  DO 100 J = 1, 6
    DNEW(I, J) = 0
  CONTINUE

GT = 0.5 * ET / (1 + FNT)
FK = 0.25 * EA / (0.5 * (1 - FNT) * (EA / ET) - FNA**2)

DNEW(1, 1) = EA + 4.0 * FK * FNA**2
DNEW(2, 1) = 2.0 * FK * FNA
DNEW(3, 1) = 2.0 * FK * FNA
DNEW(1, 2) = 2.0 * FK * FNA
DNEW(2, 2) = FK + GT
DNEW(3, 2) = FK - GT
DNEW(1, 3) = 2.0 * FK * FNA
DNEW(2, 3) = FK - GT
DNEW(3, 3) = FK + GT
DNEW(4, 4) = GT
DNEW(5, 5) = GA
DNEW(6, 6) = GA

ELSEIF (MN .EQ. 2) THEN

E = PEM(1, MN)
FN = PEM(2, MN)
GA = E / (2. * (1. + FN))

DO 320 I = 1, 6
  DO 320 J = 1, 6
    DNEW(I, J) = 0
  CONTINUE

GT = 0.5 * E / (1 + FN)
FK = 0.25 * E / (0.5 * (1 - FN) - FN**2)

DNEW(1, 1) = E + 4.0 * FK * FN**2
DNEW(2, 1) = 2.0 * FK * FN

C -------------------------------

C MATERIAL # 2: PROPERTIES = INTERPOLATED AT TEMPERATURES
C FOR USE WITH USER CONSTITUTIVE MODEL
C -------------------------------
MAC/GMC: Micromechanics Analysis Code

DNEW(3, 1) = 2.0 * FK * FN
DNEW(1, 2) = 2.0 * FK * FN
DNEW(2, 2) = FK + GT
DNEW(3, 2) = FK - GT
DNEW(1, 3) = 2.0 * FK * FN
DNEW(2, 3) = FK - GT
DNEW(3, 3) = FK + GT
DNEW(4, 4) = GT
DNEW(5, 5) = GA
DNEW(6, 6) = GA

---
c MATERIAL # 3: PROPERTIES = LINEAR FUNCTION OF TEMPERATURE
c FOR USE WITH USER CONSTITUTIVE MODEL
c---

ELSEIF (MN .EQ. 3) THEN

E = PEM(1, MN)
FN = PEM(3, MN)
GA = PEM(5, MN)

DO 300 I = 1, 6
   DO 300 J = I, 6
      DNEW(I, J) = 0
   CONTINUE

GT = 0.5 * E / (1 + FN)
FK = 0.25 * E / (0.5 * (1 - FN) - FN**2)

DNEW(1, 1) = E + 4.0 * FK * FN**2
DNEW(2, 1) = 2.0 * FK * FN
DNEW(3, 1) = 2.0 * FK * FN
DNEW(1, 2) = 2.0 * FK * FN
DNEW(2, 2) = FK + GT
DNEW(3, 2) = FK - GT
DNEW(1, 3) = 2.0 * FK * FN
DNEW(2, 3) = FK - GT
DNEW(3, 3) = FK + GT
DNEW(4, 4) = GT
DNEW(5, 5) = GA
DNEW(6, 6) = GA

ENDIF

******************************************************************************
* END USER EDITS
******************************************************************************

RETURN
END
The USRCPEVAL subroutine

The USRCPEVAL subroutine is used here to form the time derivative of the material stiffness matrix in MAC/GMC.

Purpose: user subroutine to allow formation of the time derivative of the material stiffness matrix.

This subroutine is used when:

a) material properties are user defined and functional form. That is: (mat .eq. 'U') .and. (ifm .eq. 2)

b) the constitutive model is user-defined, and the material properties are not functional form, and the material properties are temperature-dependent that is: (ncmd .eq. 99) .and. (ifm .ne. 2) .and. (ndpt .eq. 2)

Program: user subroutine to allow formation of the time derivative of the material stiffness matrix.

**Note:** This subroutine is used when:

- Material properties are user-defined and functional form.
- The constitutive model is user-defined, and the material properties are not functional form, and the material properties are temperature-dependent.

**Purpose:**

The subroutine allows the formation of the time derivative of the material stiffness matrix.

**Usage:**

- **SA:** Vector of total (integrated) quantities.
- **[dsa]** and **[sa]** arrays:
  - Strain rate (1-6) (contains ENGINEERING shears)
  - Stress rate (7-12)
  - Inelastic strain rate (13-18) (contains ENGINEERING shears)
  - 12 "slots" (19-30) for state variables
  - Thermal strain rate (31-36)

**Input Parameters:**

- **SA:** Vector of total (integrated) quantities.

**Output Parameters:**

- **DNEW:** New material stiffness matrix.
- **DOLD:** Old material stiffness matrix.
- **DDOT:** Time derivative of the material stiffness matrix.

**Other Variables:**

- **PEM**: Array of temperature-dependent material properties.
- **PVM**: Array of material properties.
- **ALPA**: Array of phase variables.
- **ALPT**: Array of temperature-dependent phase variables.
- **DSA**: Array of total quantities.
- **SA**: Array of total quantities.

**DIMENSION:**

- **DNEW(6, 6), DOLD(6, 6)**
- **DDOT(6, 6)**
- **PEM(NEP, NMTS), PVM(NVP, NMTS)**
- **ALPA(NMTS), ALPT(NMTS)**
- **DSA(NSASIZE), SA(NSASIZE)**
MAC/GMC: Micromechanics Analysis Code

c DSA - vector of rate quantities (see above)
c MN - material number
c TIME - current time
c TSTEP - current time step
c CTEMP - current temperature
c DTEMPR - time rate of change of temperature
c DNEW(6, 6) - current elastic material stiffness matrix
c DOLD(6, 6) - previous elastic material stiffness matrix
c PEM(NE, MN) - vector of current elastic constants for material MN
   (where NE = # of elastic constants --> 9 MAX)
c PVM(NV, MN) - vector of current viscoplastic constants for material MN
   (where NV = # of elastic constants --> 19 MAX)
c D(3) - vector of direction cosines
   (required for models 3, 7, & 9) local
   LOCTISO - flag indicating if ANY material exhibits transverse isotropy (and global anisotropy)
   = 0 - all materials are at most globally transversely isotropic (D not used)
   = 1 - at least one material is locally transversely isotropic (D used)
c ALPA(MN) - longitudinal cte for material MN
c ALPT(MN) - transverse cte for material MN

c expected on exit:
c DDOT(6, 6) - derivative with respect to TIME of stiffness matrix
*****************************************************************************

*****************************************************************************
* BEGIN USER EDITS
*****************************************************************************

DIMENSION DHOLD(6, 6, 3), DNEW1(6, 6), PEM1(9, 10), PVM1(19, 10)
DIMENSION ALPA1(10), ALPT1(10)

C WRITE(6, *) 'TOP OF USRCPEVAL'

-------------------------------------------------------------------
c MATERIAL # 1: E = FUNCTION OF STRAIN & PREVIOUS E
   FOR USE WITH USER CONSTITUTIVE MODEL
-------------------------------------------------------------------
IF (MN .EQ. 1) THEN

  --- method one - simply divide change in stiffness components by TSTEP
  DO 200 I = 1, 6
      DO 200 J = 1, 6
          DDOT(I, J) = (DNEW(I, J) - DOLD(I, J)) / TSTEP
  200    CONTINUE

-------------------------------------------------------------------
c MATERIAL # 2: PROPERTIES = INTERPOLATED AT TEMPERATURES
   FOR USE WITH USER CONSTITUTIVE MODEL
-------------------------------------------------------------------
ELSEIF (MN .EQ. 2) THEN

--- method 2 - if material props are only function of temperature, 
--- calculate stiffness at CTEMP - 0.5 & CTEMP + 0.5, difference 
--- equals change per temp, multiply by DTEMPR equals change per time

IF (CTEMP .LT. 21.0) CTEMP = 21.0
IF (CTEMP .GT. 400.) CTEMP = 400.

DO 300 I = 1, 6
  DO 300 J = 1, 6
    DO 300 K = 1, 2
      DHOLD(I, J, K) = 0
  CONTINUE

DO 420 K = 1, 2
  IF (K .EQ. 1) CTEMP1 = CTEMP - 0.5
  IF (K .EQ. 2) CTEMP1 = CTEMP + 0.5

--- this subroutine interpolates properties for material MN at
--- the temperature CTEMP1
CALL INTTMP(MN, CTEMP1, PEMI, ALPAI, ALPTI, PVMI)

--- calculate the stiffness components
CALL USRFORMDE(MN, PEMI, PVM, D, LOCTISO, DNEWI, 
& NE, NV, NEP, NVP, NMTS)
--- note: for internal constitutive models (ncmd .ne. 99), use
--- subroutine FORMDE to calculate stiffness components.
--- syntax: CALL FORMDE(MN, PEM1, D)

DO 480 I = 1, 6
  DO 480 J = 1, 6
    DNEWI(I, J) = (DHOLD(I, J, 0) - DHOLD(I, J, 2)) * DTEMPR
  CONTINUE

ELSEIF (MN .EQ. 3) THEN

--- method 2 - if material props are only function of temperature, 
--- calculate stiffness at CTEMP - 0.5 & CTEMP + 0.5, difference 
--- equals change per temp, multiply by DTEMPR equals change per time

IF (CTEMP .LT. 21.0) CTEMP = 21.0
IF (CTEMP .GT. 400.) CTEMP = 400.
DO 330 I = 1, 6
   DO 330 J = 1, 6
      DO 330 K = 1, 2
         DHOLD(I, J, K) = 0
   330 CONTINUE

DO 400 K = 1, 2
   IF (K .EQ. 1) CTEMP1 = CTEMP - 0.5
   IF (K .EQ. 2) CTEMP1 = CTEMP + 0.5

   c --- determine material props at the appropriate temperature
      CALL USRFUN(MN, TIME, TSTEP, CTEMP1, DTEMPR, SA, DSA,
      &           DOLD, PEM1, PVM1, D, LOCTISO, ALPA1, ALPT1,
      &           NE, NV, NMTS, NVP, NSASIZE)
   c --- calculate the stiffness components
      CALL USRFORMDE(MN, PEM1, PVM, D, LOCTISO, DNEW1,
      &               NE, NV, NMTS)
   c --- note: for internal constitutive models (ncmd .ne. 99), use
      c --- subroutine FORMDE to calculate stiffness components.
   c --- syntax: CALL FORMDE(MN, PEM1, D)
      DO 450 I = 1, 6
         DO 450 J = 1, 6
            DHOLD (I, J, K) = DNEW1(I, J)
      450 CONTINUE
      400 CONTINUE

      DO 500 I = 1, 6
         DO 500 J = 1, 6
            DDOT(I, J) = (DHOLD(I, J, 2) - DHOLD(I, J, 1)) * DTEMPR
      500 CONTINUE

ENDIF

******************************************************************************
* END USER EDITS                                                           *
******************************************************************************

RETURN
END
6.12 Example L: Fatigue Damage Analysis

Sample Input File For A Fatigue Damage Analysis

The following example is used to explain how to conduct fatigue damage analysis within MAC/GMC.

Problem Summary:

Load Type: Mechanical
Load History: Cyclic
Load Control: Stress
Load Component: 11 - direction (in the fiber direction)
Load History Data:
\[ \Delta D = 0.15 \]
\[ \Delta t_{\text{constant}} = 0.01797 \text{ sec} \]

\[ \sigma_{11,\text{max}} = 115. \text{ ksi, } \sigma_{11} = 128. \text{ ksi/sec} \]

Micromechanics model: Double Periodic
Fiber Packing Arrangement: Square Pack at 35% fiber volume ratio
Integration Algorithm: Forward Euler
Constituent Material Model: Elastic (fiber) and Bodner-Partom (matrix)
Constituents:
Fiber: SCS-6 (input manually)
Matrix: Ti - 15 - 3 (at 800°F, input manually)

Note: A similar example problem is described in more detail in:

The purpose here is to demonstrate how to use the *DAMAGE option.
Longitudinal fatigue analysis of square packed SCS6/Ti-15-3 system

*PRINT
  NPL=0%
*LOAD
  LCON=2 LOP=1 LSS=2%
*MECH
  NPTW=3 TI=0.0,0.89844,1.79688 LO=0.0,115.0%
*MODEL
  MOD=1%
*DAMAGE
  NCY=12 D=0.15 DMAX=1.0 FG=0 FL=1%
*SOLVER
  NTF=1 NPTS=3 TIM=0.0,0.89844,1.79688 STP=0.01797,0.01797%
*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=1 MAT=U IFM=1 &
  EL=58.03,58.03,0.25,0.25,23.2E3,3.5E-6,3.5E-6
  ANG=1.0 BN=2.0 BP=4.0 OMU=1.0 OMFL=1.0 ETU=1.0 &
  ETFL=1.0 ETM=1.0 BE=1.0 A=1.0 SFL=1.0 XML=1.0 &
  SU=1.0 SK=1

# Note sk=1 means skip fatigue damage for fiber, so damage properties are
# meaningless - but failure criteria is still applicable.
  T=1 IC=1 V=320.0%
*MATRIX
  NMATX=1
  NM=1 MM=1 NDPT=1 MAT=U IFM=1 &
  EL=12770.0,12770.0,0.32,0.32,4837.12,21.06E-6,21.06E-6 &
  VI=1000.0,120.0,120.0,10.0,3.0,1.0
  ANG=0.0 BN=0.0 BP=0.0 OMU=1.0 OMFL=1.0 ETU=1.0 &
  ETFL=1.0 ETM=1.0 BE=2.77 A=0.2302 SFL=20.3 XML=900.0 &
  SU=128.0 SK=0
  T=2 IC=1 V=0.01%
*MRVE
  IDP=1 VF=0.35%
*CURVE
  NP=1%
*MACRO
  NT=1
  NC=1 X=1 Y=7 NAM=apdxl%
*END
The following results is contained in the output file: dam2.data

Calculated Cycles To Failure At Increment 100 Applied Load Cycle

Subcell #   Nf
1  0.100000E+10
2  0.613603E+05
3  0.613603E+05
4  0.664385E+05

******************************************************************************
******************************************************************************
***** CONTROLLING SUBCELL 2 *****
NF =>  60494.2250830768
NF =>  60494.0000000000
******************************************************************************
******************************************************************************
**** CURRENT TOTAL # CYCLES =  60494.0000000000
*** AFTER APPLIED LOAD CYCLE  1
******************************************************************************
******************************************************************************
Calculated Cycles To Failure At Increment 200 Applied Load Cycle

Subcell #   Nf
1  0.100000E+10
2  0.116736E+04
3  0.116736E+04
4  0.545641E+04

******************************************************************************
******************************************************************************
***** CONTROLLING SUBCELL 2 *****
NF =>  674.890587838351
NF =>  674.000000000000
******************************************************************************
******************************************************************************
**** CURRENT TOTAL # CYCLES =  61168.0000000000
*** AFTER APPLIED LOAD CYCLE  2
******************************************************************************
******************************************************************************
Calculated Cycles To Failure At Increment 300 Applied Load Cycle

Subcell #   Nf
1  0.100000E+10
2  0.709630E+03
3  0.709630E+03
4  0.438086E+04

******************************************************************************
******************************************************************************
***** CONTROLLING SUBCELL 2 *****
NF =>  419.320819753184
NF =>  419.000000000000
******************************************************************************
******************************************************************************
**** CURRENT TOTAL # CYCLES =  61587.0000000000

158
*** AFTER APPLIED LOAD CYCLE 3
********************************************************************************
Calculated Cycles To Failure At Increment 400 Applied Load Cycle 4

Subcell # Nf
1 0.100000E+10
2 0.463247E+03
3 0.463247E+03
4 0.361975E+04
********************************************************************************

***** CONTROLLING SUBCELL 3 *****
NF => 307.516313605484
NF => 307.000000000000
********************************************************************************

***** CURRENT TOTAL # CYCLES = 61894.0000000000

*** AFTER APPLIED LOAD CYCLE 5
********************************************************************************
Calculated Cycles To Failure At Increment 500 Applied Load Cycle 5

Subcell # Nf
1 0.100000E+10
2 0.295356E+03
3 0.295356E+03
4 0.301874E+04
********************************************************************************

***** CONTROLLING SUBCELL 2 *****
NF => 232.831665300220
NF => 232.000000000000
********************************************************************************

***** CURRENT TOTAL # CYCLES = 62126.0000000000

*** AFTER APPLIED LOAD CYCLE 6
********************************************************************************
Calculated Cycles To Failure At Increment 600 Applied Load Cycle 6

Subcell # Nf
1 0.100000E+10
2 0.166907E+03
3 0.166907E+03
4 0.253239E+04
********************************************************************************

***** CONTROLLING SUBCELL 2 *****
NF => 158.292840626614
NF => 158.000000000000
MAC/GMC: Micromechanics Analysis Code

Current Total # Cycles = 62284.0000000000

After Applied Load Cycle 6

Calculated Cycles To Failure At Increment 700 Applied Load Cycle 7

<table>
<thead>
<tr>
<th>Subcell #</th>
<th>Nf</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.000000E+10</td>
</tr>
<tr>
<td>2</td>
<td>0.620698E+02</td>
</tr>
<tr>
<td>3</td>
<td>0.620698E+02</td>
</tr>
<tr>
<td>4</td>
<td>0.214841E+04</td>
</tr>
</tbody>
</table>

Controlling Subcell 3

NF => 62.0697834887934
NF => 62.0000000000000

Current Total # Cycles = 62346.0000000000

After Applied Load Cycle 7

Calculated Cycles To Failure At Increment 800 Applied Load Cycle 8

<table>
<thead>
<tr>
<th>Subcell #</th>
<th>Nf</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.000000E+10</td>
</tr>
<tr>
<td>2</td>
<td>0.735691E+01</td>
</tr>
<tr>
<td>3</td>
<td>0.735691E+01</td>
</tr>
<tr>
<td>4</td>
<td>0.195843E+04</td>
</tr>
</tbody>
</table>

Controlling Subcell 2

NF => 7.35691370876420
NF => 7.00000000000000

Current Total # Cycles = 62353.0000000000

After Applied Load Cycle 8

Calculated Cycles To Failure At Increment 900 Applied Load Cycle 9

<table>
<thead>
<tr>
<th>Subcell #</th>
<th>Nf</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.000000E+10</td>
</tr>
<tr>
<td>2</td>
<td>0.290197E+01</td>
</tr>
<tr>
<td>3</td>
<td>0.290197E+01</td>
</tr>
<tr>
<td>4</td>
<td>0.194075E+04</td>
</tr>
</tbody>
</table>

160
At this point the matrix subcells have all failed and the analysis stops.

Thus the life of the composite is 62357 cycles.
The following results is contained in the output file: dam3.data

**************************
SUBCELL # 1
IDAM F
CYCLES TO FAILURE XNF 1000000000.00000
**************************

**************************
SUBCELL # 2
IDAM F
CYCLES TO FAILURE XNF 61360.2956047440
**************************

**************************
SUBCELL # 3
IDAM F
CYCLES TO FAILURE XNF 61360.2956047440
**************************

**************************
SUBCELL # 4
IDAM F
CYCLES TO FAILURE XNF 66438.5151084627
**************************

** Current Damage At Increment 100 Applied Load Cycle 1
SUBCELL # D 1-D
  1  0.000000E+00  0.100000E+01
  2  0.149958E+00  0.850042E+00
  3  0.149958E+00  0.850042E+00
  4  0.632021E-03  0.999368E+00

**************************
SUBCELL # 1
IDAM F
CYCLES TO FAILURE XNF 1000000000.00000
**************************

**************************
SUBCELL # 2
IDAM T
CYCLES TO FAILURE XNF 1167.36343322662
**************************

**************************
SUBCELL # 3
IDAM T
CYCLES TO FAILURE XNF 1167.36343322663
**************************

**************************
SUBCELL # 4
IDAM T
CYCLES TO FAILURE XNF 5456.41016344678
**************************

** Current Damage At Increment 200 Applied Load Cycle 2
SUBCELL # D 1-D
  1  0.000000E+00  0.100000E+01
  2  0.299638E+00  0.700362E+00
MAC/GMC: Micromechanics Analysis Code

3  0.299638E+00  0.700362E+00
4  0.141115E-02  0.998589E+00

******************************************************************************
SUBCELL #  1
IDAM T
CYCLES TO FAILURE XNF   1000000000.00000
******************************************************************************

******************************************************************************
SUBCELL #  2
IDAM T
CYCLES TO FAILURE XNF   709.630361973297
******************************************************************************

******************************************************************************
SUBCELL #  3
IDAM T
CYCLES TO FAILURE XNF   709.630361973300
******************************************************************************

******************************************************************************
SUBCELL #  4
IDAM T
CYCLES TO FAILURE XNF   4380.85967672225
******************************************************************************

******************************************************************************
** Current Damage At Increment 300 Applied Load Cycle  3
SUBCELL #  D  1-D
  1  0.000000E+00  0.100000E+01
  2  0.449466E+00  0.550534E+00
  3  0.449466E+00  0.550534E+00
  4  0.243012E-02  0.997570E+00
******************************************************************************

******************************************************************************
SUBCELL #  1
IDAM T
CYCLES TO FAILURE XNF   1000000000.00000
******************************************************************************

******************************************************************************
SUBCELL #  2
IDAM T
CYCLES TO FAILURE XNF   463.247417312908
******************************************************************************

******************************************************************************
SUBCELL #  3
IDAM T
CYCLES TO FAILURE XNF   463.247417312908
******************************************************************************

******************************************************************************
SUBCELL #  4
IDAM T
CYCLES TO FAILURE XNF   3619.74747832020
******************************************************************************

******************************************************************************
** Current Damage At Increment 400 Applied Load Cycle  4
SUBCELL #  D  1-D
  1  0.000000E+00  0.100000E+01
  2  0.599071E+00  0.400929E+00
MAC/GMC: Micromechanics Analysis Code

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>1</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>1000000000.00000</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>2</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>295.355917088531</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>3</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>295.355917088531</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>4</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>3018.74014755274</th>
</tr>
</thead>
</table>

** Current Damage At Increment 500 Applied Load Cycle 5

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>D</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>1000000000.00000</th>
</tr>
</thead>
</table>

** Current Damage At Increment 600 Applied Load Cycle 6

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>D</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>166.907472034450</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>4</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>166.907472034450</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>3</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>2532.39380892976</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SUBCELL #</th>
<th>4</th>
<th>IDAM</th>
<th>CYCLES TO FAILURE XNF</th>
<th>2532.39380892976</th>
</tr>
</thead>
</table>

** Current Damage At Increment 600 Applied Load Cycle 6
MAC/GMC: Micromechanics Analysis Code

***********************************************************************
SUBCELL #  1
IDAM T
CYCLES TO FAILURE XNF  1000000000.0000
***********************************************************************

SUBCELL #  2
IDAM T
CYCLES TO FAILURE XNF  62.0697834887973
***********************************************************************

SUBCELL #  3
IDAM T
CYCLES TO FAILURE XNF  62.0697834887934
***********************************************************************

SUBCELL #  4
IDAM T
CYCLES TO FAILURE XNF  2148.40603399676
***********************************************************************

** Current Damage At Increment 700 Applied Load Cycle  7
SUBCELL #  D  1-D
  1  0.000000E+00  0.100000E+01
  2  0.987089E+00  0.129106E-01
  3  0.987089E+00  0.129106E-01
  4  0.796197E-02  0.992038E+00
***********************************************************************

SUBCELL #  1
IDAM T
CYCLES TO FAILURE XNF  1000000000.0000
***********************************************************************

SUBCELL #  2
IDAM T
CYCLES TO FAILURE XNF  7.35691370876420
***********************************************************************

SUBCELL #  3
IDAM T
CYCLES TO FAILURE XNF  7.35691370914608
***********************************************************************

SUBCELL #  4
IDAM T
CYCLES TO FAILURE XNF  1958.42907053144
***********************************************************************

** Current Damage At Increment 800 Applied Load Cycle  8
SUBCELL #  D  1-D
  1  0.000000E+00  0.100000E+01
  2  0.994882E+00  0.511765E-02
  3  0.994882E+00  0.511765E-02
  4  0.807672E-02  0.991923E+00
**Micromechanics Analysis Code**

SUBCELL # 1
IDAM T
CYCLES TO FAILURE XNF 1000000000.00000

SUBCELL # 2
IDAM T
CYCLES TO FAILURE XNF 2.90196867892969

SUBCELL # 3
IDAM T
CYCLES TO FAILURE XNF 2.90196868203595

SUBCELL # 4
IDAM T
CYCLES TO FAILURE XNF 1940.74760195980

**Current Damage At Increment 900 Applied Load Cycle 9**

SUBCELL # D 1-D
1 0.000000E+00 0.100000E+01
2 0.996420E+00 0.357991E-02
3 0.996420E+00 0.357991E-02
4 0.811000E-02 0.991890E+00

SUBCELL # 1
IDAM T
CYCLES TO FAILURE XNF 1000000000.00000

SUBCELL # 2
IDAM T
CYCLES TO FAILURE XNF 2.0280198635817

SUBCELL # 3
IDAM T
CYCLES TO FAILURE XNF 2.0280198734260

SUBCELL # 4
IDAM T
CYCLES TO FAILURE XNF 1936.69355062971

**Current Damage At Increment 1000 Applied Load Cycle 10**

SUBCELL # D 1-D
1 0.000000E+00 0.100000E+01
2 0.999034E+00 0.966462E-03
3 0.999034E+00 0.966462E-03
4 0.814345E-02 0.991857E+00
MAC/GMC: Micromechanics Analysis Code

******************************************************************************
SUBCELL #       1
IDAM  T
CYCLES TO FAILURE XNF    1000000000.00000
******************************************************************************
SUBCELL #       2
IDAM  T
CYCLES TO FAILURE XNF    -999.000000000000
******************************************************************************
SUBCELL #       3
IDAM  T
CYCLES TO FAILURE XNF    -999.000000000000
******************************************************************************
SUBCELL #       4
IDAM  T
CYCLES TO FAILURE XNF    1931.10439061903
******************************************************************************
6.13 Example M: Longitudinal Discrete Fiber Breakage

Sample Input File for Longitudinal Discrete Fiber Breakage

Problem Summary

Load Type: Thermomechanical

Load History: Cyclic

Load Control: Strain

Load Component: 11-direction (in fiber direction)

Load History Data: Cool-down from 900°C to 23°C, heat-up to 650°C, then hold temperature during mechanical loading

\[ \dot{\varepsilon} = 0.0152 \, \text{°C/sec}, \quad 0.0871 \, \text{°C/sec} \]

\[ \varepsilon_{\text{max}} = 0.012, \quad \varepsilon_{\text{min}} = 0. \]

\[ \Delta t_{\text{thermal}} = 500, \quad 40 \, \text{sec.}; \quad \Delta t_{\text{mech}} = 0.4 \, \text{sec}. \]

Micromechanics Model: Triple Periodicity

Fiber Packing Arrangement: Square Pack, 35% fiber volume ratio

Repeating Unit Cell: 1x8x14 unit cell with 28 square fibers (input manually)

Integration Algorithm: Forward Euler

Constituent Material Model: Fiber: Elastic, transversely isotropic

Matrix: GVIPS - isotropic form

Constituents:

Fiber: SCS-6 (temp. dep. properties input manually)

Matrix: TIMETAL21S

Fiber Breakage: Second implementation of debond model

Fiber Breakage Parameters: taken from room temperature SCS-6 fiber strength histogram with strengths decreased by 5.8% to account for elevated temperature effect.

\[ \Lambda = 1.0 \times 10^{-5} \frac{in^3}{kip}, \quad B = 10 \frac{1}{s} \]
**Note:** In the case of longitudinal discrete fiber breakage, each fiber is modeled with an internal weak interface oriented normal to the fiber direction encompassing the entire fiber cross-section. This interface is then given a strength, $\sigma_{DB}$, which corresponds to the fiber ultimate strength. Thus, during simulated loading, when the longitudinal stress in the fiber reaches $\sigma_{DB}$, the fiber's internal interface debonds, and the longitudinal stress in the fiber begins to unload. This simulates a local fiber failure in a real composite.

In order to model longitudinal discrete fiber breakage as realistically as possible, a subcell containing 28 fibers has been used. Square-shaped fibers have been employed because longitudinal behavior is insensitive to fiber shape, and square fibers require the smallest number of subcells. Strength data for the SCS-6 fiber was taken from a vendor-supplied histogram. The room-temperature simulated distribution, as well as the actual strength distribution are shown in the subsequent figure. Note that the strengths employed in the example (in which the tensile simulation is performed at 650°C) were reduced by 5.8% from the room-temperature values to account for the effect of the elevated temperature. In addition, the fiber strengths have been distributed over the strength ranges indicated by the histogram, rather than bunched up. That is, rather than providing 5 fibers with a strength of 600 ksi, the strengths of those 5 fibers were distributed between 575 ksi and 625 ksi (prior to the 5.8% reduction).

The results of this example shown in the subsequent figure have been plotted only to an applied strain level of 0.0096 because at this point the slope of the predicted global stress-strain curve becomes negative. We are treating this as a criterion for simulated global failure of the composite.
MAC/GMC: Micromechanics Analysis Code

Test of longitudinal discrete fiber breakage

*PRINT
  NPL=1 %

*LOAD
  LCON=3 LOP=1 LSS=1 %

*MECH
  NPTW=5 T1=0,24000,57600,64800,64920. LO=0.0,0.0,0.0,0.0,0.012 %

*THERM
  NPTT=5 T1=0,24000,57600,64800,64920. TE=900.534.583,23.,650.,650. %

*MODEL
  MOD=2 %

*SOLVER
  NTF=1 NPTS=5 TIM=0,24000,57600,64800,64920. STP=500.,40.,40.,0.4 %

*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=2 MAT=U IFM=1
  NTP=6
  TEM=21.1,204.44,315.56,426.67,537.78,871.11
  EA=57.0E3,55.98E3,54.82E3,54.24E3,53.36E3
  ET=57.0E3,55.98E3,55.4E3,54.82E3,54.24E3,53.36E3
  NUA=0.25,0.25,0.25,0.25,0.25,0.25
  NUT=0.25,0.25,0.25,0.25,0.25,0.25
  ALPA=3.564E-6,3.618E-6,3.726E-6,3.906E-6,4.068E-6,4.572E-6
  ALPT=3.564E-6,3.618E-6,3.726E-6,3.906E-6,4.068E-6,4.572E-6

*MATRIX
  NMATX=1
  NM=1 MM=4 NDPT=2 MAT=A %

*MRVE
  IDP=99
  NA=1 NB=8 NG=14
  D=1.0

#NOTE: VF = 35%
  H=0.5916,0.4084,0.5916,0.4084,0.5916,0.4084
  L=0.5916,0.4084,0.5916,0.4084,0.5916,0.4084 &
  0.5916,0.4084,0.5916,0.4084,0.5916,0.4084
  CM=F1,M1,F1,M1,F1,M1
  CM=M1,M1,M1,M1,M1,M1
  CM=F1,M1,F1,M1,F1,M1
  CM=M1,M1,M1,M1,M1,M1
  CM=F1,M1,F1,M1,F1,M1
  CM=M1,M1,M1,M1,M1,M1
  CM=F1,M1,F1,M1,F1,M1
  CM=M1,M1,M1,M1,M1,M1
  CM=F1,M1,F1,M1,F1,M1
  CM=M1,M1,M1,M1,M1,M1
  CM=F1,M1,F1,M1,F1,M1
  CM=M1,M1,M1,M1,M1,M1

170
MAC/GMC: Micromechanics Analysis Code

CM=F1,M1,F1,M1,F1,M1
CM=M1,M1,M1,M1,M1,M1
CM=F1,M1,F1,M1,F1,M1
CM=M1,M1,M1,M1,M1,M1

*DEBOND

NII=28
DBCH=2 NAI=1 NBI=1 NGI=1 FACE=1 BDN=357 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=1 NGI=3 FACE=1 BDN=403 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=1 NGI=5 FACE=1 BDN=469 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=1 NGI=7 FACE=1 BDN=565 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=1 NGI=9 FACE=1 BDN=496 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=1 NGI=11 FACE=1 BDN=503 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=1 NGI=13 FACE=1 BDN=612 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=3 NGI=1 FACE=1 BDN=450 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=3 NGI=3 FACE=1 BDN=665 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=3 NGI=5 FACE=1 BDN=603 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=3 NGI=7 FACE=1 BDN=512 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=3 NGI=9 FACE=1 BDN=407 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=3 NGI=11 FACE=1 BDN=517 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=3 NGI=13 FACE=1 BDN=556 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=5 NGI=1 FACE=1 BDN=716 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=5 NGI=3 FACE=1 BDN=593 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=5 NGI=5 FACE=1 BDN=631 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=5 NGI=7 FACE=1 BDN=640 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=5 NGI=9 FACE=1 BDN=678 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=5 NGI=11 FACE=1 BDN=584 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=5 NGI=13 FACE=1 BDN=488 GCN=0.00001 BCN=10. TOLN=1 & BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=7 NGI=1 FACE=1 BDN=575 GCN=0.00001 BCN=10. TOLN=1 &
BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=7 NGI=3 FACE=1 BDN=697 GCN=0.00001 BCN=10. TOLN=1 &
BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=7 NGI=5 FACE=1 BDN=754 GCN=0.00001 BCN=10. TOLN=1 &
BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=7 NGI=7 FACE=1 BDN=546 GCN=0.00001 BCN=10. TOLN=1 &
BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=7 NGI=9 FACE=1 BDN=537 GCN=0.00001 BCN=10. TOLN=1 &
BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=7 NGI=11 FACE=1 BDN=537 GCN=0.00001 BCN=10. TOLN=1 &
BDS=100 GCS=0.1 BCS=100 TI=0
DBCH=2 NAI=1 NBI=7 NGI=13 FACE=1 BDN=664 GCN=0.00001 BCN=10. TOLN=1 &
BDS=100 GCS=0.1 BCS=100 TI=0
*CURVE
NP=3 %
*MACRO
NT=2
NC=1 X=1 Y=7 NAM=LONG-M
NC=2 X=39 Y=1 NAM=LONG-T
*MICRO
NT=3
NC=1 CELL=1 X=1 Y=7 NAM=LONG-1
NC=2 CELL=17 X=1 Y=7 NAM=LONG-2
NC=3 CELL=33 X=1 Y=7 NAM=LONG-3 %
*END

172
Room Temperature SCS-6 Fiber Strength Histogram

Longitudinal Prediction of SCS-6/Timetal 21S with 35% fiber volume content at 650°C
6.14 Example N: Plain Weave Composite

Sample Input File for Plain Weave Composite: Step 1 - Determine Yarn Properties

Problem Summary

Load Type: Mechanical
Load Component: 33-direction (transverse to fiber)
Load History: Monotonic
Load Control: Strain
Load History Data: \( \dot{\varepsilon} = 1.0 \times 10^{-4}/\text{sec}, \epsilon_{\text{max}} = 0.018, \epsilon_{\text{min}} = 0. \)
\( \Delta t_{\text{mech}} = 4. \text{ sec}. \)
Micromechanics Model: Double Periodicity
Fiber Packing Arrangement: Square Pack, R = 1., 65% fiber volume ratio
Repeating Unit Cell: 26x26 circular fiber cross-section approx.
Integration Algorithm: Forward Euler
Constituent Material Model: Fiber: Elastic, transversely isotropic
Matrix: Elastic, isotropic
Constituents:
Fiber: AS-4 Graphite Fiber (properties input manually)
Matrix: PMR-15 Epoxy (properties input manually)

Note: Step 1 of analyzing a woven composite with MAC/GMC involves determining the effective properties of the fiber/matrix yarns which reinforce the woven composites. For the elastic example shown, this is quite simple and involves determining only the five transversely isotropic elastic properties and the two transversely isotropic CTEs for the unidirectional fiber/matrix yarn. These effective properties will then be employed in step 2 (using ncmd = 9, transversely isotropic model) to analyze the actual woven composite unit cell. Inelastic analysis of woven composites is possible with MAC/GMC, but the inelastic characterization of the unidirectional fiber/matrix yarns (for implementation with one of the transversely isotropic inelastic constitutive models) becomes more complex. Future versions of MAC/GMC may include automated routines to aid in the analysis of thermo-inelastic woven composites.
analysis of a graphite/epoxy yarn

*PRINT
  NPL=3 %
*LOAD
  LCON=2 LOP=3 LSS=1 %
*MECH
  NPTW=2 Tl=0.,108. LO=0.,0.018 %
*MODEL
  MOD=1 %
*SOLVER
  NTF=1 NPTS=2 TIM=0.,108. STP=4. %
*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=1 MAT=U IFM=1 &
  EL=31.E6,2.E6,0.2,0.25,2.E6,-0.55E-6,5.6E-6 %
*MATRIX
  NMATX=1
  NM=1 MM=6 NDPT=1 MAT=U IFM=1 &
  EL=0.5E6,0.5E6,0.41,0.41,0.1773E6,57.E-6,57.E-6 %
*MRVE
  IDP=13 VF=0.65 R=1.0
*CURVE
  NP=1 %
*MACRO
  NT=1
  NC=1 X=3 Y=9 NAM=YARN
*END

Results for fiber/matrix yarn:

EFFECTIVE ENGINEERING MODULI
E11S= .203E+08
N12S= .272E+00
E22S= .134E+07
N23S= .432E+00
E33S= .134E+07
G23S= .359E+06
G13S= .596E+06
G12S= .596E+06

EFFECTIVE THERMAL EXPANSION COEFFICIENTS
.493E-07  .310E-04  .310E-04
Sample Input File for Plain Weave Composite: Step 2 - Composite Behavior

Problem Summary

Load Type: Mechanical
Load Component: 33-direction (in the plane of the woven reinforcement)
Load History: Monotonic
Load Control: Strain
Load History Data: \( \dot{\varepsilon} = 1.0 \times 10^{-4}/\text{sec} \), \( \varepsilon_{\text{max}} = 0.018 \), \( \varepsilon_{\text{min}} = 0 \).
\( \Delta t_{\text{mech}} = 4. \text{ sec} \).
Micromechanics Model: Triple Periodicity
Repeating Unit Cell: Input manually - approximates a plain weave reinforced composite
Integration Algorithm: Forward Euler
Constituent Material Model: Fiber/Matrix Yarns: Elastic, locally transversely isotropic
Pure Matrix: Elastic, isotropic
Constituents: Fibers 1 - 4: \( V_f = 65\% \) AS-4/PMR-15 yarns with different fiber orientations (properties input manually)
Matrix 1: PMR-15 Epoxy (properties input manually)
Matrix 2, 3: \( V_f = 65\% \) AS-4/PMR-15 yarns with different fiber orientations (properties input manually)

Note: MAC/GMC currently limits the number of fiber and matrix constituents to 4 each.
**Note:** Step 2 of analyzing a woven composite with MAC/GMC involves assembling a repeating unit cell that represents the woven composite using the effective fiber/matrix yarn properties determined in step 1. The repeating unit cell employed for a plain weave composite in this example is shown in the figure below. The heterogeneous subcells in this figure are represented by effective fiber/matrix yarn properties, while the local fiber direction of these subcells is accounted for by the $D$ vector in the locally transversely isotropic constitutive model.
plain weave reinforced composite

*PRINT
  NPL= 3%

*LOAD
  LCON=2 LOP=3 LSS=1 %

*MECH
  NPTW=2 TI=0.,108. LO=0.,0.018 %

*THERM
  NPTT=2 TI=0.,108. TE=23.,23.

*MODEL
  MOD=2 %

*SOLVER
  NTF=1 NPTS=2 TIM=0.,108. STP=4. %

*FIBER
  NFIBS=4
  NF=1 MF=9 NDPT=1 MAT=U IFM=1 &
  EL=20.3E6,1.34E6,0.272,0.432,0.596E6,0.0493E-6,31.E-6 D=0.,0.,1. %
  NF=2 MF=9 NDPT=1 MAT=U IFM=1 &
  EL=20.3E6,1.34E6,0.272,0.432,0.596E6,0.0493E-6,31.E-6 D=0.,1.,0. %
  NF=3 MF=9 NDPT=1 MAT=U IFM=1 &
  EL=20.3E6,1.34E6,0.272,0.432,0.596E6,0.0493E-6,31.E-6 D=0.25,1.,0. %
  NF=4 MF=9 NDPT=1 MAT=U IFM=1 &
  EL=20.3E6,1.34E6,0.272,0.432,0.596E6,0.0493E-6,31.E-6 D=-0.25,1.,0. %

*MATRIX
  NMATX=3
  NM=1 MM=6 NDPT=1 MAT=U IFM=1 &
  EL=0.5E6,0.5E6,0.41,0.41,0.1773E6,57.E-6,57.E-6 %
  NM=2 MM=9 NDPT=1 MAT=U IFM=1 &
  EL=20.3E6,1.34E6,0.272,0.432,0.596E6,0.0493E-6,31.E-6 D=0.25,0.,1. %
  NM=3 MM=9 NDPT=1 MAT=U IFM=1 &
  EL=20.3E6,1.34E6,0.272,0.432,0.596E6,0.0493E-6,31.E-6 D=-0.25,0.,1. %

*MRVE
  IDP=99
  NA=4 NB=4 NG=4
  D=0.25,0.25,0.25,0.25
  H=1.,1.,1.,1.
  L=1.,1.,1.,1.
  CM=F1,M1,F2,M1
  CM=F1,F3,F2,F4
  CM=F2,F3,F1,F4
  CM=F2,M1,F1,M1
  CM=M1,M1,M1,M1
  CM=M3,M1,M2,M1
  CM=M3,M1,M2,M1
  CM=M1,M1,M1,M1
  CM=F2,M1,F1,M1
CM=F2,F4,F1,F3
CM=F1,F4,F2,F3
CM=F1,M1,F2,M1
CM=M1,M1,M1,M1
CM=M2,M1,M3,M1
CM=M2,M1,M3,M1
CM=M1,M1,M1,M1
*CURVE
NP=1
*MACRO
NT=1
NC=1 X=3 Y=9 NAM=WEAVE
*END
Results for woven composite:

**EFFECTIVE ENGINEERING MODULI**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_{11S}$</td>
<td>.920E+06</td>
</tr>
<tr>
<td>$N_{12S}$</td>
<td>.288E+00</td>
</tr>
<tr>
<td>$E_{22S}$</td>
<td>.158E+07</td>
</tr>
<tr>
<td>$N_{23S}$</td>
<td>.151E+00</td>
</tr>
<tr>
<td>$E_{33S}$</td>
<td>.158E+07</td>
</tr>
<tr>
<td>$G_{23S}$</td>
<td>.357E+06</td>
</tr>
<tr>
<td>$G_{13S}$</td>
<td>.351E+06</td>
</tr>
<tr>
<td>$G_{12S}$</td>
<td>.351E+06</td>
</tr>
</tbody>
</table>

**EFFECTIVE THERMAL EXPANSION COEFFICIENTS**

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_{11}$</td>
<td>.548E-04</td>
</tr>
<tr>
<td>$\alpha_{22}$</td>
<td>.261E-04</td>
</tr>
<tr>
<td>$\alpha_{33}$</td>
<td>.261E-04</td>
</tr>
<tr>
<td>$\alpha_{12}$</td>
<td>.000E+00</td>
</tr>
<tr>
<td>$\alpha_{13}$</td>
<td>.318E-22</td>
</tr>
<tr>
<td>$\alpha_{23}$</td>
<td>-.567E-23</td>
</tr>
</tbody>
</table>

**Note:** The overall volume fraction is 32.5% while that of the yarn itself is 65%.
6.15 Example O: FEAMAC Analysis

Sample Input Files For FEAMAC Analysis

Problem Summary:

Load Type: Thermomechanical; Load Control (ABAQUS)

Load History Data: From ABAQUS:

STEP 1: Cool down
Temperature: 860.0 to 427.0°C
Strain: 0.0 - 0.0

STEP 2: Transverse loading
Temperature held fixed at 427.0°C
Strain: 0.0 to 0.015

Micromechanics model: Double Periodic

Fiber Packing Arrangement:
Square Fiber (idp=2)
Triangular Pack at 33% fiber volume ratio

Circular Fiber (idp=6)
Rectangular Pack at 33% fiber volume ratio

Integration Algorithm: Forward Euler (ABAQUS)

Constituent Material Model:
Fiber: Elastic (ncmd = 7)
Matrix: TGVIPS (ncmd = 7)

Constituents:
Fiber: SCS-6
Matrix: Ti-6-4 (isotropic)

Debonding: First implementation of debond model

Finite Element Model: Consists of 2 (8-noded brick) elements each with a different material. In this way the example demonstrates the use of multiple material groups.

FEAMAC Note: All data that is struck-out in the listing is not required by FEAMAC and is ignored if present. Warnings are generated informing the user that the data provided is being ignored. FEAMAC can be employed with any size of a finite element mesh. However, FEAMAC requires that an appropriate amount of state variable space be allocated for each user material found in the finite element model. The amount of state variable space to be allocated for a given user material is based on the size of the RVE requested in the corresponding MAC/GMC.
GMC input deck. The number of state variables must be less than 1000. The *DEPVAR option under each of the *USER MATERIAL sections should be used to define the size of this space. The number of CONSTANTS should be set to one on the *USER MATERIAL line and on the following line the value should be set to one. If the material model is linear elastic and no micro curve data files have been requested then no state variable space is required for that material.

To determine the amount of space to request use the following equation:

\[
NG = \text{Number of cells in the gamma direction}
\]

\[
NB = \text{Number of cells in the beta direction}
\]

\[
NDEPVAR = \text{Number of state variables to be provided to the *DEPVAR option.}
\]

\[
NDEPVAR = 46 + (42 \times NG \times NB) < 1000
\]

Therefore, based on the predefined RVE's in section 4.2.13:

<table>
<thead>
<tr>
<th>IDP</th>
<th>NDEPVAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>214</td>
</tr>
<tr>
<td>2</td>
<td>718</td>
</tr>
<tr>
<td>3</td>
<td>718</td>
</tr>
<tr>
<td>4</td>
<td>718</td>
</tr>
<tr>
<td>6</td>
<td>2104</td>
</tr>
<tr>
<td>9</td>
<td>2104</td>
</tr>
<tr>
<td>11</td>
<td>214</td>
</tr>
<tr>
<td>13</td>
<td>28,438</td>
</tr>
</tbody>
</table>

 Too Large: Must use alternate method described below

**EXAMPLE:**

```plaintext
... *MATERIAL, NAME=MACDECK1 *USER MATERIAL, CONSTANTS=1 1 *DEPVAR 214 ...
```

Where MACDECK1 references an IDP of size 2x2.
ALTERNATE METHOD:

If a chosen IDP requires more than 1000 state variables then the following alternate procedure must be followed:

On the *USER MATERIAL line set the number of CONSTANTS equal to 1 and on the following line set the value of this constant to be two. Also set the number of state variables on the *DEPVAR option to zero.

EXAMPLE:

...  
*MATERIAL, NAME=MACDECK2  
*USER MATERIAL, CONSTANTS=1  
  2  
*DEPVAR  
  0  
...

Where MACDECK2 references an IDP that would require more than 1000 state variables (e.g. IDP = 9).

FEAMAC Note: This alternate procedure will cause execution times to increase greatly.

FEAMAC Note: It is highly recommended that linear elastic materials with no requests for micro curve data be used in all areas of the model except in areas of suspected material nonlinearity, irrespective of memory approach.
MAC/GMC Input Listings: MACMAN1:

FEAMAC DEMO - Material group No. 1, identified in ABAQUS deck as MACMAN1

*PRINT
   NPL=1 %

*LOAD
   LCON=3 LOP=3 LSS=1 %

*MECH
   NPTW=3 TW=0.0, 58000.0, 58150.0 LO=0.0, 0.0, 0.015 %

*THERM
   NPTT=3 TT=0.0, 58000.0, 58150.0 TE=860.0, 427.0, 427.0 %

*MODEL
   MOD=1 %

*SOLVER
   NTF=1 &
   NPTS=3 TIM=0.0, 58000.0, 58150.0 STP=2000.0, 1.0 %

*FIBER
   NFIBS=1
   NF=1 MF=4 NDPT=1 MAT=U IFM=1 &
   EL=58.3E3, 0.32, 3.5E-6 &
   VI=0.8E-9, 1.E20, 0.1E-5, 0.85E-4, 0.05, 0.1, 3.3, 1.8, 1.35, 1.0, 0.01 %

*MATRIX
   NMATX=1
   NM=1 MM=7 NDPT=2 MAT=A D=1.0, 1.0, 1.0 %

*MRVE
   IDP=2 VF=0.33 %

*DEBOND
   NIl=2
   DBCH=1 NBl=1 NGI=2 FACE=2 RN=0.1 BDN=35.0 RS=0.0 BDS=0.0 T=58000.0
   DBCH=1 NBl=1 NGI=3 FACE=2 RN=0.1 BDN=35.0 RS=0.0 BDS=0.0 T=58000.0 %

*CURVE
   NP=1 %

*MACRO
   NT=1
   NC=1 X=3 Y=9 NAM=macman1_s3_e3n %

*END
MAC/GMC Input Listings: MACMAN2:

FEAMAC DEMO - Material group No. 2, identified in ABAQUS deck as MACMAN2
*PRINT
   NPL=1 %
*LOAD
   LCON=3 LOP=3 LSS=1 %
*MECH
   NPTW=3 TI=0.0, 58000.0, 58150.0 LO=0.0, 0.0, 0.015 %
*THERM
   NPTT=3 TI=0.0, 58000.0, 58150.0 TE=860.0, 427.0, 427.0 %
*MODEL
   MOD=1 %
*SOLVER
   NTF=1 &
   NPTS=3 TIM=0.0, 58000.0, 58150.0 STP=2000.0, 1.0 %
*FIBER
   NFIBS=1
   NF=1 MF=9 NDPT=2 MAT=A D=1.0, 0.0, 0.0 %
*MATRIX
   NMATX=1
   NM=1 MM=7 NDPT=2 MAT=A D=1.0, 1.0, 1.0 %
*MRVE
   IDP=6 VF=0.33 R=0.75 %
*DEBOND
   NII=10
   DBCH=1 NBI=4 NGI=1 FACE=2 RN=0.1 BDN=14.0 RS=0.0 BDS=0.0 TI=58000.0
   DBCH=1 NBI=4 NGI=6 FACE=2 RN=0.1 BDN=14.0 RS=0.0 BDS=0.0 TI=58000.0
   #
   DBCH=1 NBI=2 NGI=3 FACE=2 RN=0.1 BDN=22.0 RS=0.0 BDS=0.0 TI=58000.0
   DBCH=1 NBI=2 NGI=4 FACE=2 RN=0.1 BDN=22.0 RS=0.0 BDS=0.0 TI=58000.0
   #
   DBCH=1 NBI=3 NGI=2 FACE=2 RN=0.1 BDN=18.0 RS=0.0 BDS=0.0 TI=58000.0
   DBCH=1 NBI=3 NGI=5 FACE=2 RN=0.1 BDN=18.0 RS=0.0 BDS=0.0 TI=58000.0
   #
   DBCH=1 NBI=6 NGI=3 FACE=2 RN=0.1 BDN=22.0 RS=0.0 BDS=0.0 TI=58000.0
   DBCH=1 NBI=6 NGI=4 FACE=2 RN=0.1 BDN=22.0 RS=0.0 BDS=0.0 TI=58000.0
   #
   DBCH=1 NBI=5 NGI=2 FACE=2 RN=0.1 BDN=18.0 RS=0.0 BDS=0.0 TI=58000.0
   DBCH=1 NBI=5 NGI=5 FACE=2 RN=0.1 BDN=18.0 RS=0.0 BDS=0.0 TI=58000.0
   #
*CURVE
   NP=1 %
*MACRO
   NT=1
   NC=1 X=3 Y=9 NAM=macman2_s3_e3n %
*END
ABAQUS 5.8 Input Listing:

*HEADING
VERIFICATION OF FEAMAC

*NODE, NSET=ALL
1, 0.0, 0.0, 0.0
2, 0.0, 1.0, 0.0
3, 0.0, 1.0, 1.0
4, 0.0, 0.0, 1.0
5, 1.0, 0.0, 0.0
6, 1.0, 1.0, 0.0
7, 1.0, 1.0, 1.0
8, 1.0, 0.0, 1.0
9, 2.0, 0.0, 0.0
10, 2.0, 1.0, 0.0
11, 2.0, 1.0, 1.0
12, 2.0, 0.0, 1.0

*ORIENTATION, NAME=TRANS
0.0, 0.0, 1.0, 0.0, 1.0, 0.0

*ELEMENT, TYPE=C3D8, ELSET=EL1
1, 1, 2, 3, 4, 5, 6, 7, 8

*ELEMENT, TYPE=C3D8, ELSET=EL2
2, 5, 6, 7, 8, 9, 10, 11, 12

*SOLID SECTION, MATERIAL=MACMAN1, ELSET=EL1, ORIENTATION=TRANS
1.0

*SOLID SECTION, MATERIAL=MACMAN2, ELSET=EL2, ORIENTATION=TRANS
1.0

*MATERIAL, NAME=MACMAN1
*USER MATERIAL, CONSTANTS=1
1

*DEPVAR
718

*USER DEFINED FIELD
*EXPANSION, TYPE=ANISO, USER
*MATERIAL, NAME=MACMAN2
*USER MATERIAL, CONSTANTS=1
2

*USER DEFINED FIELD
*EXPANSION, TYPE=ANISO, USER
*USER SUBROUTINE
*INCLUDE, INPUT=/home/smdant/macdev/feabench/man_demo/umacman.f

*INITIAL CONDITIONS, TYPE=TEMPERATURE
ALL, 860.0

*INITIAL CONDITIONS, TYPE=FIELD, VARIABLE=1
ALL, 1.0

** ...........................................

*STEP, AMPLITUDE=RAMP, INC=2000000
STEP ONE - COOL DOWN
*STATIC, DIRECT
2000.0, 58000.0
*TEMPERATURE
ALL, 427.0
*BOUNDARY
1, 1, 3
2, 1
2, 3
3, 1
4, 1, 2
5, 2, 3
6, 3
8, 2
9, 2, 3
10, 3,
12, 2
*NSET, NSET=ALL
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12
*EL PRINT, FREQ=1000
S
*EL PRINT, FREQ=1000
E
*EL PRINT, FREQ=1000
THE, TEMP
*NODE PRINT, FREQ=1000, NSET=ALL
U
*NODE PRINT, FREQ=1000, NSET=ALL
CF
*NODE PRINT, FREQ=1000, NSET=ALL
RF
*END STEP
**-------------------------------------------
*STEP, AMPLITUDE=RAMP, INC=2000000
STEP TWO - LOAD UP
*STATIC, DIRECT
1.0, 150.0
*BOUNDARY
1, 1, 3
2, 1
2, 3
3, 1
4, 1, 2
5, 2, 3
6, 3
8, 2
The following figure was obtained from the macro x-y plot data files produced by the present example.
ABAQUS UMAT Subroutine Listing:

C COMMON TO SOLIDS AND SHELLS
C
C 1111111111222222222233333334444444455555555666666666777
C23456789012345678901234567890123456789012345678901234567890123456782
C
C
C
C
C
C 1111111111222222222233333334444444455555555666666666777
C23456789012345678901234567890123456789012345678901234567890123456789
C
C ORDER OF ABAQUS FUNCTION CALLS
C
C (if needed) /
C USDFLD ==> UEXPAN =< or
C /
C UGENS (SHELL) <== TBD
C
C ---------------------
SUBROUTINE USDFLD(FIELD, STATEV, PNEWDT, DIRECT, T, CELENT,
& TIME, DTIME, CMNAME, ORNAME, NFIELD, NSTATV, NOEL, NPT,
& LAYER, KSTEP, KINC, KINC, KINC, NSHR)

INCLUDE 'ABA_PARAM.INC'

PARAMETER (ZERO = 0.0D0, HALF = 0.5D0)
PARAMETER (ONE = 1.0D0, TWO = 2.0D0, THREE = 3.0D0)
PARAMETER (NFROM_UTHERM = i, NFROM_UMECH = 2)

CHARACTER*8 CMNAME, ORNAME
DIMENSION FIELD(NFIELD), STATEV(NSTATV), DIRECT(3,3)
DIMENSION T(3,3), TIME(2)

CHARACTER*80 FNAME
CHARACTER*80 EXTSION
CHARACTER*80 DFNAME
CHARACTER*80 PATH
CHARACTER*80 TSTRING

CHARACTER*80 TNAME1
CHARACTER*80 TNAME2

INTEGER AB_PRINT
INTEGER GAB_PRINT
INTEGER AL_STORE
INTEGER AB_STEP

LOGICAL SIGAL_STATV

CHARACTER*8 FLGRAY(15)
DIMENSION ARRAY(15), JARRAY(15)

COMMON / KINCINFO / AB_STEP, INC, NCYCLE
COMMON / KTIME / STEP_TIME, TOT_TIME, GDT
COMMON / KMACFLAGS / NFIRSTTIME, NFIRSTELEM, NFIRSTIP,
& AL_STORE, N_PEN(100), N_PIN(100), NPTEL,
& N_PATPEN(100), N_PATPIN(100), NPATPEL,
& AB_PRINT, GAB_PRINT, SIGAL_STATV, NCALL, NFEA_FROM
COMMON / KABNAMES / PATH, FNAME, DFNAME, TNAME1, TNAME2, EXTSION
COMMON / KPREMAC / NDU, NNOEL, NNPT, NNDI, NNSHR
COMMON / KTHERMONLY / DTERMSTRESS(6), DTERMSTRAIN(6),
& DDTHERMSTRAIN_DT(6), THERMSTRESS(6), THERMSTRAIN(6)

DATA NFIRSTELEM / 0 /
DATA NFIRSTIP / 0 /
DATA NFIRSTTIME / 1 /

NFEA_FROM = 0

C****************************************************
C INITIALIZE
C****************************************************
NNOEL = NOEL
NNPT = NPT
NNDI = NDI
NNSHR = NSHR
NCALL = 0

IF (NFIRSTTIME .EQ. 1) THEN
  CALL FEAMAC_INIT(NOEL, NPT)
ENDIF

AB_PRINT = GAB_PRINT

C*****************************************************************************
C DEFINE MAC INPUT DECK NAME
C*****************************************************************************
CALL FILLSTRING(TSTRING, ' ')
CALL FILLSTRING(FNAME, ' ')

CALL STRING_COPY(FNAME, CMNAME)
CALL STRING_APPEND(FNAME, EXTSION)

TSTRING = PATH
CALL STRING_APPEND(TSTRING, FNAME)
FNAME = TSTRING

C*****************************************************************************
C DEFINE STEP INFORMATION
C*****************************************************************************
IF ( ( NOEL .EQ. NFIRSTELEM )
& .AND. ( NPT .EQ. 1 )
& .AND. ( KINC - 1 ) .NE. INC ) THEN
  NCYCLE = -1
ENDIF

IF ( ( NOEL .EQ. NFIRSTELEM )
& .AND. ( NPT .EQ. 1 ) ) THEN
  NCYCLE = NCYCLE + 1

190
ENDIF

AB_STEP = KSTEP
GDT = DTIME
STEP_TIME = TIME(1)
TOT_TIME = TIME(2)
INC = KINC - 1

C WE CAN MAKE CALLS TO GETVRM FROM HERE TO GET ADDITIONAL INFO IF NEEDED
CALL GETVRM('THE', ARRAY, JARRAY, FLGRAY, JRDC)

DO J = 1, 6
   THERMSTRAIN(J) = ARRAY(J)
   THERMSTRESS(J) = 0.0
   DTHERMSTRESS(J) = 0.0
   DTHERMSTRAIN(J) = 0.0
   DDTHERMSTRAIN_DT(J) = 0.0
ENDDO

IF ( AB_PRINT .GE. 1 ) THEN
   WRITE(NDU, *) '            ', TIME(1), STEP_TIME
   WRITE(NDU, *) '            ', TIME(2), TOT_TIME
   WRITE(NDU, *) '            ', DTIME, GDT
   WRITE(NDU, *) '            ', KSTEP, AB_STEP
   WRITE(NDU, *) '            ', KINC, INC+1
   WRITE(NDU, *) '            ', NOEL, NNOEL
   WRITE(NDU, *) '            ', NPT, NNPT
   WRITE(NDU, *) '            ', NDI
   WRITE(NDU, *) '            ', NSHR
   WRITE(NDU, *) '            ', NCYCLE
   WRITE(NDU, *) '            ', NFIRSTELEM
   WRITE(NDU, *) '            ', NFIRSTIP
   WRITE(NDU, *) '            ', NCALL
   WRITE(NDU, *) '            ', FNAME = | , FNAME, | ' 
ENDIF

RETURN
END
SUBROUTINE FEAMAC_INIT(NOEL, NPT)

INCLUDE 'ABA_PARAM.INC'

CHARACTER*80 FNAME
CHARACTER*80 EXTSION
CHARACTER*80 DFNAME
CHARACTER*80 PATH
CHARACTER*80 TSTRING
CHARACTER*80 TNAME1
CHARACTER*80 TNAME2

INTEGER AB_PRINT
INTEGER GAB_PRINT
INTEGER AL_STORE
INTEGER AB_STEP

COMMON / KINCINFO / AB_STEP, INC, NCYCLE
COMMON / KTIE / STEP_TIME, TOT_TIME, GDT

COMMON / KMACFLAGS / NFIRSTTIME, NFIRSTELEM, NFIRSTIP,
& AL_STORE, N_PEN(100), N_PIN(100), NPEL,
& N_PATPEN(100), N_PATPIN(100), NPATPEL,
& AB_PRINT, GAB_PRINT, SIGAL_STATV, NCALL, NFEA_FROM

COMMON / KABNAMES / PATH, FNAME, DFNAME, TNAME1, TNAME2, EXTSION

COMMON / KPREMAC / NDU, NNOEL, NNPT, NNDI, NNSHR

LOGICAL SIGAL_STATV

CALL FILL_STRING(PATH, ' ')
CALL FILL_STRING(FNAME, ' ')
CALL FILL_STRING(EXTSION, ' ')
CALL FILL_STRING(DFNAME, ' ')
CALL FILL_STRING(TSTRING, ' ')
CALL FILL_STRING(TNAME1, ' ')
CALL FILL_STRING(TNAME2, ' ')

******************************************************************************

***
BEGIN USER EDITS
*** USER NEEDS TO CHANGE THE NEXT LINES AS NEEDED...
PATH = '/home/smdant/macdev/feabench/man_demo'

NOTE: ABAQUS WILL CONVERT THE MATERIAL NAME TO UPPER CASE.
MAC INPUT DECK NAME EXTENSION
EXTSION = '.MAC'

THE FOLLOWING IS ONLY USED TO RESOLVE VERSION DIFFERENCES BETWEEN
SUN AND SGI
NAME OF DIAGNOSTIC FILE
DFNAME = 'MACDIAG'

MAC PLOT FILES:
- # OF POINTS TO PLOT, I.E # OF PAIRS TO FOLLOW
  NPEL = 2
- ELEMENT # AND INTEGRATION POINT # PAIRS TO PLOT
  N_PEN - ELEMENT NUMBERS
  N_PIN - INTEGRATION POINT NUMBERS
J = 1
  N_PEN(J) = 1
  N_PIN(J) = 1
J = J + 1
  N_PEN(J) = 2
  N_PIN(J) = 1

PATRAN PLOT FILES:
- # OF POINTS TO PLOT PATRAN DATA, I.E # OF PAIRS TO FOLLOW
  NPATPEL = 0
- ELEMENT # AND INTEGRATION POINT # PAIRS TO PLOT PATRAN DATA
  N_PATPEN - ELEMENT NUMBERS
  N_PATPIN - INTEGRATION POINT NUMBERS
J = 1
  N_PATPEN(J) = 1
  N_PATPIN(J) = 1
J = J + 1
  N_PATPEN(J) = 2
  N_PATPIN(J) = 1

THE FOLLOWING IS DEPRECATED OPTION AND MAY NOT BE SUPPORTED
IN THE NEXT RELEASE OF FEAMAC. IT'S USE IS NOT RECOMMENDED.
C SET THE AL_STORE VARIABLE TO 1 IF YOU WANT TO POPULATE THE STATEV
C ARRAY PER ALLISON ENGINE COMPANY’S SPECS. (IDP 1 & 11 ONLY)

C NOTES: 1) THIS OPTION REQUIRES THE USE OF THE ALTERNATE METHOD OF
C USER MATERIAL DEFINITION AS DESCRIBED IN THE MAC MANUAL.

2) A STATE VARIABLE SPACE OF 900 IS REQUIRED.

\[
\begin{array}{c|c}
M1 & M2 \\
(3) & (4) \\
\end{array}
\]

\[
\begin{array}{c|c}
\text{BETA} & F \\
\wedge & M3 \\
(1) & (2) \\
\end{array}
\]

\[
\begin{array}{c|c}
\text{INDICES} & \text{DESCRIPTION} \\
1 \text{ - } 12 & \text{MATRIX STRESS AND STRAIN IN SUBCELL M1} \\
23 \text{ - } 34 & \text{FIBER STRESS AND STRAIN} \\
132 \text{ - } 137 & \text{MATRIX MECHANICAL STRAIN COMPONENTS IN SUBCELL M1} \\
301 \text{ - } 312 & \text{MATRIX STRESS AND STRAIN IN SUBCELL M2} \\
432 \text{ - } 437 & \text{MATRIX MECHANICAL STRAIN COMPONENTS IN SUBCELL M2} \\
601 \text{ - } 612 & \text{MATRIX STRESS AND STRAIN IN SUBCELL M3} \\
732 \text{ - } 737 & \text{MATRIX MECHANICAL STRAIN COMPONENTS IN SUBCELL M3} \\
\end{array}
\]

AL_STORE = 0

AB_PRINT = 0

C *****************************************
C *** TEMP FILE LOCATIONS
C *** SYS ADMIN OR USER NEEDS TO CHANGE THE NEXT LINES AS NEEDED...
TNAME1 = '/var/tmp/feamacl'
TNAME2 = '/var/tmp/feamac2'

C *****************************************
C *** END OF USER EDITS ***
**MAC/GMC: Micromechanics Analysis Code**

```fortran
C ***********************************************
TSTRING = PATH
CALL STRING_APPEND(TSTRING, '/ ')
PATH = TSTRING

TSTRING = PATH
CALL STRING_APPEND(TSTRING, DFNAME)
DFNAME = TSTRING

DO 100 J = 1, NPEL
   IF ((N_PIN(J) .LT. 1) .OR. (N_PIN(J) .GT. 8)) THEN
      N_PIN(J) = 1
   ENDIF
CONTINUE
100 CONTINUE

NDU = 7
C UN-COMMENT NEXT 2 LINES IF YOUR ON A SGI
C NDU = 18
C OPEN(UNIT = NDU, FILE = DFNAME, STATUS = 'UNKNOWN')
WRITE(NDU, '*') 'BEGIN'
NFIRSTELEM = NOEL
NFIRSTIP = NPT
NCYCLE = -1
GAB_PRINT = AB_PRINT
RETURN
END

SUBROUTINE UMAT(STRESS, STATEV, DDSDE, SSE, SPD, SCD, RPL,
& DDSDDE, DRPLDE, DRPLDT, STRAN, DSTRAN, TIME, DTIME, TEMP,
& DTEMP, PREDEF, DPRED, CMNAME, NDI, NSHR, NTENS, NSTATV,
& PROPS, NPROPS, COORDS, DROT, PNEWDT, CELENT, DFRGD0,
& DFRGD1, NOEL, NPT, LAYER, KSPT, KSTEP, KINC)

C STRESS(NTENS) - PASSED IN AS THE STRESSES TENSOR AT THE BEGINNING
C OF THE INCREMENT, SIGMA_I_J.  (THIS SHOULD BE UPDATED IN
C THIS ROUTINE.)
C -IN/OUT
```

195
STATEV(NSTATV) - SOLUTION DEPENDANT STATE VARIABLES.
- IN/OUT
   Format of state variable space...
   total strain (1-6)
   cauchy stress (7-12)
   inelastic strain (13-18)
   state variable(s) (19-30)
   (space for 2 (6x1) vectors)
   thermal strain (31-36)
   temperature (37)
   future use (38-40)
   future use (41-43)
   future use (44-46)

   total strain (47-52)
   cauchy stress (53-58)
   inelastic strain (59-64)
   internal state variable(s) (65-76)
   (space for 2 [6x1] vectors)
   thermal strain (77-82)
   rn & rd debond parameters

   etc. (89-...)

DDSDDE(NTENS, NTENS) - JACOBIAN MATRIX OF THE CONSTITUTIVE
MODEL.
- OUT
SSE - SPECIFIC ELASTIC STRAIN ENERGY.
- OUT
SPD - PLASTIC DISSIPATION.
- OUT
SCD - CREEP DISSIPATION.
- OUT
RPL - VOLUMETRIC HEAT GENERATION PER UNIT TIME AT THE END
OF THE INC.
- OUT (COUPLED)
DDSDDT(NTENS) - VARIATION OF STRESS INCREMENTS WITH
RESPECT TO TEMPERATURE.
- OUT (COUPLED)
DRPLDE(NTENS) - VARIATION OF RPL WRT STRAIN INCREMENTS.
KINC - INCREMENT NUMBER

INCLUDE 'ABA_PARAM.INC'
PARAMETER (ZERO = 0.0D0, HALF = 0.5D0)
PARAMETER (ONE = 1.0D0, TWO = 2.0D0, THREE = 3.0D0)
PARAMETER (NFROM_UMEC = 1, NFROM_UMECH = 2)

CHARACTER*8 CMNAME
CHARACTER*80 FNAME
CHARACTER*80 EXTENSION
CHARACTER*80 DFNAME
CHARACTER*80 PATH
CHARACTER*80 TNAME1
CHARACTER*80 TNAME2

DIMENSION STRESS(NTENS)
DIMENSION STATEV(NSTATV)
DIMENSION DDSDE(NTENS, NTENS)
DIMENSION DDSDDT(NTENS)
DIMENSION DRPLDE(NTENS)
DIMENSION STRAN(NTENS)
DIMENSION DSTRAN(NTENS)
DIMENSION TIME(2)
DIMENSION PREDEF(1)
DIMENSION DPRED(1)
DIMENSION PROPS(NPROPS)
DIMENSION COORDS(3)
DIMENSION DROT(3, 3)
DIMENSION DFGRD0(3, 3)
DIMENSION DFGRDI(3, 3)

DIMENSION PLSTRAN(6)
DIMENSION OSTRESS(6)
DIMENSION OSTRAIN(6)
DIMENSION ODSTRAIN(6)

INTEGER N(2)
INTEGER AB_PRINT
INTEGER GAB_PRINT
INTEGER AL_STORE
INTEGER AB_STEP

COMMON / KINCINFO / AB_STEP, INC, NCYCLE
COMMON / KTIME / STEP_TIME, TOT_TIME, GDT

COMMON / KMACFLAGS / NFIRSTTIME, NFIRSTELEM, NFIRSTIP,
&       AL_STORE, N_PIN(100), N_PEN(100), NPEL,
&       N_PATPIN(100), N_PATPEN(100), NPATPEL,
&       AB_PRINT, GAB_PRINT, SIGAL_STATV, NCALL, NFEA_FROM

COMMON / KABNAMES / PATH, FNAME, DFNAME, TNAME1, TNAME2, EXTENSION
COMMON / KPREDMAC / NDU, NNOEL, NNPT, NNDI, NNSHR
COMMON / KHERMONLY / DTHERMSTRESS(6), DTHERMSTRAIN(6), &
          DDTHERMSTRAIN_DT(6), THERMSTRESS(6), THERMSTRAIN(6)

LOGICAL SIGAL_STATV

C ================

NFEA_FROM = NFROM_UMECH

C*************************************************
C KEEP TRACK OF THE NUMBER OF CALLS TO FEAMAC
C*************************************************
C IF NCALL == 1->2 : WE HAVE ALREADY BEEN TO THERMAL
C IF NCALL == 0->1 : NO THERMAL WAS REQUESTED OR DTEMP IS ZERO
   NCALL = NCALL + 1

C*************************************************
C FLAG TO DETERMINE HOW DATA WILL BE STORED
C IF SIGAL_STATV == TRUE - DATA STORED IN STATE VARIABLE
C IF SIGAL_STATV == FALSE - DATA STORED IN FILES
C*************************************************
SIGAL_STATV = .TRUE.
AB_PRINT = GAB_PRINT
IF( NPROPS .GE. 1 ) THEN
   IF( PROPS(1) .EQ. 2 ) SIGAL_STATV = .FALSE.
   IF( NPROPS .GE. 2 ) AB_PRINT = INT(PROPS(2))
ELSE
   SIGAL_STATV = .FALSE.
ENDIF

N(1) = NOEL
N(2) = NOEL

IF ( AB_PRINT .GE. 1 ) THEN
   WRITE(NDU, *) '&
   WRITE(NDU, *)' '=================================================================='
   WRITE(NDU, '*') '-=> START OF UMAT <=-'
   WRITE(NDU, '*') '-=> STORE IN STATEV ? =', SIGAL_STATV
   WRITE(NDU, '*') '-=> VALUE OF STEP TIME =', TIME(1), STEP_TIME
   WRITE(NDU, '*') '-=> TOTAL TIME =', TIME(2), TOT_TIME
   WRITE(NDU, '*') '-=> DTIME =', DTIME, GDT
   WRITE(NDU, '*') '-=> KSTEP =', KSTEP, AB_STEP
   WRITE(NDU, '*') '-=> KINC =', KINC, INC+1
   WRITE(NDU, '*') '-=> NOEL =', NOEL, NNOEL
   WRITE(NDU, '*') '-=> NPT =', NPT, NNPT
   WRITE(NDU, '*') '-=> NCYCLE =', NCYCLE
   WRITE(NDU, '*') '-=> NFIRSTLEM =', NFIRSTLEM
   WRITE(NDU, '*') '-=> NFIRSTTP =', NFIRSTTP
   WRITE(NDU, '*') '-=> NCALL =', NCALL
   WRITE(NDU, '*') '-=> FNAME =', FNAME
   WRITE(NDU, '*') '-=> TEMP =', TEMP
   WRITE(NDU, '*') '-=> DTEMP =', DTEMP
   WRITE(NDU, '*') 'STRAIN IN:'
   WRITE(NDU, '*') '|', STRAN(1), STRAN(2), STRAN(3)
WRITE(NDU, *) 'CMNAME = ', CMNAME, '|
WRITE(NDU, *) 'NDI = ', NDI
WRITE(NDU, *) 'NSHR = ', NSHR
WRITE(NDU, *) 'NTENS = ', NTENS
WRITE(NDU, *) 'NSTATV = ', NSTATV
WRITE(NDU, *) 'NPROPS = ', NPROPS
WRITE(NDU, *) 'CELENT = ', CELENT
WRITE(NDU, *) 'NOEL = ', NOEL
WRITE(NDU, *) 'NOEL = ', NOEL
WRITE(NDU, *) 'NPT = ', NPT
WRITE(NDU, *) 'LAYER = ', LAYER
WRITE(NDU, *) 'KSPT = ', KSPT
WRITE(NDU, *) 'KSTEP = ', KSTEP
WRITE(NDU, *) 'KINC = ', KINC

WRITE(NDU, *) '-=> N=', N, '|
WRITE(NDU, *) '-=> INC =', INC
WRITE(NDU, *) '-=> NCYCLE =', NCYCLE
WRITE(NDU, *) '-=> TEMP =', TEMP
WRITE(NDU, *) '-=> DTEMP =', DTEMP
WRITE(NDU, *) '-=> BEFORE FEAMAC_PRE <=-
ENDIF

DO J = 1, 6
OSTRESS(J) = STRESS(J)
STRAN(J) = STRAN(J) + THERMSTRAIN(J)
DSTRAN(J) = DSTRAN(J) + DTHERMSTRAIN(J)
ODSTRAIN(J) = DSTRAN(J)
ENDDO

CALL FEAMAC_PRE(DDSDDE, PLSTRAN, STRAN, DSTRAN, STRESS, STATEV, & NSTATV, NTENS, N, NPT, LAYER, NDI, NSHR, & TEMP, DTEMP, THERMSTRAIN)

IF( (NCALL .EQ. 2) .AND. (NCYCLE .NE. 0) ) THEN
DO I = 1, 6
STRESS(I) = OSTRESS(I)
DO J = 1, 6
STRESS(I) = STRESS(I)
& + DDSDE(I, J) * (ODSTRAIN(J) - DTHERMSTRAIN(J))
ENDDO
ENDDO

DO J = 1, 6
IF ( DTEMP .NE. 0.0 ) THEN
   DDSDDT(J) = DTMERSTRESS(J) / DTEMP
ELSE
   DDSDDT(J) = 0.0
ENDIF
ENDDO
DRPLDE(J) = 0.0
ENDIF

IF ( AB_PRINT .GE. 1 ) THEN
   WRITE(NDU, *) '=> AFTER FEAMAC_PRE <=' 
   WRITE(NDU, *) 'DTHERMAL STRESS:'
   WRITE(NDU, *) 'I', DTMERSTRESS(1), DTMERSTRESS(2), &
      DTMERSTRESS(3)
   WRITE(NDU, *) 'I', DTMERSTRESS(4), DTMERSTRESS(5), &
      DTMERSTRESS(6)
   WRITE(NDU, *) 'DTHERMAL STRAIN:'
   WRITE(NDU, *) 'I', DTMERSTRAIN(1), DTMERSTRAIN(2), &
      DTMERSTRAIN(3)
   WRITE(NDU, *) 'I', DTMERSTRAIN(4), DTMERSTRAIN(5), &
      DTMERSTRAIN(6)
   WRITE(NDU, *) 'DDSDDT:'
   WRITE(NDU, *) 'I', DDSDDT(1), DDSDDT(2), &
      DDSDDT(3)
   WRITE(NDU, *) 'I', DDSDDT(4), DDSDDT(5), &
      DDSDDT(6)
   WRITE(NDU, *) '=> END OF UMAT <=' 
ENDIF

C WHATEVER IS IN STRAN/DSTRAN WILL SHOW UP AS MECHANICAL STRAIN
C MECHANICAL STRAIN + THERMAL STRAIN WILL GIVE GLOBAL STRAIN
C whatever is in stress will be given as global stress
DO I = 1, 6
   STRAN(I) = OSTRAIN(I) - THERMSTRAIN(I)
   DSTRAN(I) = ODSTRAIN(I) - DTHERMSTRAIN(I)
ENDDO

NFIRSTTIME = 0
RETURN
END

C

SUBROUTINE UEXPAN(EXPAN, DEXPANDT, TEMP, TIME, DTIME, PREDEF, &
       DPRED, STATEV, CMNAME, NSTATV)

INCLUDE 'ABA_PARAM.INC'

PARAMETER (ZERO = 0.0D0, HALF = 0.5D0)
PARAMETER (ONE = 1.0D0, TWO = 2.0D0, THREE = 3.0D0)
PARAMETER (NFROM_UTHERM = 1, NFROM_UMECH = 2)

CHARACTER*8 CMNAME

DIMENSION EXPAN(*), DEXPANDT(*), TEMP(2), TIME(2), PREDEF(*)
DIMENSION DPRED(*), STATEV(NSTATV)

CHARACTER*80 FNAME
CHARACTER*80 EXTSION
CHARACTER*80 DFNAME
CHARACTER*80 PATH
CHARACTER*80 TNAME1
CHARACTER*80 TNAME2

DIMENSION STRAN(6), DSTRAN(6)
DIMENSION STRESS(6)
DIMENSION PLSTRAN(6)
DIMENSION DDEDDT(6, 6)

INTEGER N(2)
INTEGER AB_PRINT
INTEGER GAB_PRINT
INTEGER AL_STORE
INTEGER AB_STEP

COMMON / KINCINFO / AB_STEP, INC, NCYCLE
COMMON / KTIME / STEP_TIME, TOT_TIME, GDT

COMMON / KMACFLAGS / NFIRSTTIME, NFIRSTELEM, NFIRSTIP, &
   AL_STORE, N_PIN(100), N_PEN(100), NPEL, &
   N_PATPIN(100), N_PATPEN(100), NPATPEL, &
   AB_PRINT, GAB_PRINT, SIGAL_STATV, NCALL, NFEA_FROM

COMMON / KABNAMES / PATH, FNAME, DFNAME, TNAME1, TNAME2, EXTSION
COMMON / KPREMAC / NDU, NNOEL, NNPT, NNDI, NNSHR
COMMON / KTHERMONLY / DTERMSTRESS(6), DTERMSTRAIN(6), &
   DDTHERMSTRAIN_DT(6), THERMSTRESS(6), THERMSTRAIN(6)

LOGICAL SIGAL_STATV

C ===============
NFEA_FROM = NFROM_UTHERM

AB_PRINT = GAB_PRINT

DO J = 1, 6
   PLSTRAN(J) = 0.0
   STRAN(J) = 0.0
   DSTRAN(J) = 0.0
   STRESS(J) = 0.0
   PLSTRAN(J) = 0.0

C 202
MAC/GMC: Micromechanics Analysis Code

```
DTHERMSTRESS(J) = 0.0
DTHERMSTRAIN(J) = 0.0
DDTHERMSTRAIN_DT(J) = 0.0
DO I = 1, 6
    DDEDDT(I, J) = 0.0
ENDDO
EXPAN(J) = 0.0
DEXPANDT(J) = 0.0
ENDDO

C NOTE THE "TIME" ARRAY FOR THIS ROUTINE IS PROVIDED AT THE END OF THE INCREMENT
TIME(1) = TIME(1) - DTIME
TIME(2) = TIME(2) - DTIME

C NOTE THE "TEMP" ARRAY FOR THIS ROUTINE IS PROVIDED AT THE END OF THE INCREMENT
DTEMP = TEMP(2)
TEMP(1) = TEMP(1) - DTEMP
TEMPERATURE = TEMP(1)

IF ( DTEMP .EQ. 0.0 ) THEN
    WRITE(NDU, *) 'UEXPAN EARLY EXIT - DTEMP = ', DTEMP,
    RETURN
ENDIF
NCALL = NCALL + 1
SIGAL_STATV = .FALSE.
IF ( AB_PRINT .GE. 1 ) THEN
    WRITE(NDU, *) '========================================================================
    WRITE(NDU, *) 'START OF UEXPAN <=-
    WRITE(NDU, *) 'VALUE OF STEP TIME = ', TIME(1), STEP_TIME
    WRITE(NDU, *) 'TOTAL TIME = ', TIME(2), TOT_TIME
    WRITE(NDU, *) 'DTIME = ', DTIME, GDT
    WRITE(NDU, *) 'KSTEP = ', AB_STEP
    WRITE(NDU, *) 'KINC = ', INC+1
    WRITE(NDU, *) 'NOEL = ', NNOEL
    WRITE(NDU, *) 'NPT = ', NNPT
    WRITE(NDU, *) 'NNDI = ', NNDI
    WRITE(NDU, *) 'NNSHR = ', NNSHR
    WRITE(NDU, *) 'NCYCLE = ', NCYCLE
    WRITE(NDU, *) 'NFIRSTELEM = ', NFIRSTELEM
    WRITE(NDU, *) 'NFIRSTIP = ', NFIRSTIP
    WRITE(NDU, *) 'NCALL = ', NCALL
    WRITE(NDU, *) 'FNAME = '|, FNAME, '|
    WRITE(NDU, *) 'TEMPERATURE = ', TEMPERATURE
    WRITE(NDU, *) 'DTEMP = ', DTEMP
    WRITE(NDU, *) 'BEFORE FEAMAC_PRE <=-
ENDIF

N(1) = NNOEL
N(2) = NNOEL
LAYER = 0
NTENS = 6
```
C NOTE: HERE DSTRAN IS NOT A RATE BUT THE ALPHAS INSTEAD - I.E. THE VARIATION OF THERMAL STRAINS WITH RESPECT TO TEMPERATURE (NOT TIME)

CALL FEAMAC_PRE(DDEDDT, PLSTRAN, STRAN, DSTRAN, STRESS, STATEV,
    & NSATV, NTENS, N, NNPT, LAYER, NNDI, NNSHR,
    & TEMPERATURE, DTEMP, THERMSTRAIN)

C NOTE: THESE ARE INCREMENTS IN THERMAL STRAIN/STRESS
C NEED TO FIGURE OUT A GOOD WAY TO GET THE SIZES OF THESE

DO J = i, 6
    EXPAN(J) = STRAN(J)
    DTHERMSTRAIN(J) = STRAN(J)
    DDETTHERMSTRAIN_DT(J) = DSTRAN(J)
    DEXPANDT(J) = DSTRAN(J)
ENDDO

DO I = i, 6
    DTHERMSTRESS(I) = 0.0
    DO J = i, 6
        DTHERMSTRESS(I) = DTHERMSTRESS(I) + DDEDDT(I, J) * DTHERMSTRAIN(J)
    ENDDO
    STRESS(I) = DTHERMSTRESS(I)
ENDDO

RETURN
END
6.16 Example P: Non-Symmetric Laminate

Sample Input File For Non-Symmetric Laminate Problem

The following example is used to explain the use of monolithic layer in more detail.

Problem Summary:

Load Type: Thermomechanical
Load Component: 11-direction (applied mid-plane strain)
Load History: Monotonic
Load Control: Strain
Load History Data: Cool-down from 371.1°C to 21.1 °C then hold temperature constant during mechanical loading

\[ \dot{T} = 35.\, °C/sec \]

\[ \dot{\varepsilon} = 1 \times 10^{-3} /sec, \quad \varepsilon_{max} = 0.01, \]
\[ \varepsilon_{min} = 0. \]

\[ \Delta t_{thermal} = 0.05 \, sec, \quad \Delta t_{mech} = 0.01 \, sec \]

Micromechanics model: Laminate Option

Laminate Details:

<table>
<thead>
<tr>
<th>Layer</th>
<th>Thickness (mm)</th>
<th>Fiber</th>
<th>Matrix</th>
<th>Fiber Volume Fraction</th>
<th>Unit Cell</th>
<th>Packing Arrangement</th>
<th>Angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>monolithic</td>
<td>Aluminum (2024-T4)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>Graphite T-50</td>
<td>Aluminum (2024-T4)</td>
<td>30%</td>
<td>2x2 sub-cells</td>
<td>square</td>
<td>90</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>Graphite T-50</td>
<td>Aluminum (2024-T4)</td>
<td>30%</td>
<td>2x2 sub-cells</td>
<td>square</td>
<td>0</td>
</tr>
</tbody>
</table>

Integration Algorithm: Forward Euler

Constituent Material Model: Bodner-Partom
Note: The laminate analyzed in this example is shown in following figure. The macro output for the laminate option consists of stress resultants and mid-plane strains. The micro output is always in the local coordinates of the layer in question. Micro output is given for both integration points within the layer in question.
example of a non-symmetric laminate with monolithic and heterogeneous layers
*PRINT
  NPL=0 %
*LOAD
  LCON=3 LOP=1 LSS=1
*Mech
  NPTW=3 TI=0.0, 10.0, 20.0 LO=0.0, 0.0, 0.01
*THERM
  NPTT=3 TI=0.0, 10.0, 20.0 &
  TE=371.1, 21.1, 21.1 %
*MODEL
  MOD=3 MATSYS=2 NLY=3 THK=0.001,0.001,0.001 CON=1,2,2 &
  SYS=1,2,2 ANG=0.,90.,0. %
*SOLVER
  NTF=1 NPTS=3 TIM=0.0, 10.0, 20.0 STP=0.05, 0.01 %
*FIBER
  NFIBS=1
  NF=1 MS=2 MF=9 NDPT=2 MAT=A D=1.,0.,0. %
*MATRIX
  NMATX=1
  NM=1 MS=2 MM=1 NDPT=2 MAT=A %
*MONOL
  NMON=1
  NMO=1 MS=1 MMO=1 NDPT=2 MAT=A %
*MRVE
  IDP=0,1,1
  L=1
  L=2 VF=0.3
  L=3 VF=0.3 %
*CURVE
  NP=20 %
*MACRO
  NT=1
  NC=1 X=1 Y=7 NAM=lam-x %
*MICRO
  NT=3
  NC=1 LYR=1 X=37 Y=1 NAM=lam-x1
  NC=2 LYR=2 CELL=2 X=37 Y=2 NAM=lam-x2
  NC=3 LYR=3 CELL=2 X=37 Y=1 NAM=lam-x3 %
*END
The following figures were obtained from the macro and micro x-y plot data files produced by the present example.

**Global Level Results**

**Local Level Results**

Note: The increasing magnitude of the strain towards the lower surface of the laminate is due to the increased bending resulting from the nonsymmetric lay-up.
6.17 Example Q: Yield Surface Plot

Sample Input File for a Yield Surface Plot

The following example is used to explain the yield surface capabilities in more detail.

Problem Summary:

Load Type: Themomechanical

Load Component: Combined 11-component and 22-component

Load History: Yield surface probing

Load Control: Strain

Load History Data: Constant temp. = 23 °C

\[ \varepsilon = 1.0 \times 10^{-5}/\text{sec}, \varepsilon_{\text{max}} = 0.05, \varepsilon_{\text{min}} = 0. \]

(Along each probing angle)

\[ \Delta t = 5. \text{ sec.} \]

Micromechanics model: Double Periodicity

Fiber Packing Arrangement: Square pack at 35% fiber volume ratio

Integration Algorithm: Forward Euler

Constituent Material Model: Fiber: Elastic

Matrix: GVIPS – isotropic form

Constituents: Fiber: SCS-6 (properties input manually)

Matrix: TIMETAL21S
test of yield surface option
*PRINT
  NPL=0  %
*LOAD
  LCON=3  LOP=1  LSS=1  %
*MECH
  NPTW=2  TI=0.,5000.  LO=0.,0.05  %
*THERM
  NPTT=2  TI=0.,5000.  TE=23.,23.  %
*SURF
  NPRE=0  ISP=1  IAN=10
  C1=1.0E-6  C2=1.45E-7  C3=0.0  C4=0.0
*MODEL
  MOD=1  %
*SOLVER
  NTF=1  NPTS=2  TIM=0.,5000.  STP=5.  %
*FIBER
  NFIBS=1
  NF=1  MF=6  NDPT=1  MAT=U  IFM=1&
  EL=58.E3,58.E3,0.20,0.20,24.167E3,4.5E-6,4.5E-6  %
*MATRIX
  NMATX=1
  NM=1  MM=4  NDPT=2  MAT=A  %
*MRVE
  IDP=1  VF=0.35  %
*CURVE
  NP=1  %
*MACRO
  NT=1
  NC=1  X=2  Y=8  NAM=surf  %
*END

**Note:** A zero value for a given criteria means that specific criteria will be immediately satisfied, consequently the output associated with that criteria is meaningless. The output associated with the nonzero criteria are correct.
Resulting yield surface

Equiv. Plastic Strain = 1.0E-05
SCRD = 1.45E-07 Ksi/sec
6.18 Example R: Ellipsoidal Inclusions

Sample Input File for Ellipsoidal Inclusions

Problem Summary

Load Type: Thermomechanical
Load Direction: 33
Load History: Monotonic
Load Control: Strain
Load History Data: Hold temperature during mechanical loading
\[ \varepsilon = 6.667 \times 10^{-5} \text{ /sec}, \]  
\[ \varepsilon_{\text{max}} = 0.01, \varepsilon_{\text{min}} = 0. \]
\[ T = 427 \degree C; \Delta t_{\text{initial}} = 0.01 \text{ sec}. \]

Micromechanics Model: Triple Periodicity
Repeating Unit Cell: Ellipsoidal Inclusion RVE (IDP=4)
\[ a = 1, b = 1, c = 1.3, d/h = 1, l/h = 1 \]
Integration Algorithm: Predictor/Corrector
Constituent Material Model: Fiber: Elastic
Matrix: TGVIPS
Constituents: Fiber: SCS-6
Matrix: Ti-6-4
*PRINT
  NPL=1 %
*LOAD
  LCON=3 LOP=3 LSS=1 %
*MECH
  NPTW=2 TI=0.0, 150.0 LO=0.0, 0.01 %
*THERM
  NPTT=2 TI=0.0, 150.0 TE=427.0, 427.0 %
*MODEL
  MOD=2 %
*SOLVER
  NTF=2 ISTM=0.01 ISTALL=5. ERR=0.01 %
*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=2 MAT=E %
*MATRIX
  NMATX=1
  NM=1 MM=7 NDPT=2 MAT=A D=1.0, 1.0, 1.0 %
*MRVE
  IDP=4
    OPT=2 VF=0.3 A=1 B=1 C=1.3 RD=1 RL=1 %
    # OPT=2 VF=0.3 A=1 B=1 C=1.2 RD=1 RL=1 %
    # OPT=2 VF=0.3 A=1 B=1 C=1 RD=1 RL=1 %
    # OPT=2 VF=0.3 A=1 B=1 C=0.83333 RD=1 RL=1 %
*CURVE
  NP=1 %
*MACRO
  NT=1
  NC=1 X=3 Y=9 NAM=ellipsoid %
*END

* Note: To generate the four curves shown in the plot, change which line does not start with "#" under *MRVE
Results of Discontinuously reinforced composite with 30% volume fraction of ellipsoidal inclusions

**Note:** As the ellipsoidal inclusion becomes longer and thinner (i.e., increasing c) and thus more fiber-like, the response becomes stiffer.
6.19 Example S: PATRAN/MSC MACPOST Output

Sample Input File for PATRAN/MSC MACPOST Output

Problem Summary

Load Type: Thermomechanical
Load Direction: 33
Load History: Monotonic
Load Control: Strain
Load History Data: Cool from 900°C to 23°C, then hold temperature during mechanical loading
\[ \dot{T} = 0.0152 \, ^\circ\text{C}/\text{sec} \]
\[ \dot{\varepsilon} = 1.667 \times 10^{-4} \, /\text{sec}, \, \varepsilon_{\max} = 0.015, \, \varepsilon_{\min} = 0. \]
\[ \Delta t_{\text{thermal}} = 100 \, \text{sec.}; \, \Delta t_{\text{mech}} = 0.1 \, \text{sec}. \]

Micromechanics Model: Double Periodicity
Fiber Packing Arrangement: Square Pack, 40% fiber volume ratio
Repeating Unit Cell: 26x26 circular fiber cross-section approximation
Integration Algorithm: Forward Euler
Constituents:
Fiber: SCS-6 Matrix: TIMETAL21S
PATRAN Output: Start output after cooldown: \( t_{\text{pre}} = 57600 \, \text{sec.} \)
Write to output files every 300 time steps
test of PATRAN output
*PRINT
  NPL=1 %
*LOAD
  LCON=3 LOP=3 LSS=1 %
*MECH
  NPTW=3 TI=0.,57600.,57690. LO=0.,0.,0.015 %
*THERM
  NPTT=3 TI=0.,57600.,57690. TE=900.,23.,23. %
*MODEL
  MOD=1 %
*PATRAN
  FN=apdxs TPRE=57600 STP=300 %
*SOLVER
  NTF=1 NPTS=3 TIM=0.,57600.,57690. STP=100,0.1 %
*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=2 MAT=E %
*MATRIX
  NMATX=1
  NM=1 MM=4 NDPT=2 MAT=A %
*MRVE
  IDP=13 VF=0.4 R=1
*CURVE
  NP=2 %
*MACRO
  NT=1
  NC=1 X=3 Y=9 NAM=apdxs
*END
MAC/GMC: Micromechanics Analysis Code

Macro stress-strain behavior results

Contour of the microlevel $J_2$ invariant at 1.5% macrostrain (© pt. A)
Contour of the microlevel $I_1$ invariant at 1.5% macrostrain (@ pt. A)
6.20 Example T: Effective Thermal Conductivity Calculation

Sample Input File For Effective Thermal Conductivity Calculation

The following example is used to explain the calculation of effective thermal conductivities.

Problem Summary:

Loading: Not Used (nplvl = -1 used to determine effective properties only)

Micromechanics Model: Double Periodicity

Fiber Packing Arrangement: Square Pack, R = 1., 50% fiber volume ratio

Repeating Unit Cell: 7x7 circular fiber cross-section approximation

Integration Algorithm: Not Used

Constituent Material Model: Fiber: Elastic, isotropic

                      Matrix: Bodner-Partom (viscoplastic properties not used)

Constituents: Fiber: Fictitious temperature dependent material (based loosely on tungsten)

                      Matrix: Fictitious temperature dependent material (based loosely on aluminum)
Test of user input properties with thermal conductivity

*PRINT
  NPL=-1%
*LOAD
  LCON=3 LOP=2 LSS=1%
*MECH
  NPTW=3 T1=0.,200.,218. LO=0.,0.,0.015%
*THERM
  NPTT=3 T1=0.,200.,218. TE=600.,21.,21.%
*MODEL
  MOD=1%
*COND
*SOLVER
  NTF=1 NPTS=3 TIM=0.,200.,218 STP=0.2,0.01%
*FIBER
  NFIBS=1
  NF=1 MF=6 NDPT=2 MAT=U IFM=I
  NTP=4
  TEM=18.,200.,400.,600.
  NUA=0.41,0.41,0.41,0.41
  NUT=0.41,0.41,0.41,0.41
  GA=111.38E9,106.38E9,99.29E9,70.92E9
  ALPA=4.5E-6,4.8E-6,5.1E-6,5.5E-6
  ALPT=4.5E-6,4.8E-6,5.1E-6,5.5E-6
  KA=0.2,0.25,0.31,0.44
  KT=0.2,0.25,0.31,0.44
*MATRIX
  NMATX=1
  NM=1 MM=1 NDPT=2 MAT=U IFM=I
  NTP=4
  TEM=18.,200.,400.,600.
  NUA=0.33,0.33,0.33,0.33
  NUT=0.33,0.33,0.33,0.33
  GA=27.07E9,25.19E9,20.68E9,12.41E9
  ALPA=22. E-6,24. E-6,28. E-6,33. E-6
  ALPT=22. E-6,24. E-6,28. E-6,33. E-6
  V1=1.E4, 1.E4, 1.E4, 1.E4
  V4=300.0, 300.0, 300.0, 300.0
  V5=10.0, 4.0, 1.6, 0.55
  V6=1.0, 1.0, 1.0, 1.0

220
Thermal conductivity results from outfile:

<table>
<thead>
<tr>
<th>Temperature</th>
<th>K Axial</th>
<th>K Transverse</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.0</td>
<td>0.201E+02</td>
<td>0.889E+01</td>
</tr>
<tr>
<td>200.0</td>
<td>0.191E+02</td>
<td>0.852E+01</td>
</tr>
<tr>
<td>400.0</td>
<td>0.177E+02</td>
<td>0.795E+01</td>
</tr>
<tr>
<td>600.0</td>
<td>0.147E+02</td>
<td>0.683E+01</td>
</tr>
</tbody>
</table>
This Page Intentionally Left Blank
## INDEX

### Keywords

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>*COND</td>
<td>67, 71</td>
</tr>
<tr>
<td>*CURVE</td>
<td>99</td>
</tr>
<tr>
<td>*DAMAGE</td>
<td>60, 69</td>
</tr>
<tr>
<td>*DEBOND</td>
<td>95</td>
</tr>
<tr>
<td>*FIBER</td>
<td>68, 73, 74</td>
</tr>
<tr>
<td>*INTERFACE</td>
<td>94</td>
</tr>
<tr>
<td>*LOAD</td>
<td>54, 55</td>
</tr>
<tr>
<td>*MACRO</td>
<td>100</td>
</tr>
<tr>
<td>*MATRIX</td>
<td>68, 74</td>
</tr>
<tr>
<td>*MECH</td>
<td>57</td>
</tr>
<tr>
<td>*MICRO</td>
<td>101</td>
</tr>
<tr>
<td>*MODEL</td>
<td>63</td>
</tr>
<tr>
<td>*MONOL</td>
<td>68</td>
</tr>
<tr>
<td>*MRVE</td>
<td>77</td>
</tr>
<tr>
<td>*PATRAN</td>
<td>64</td>
</tr>
<tr>
<td>*PRINT</td>
<td>53</td>
</tr>
<tr>
<td>*SOLVER</td>
<td>66</td>
</tr>
<tr>
<td>*SURF</td>
<td>62</td>
</tr>
<tr>
<td>*THERM</td>
<td>58</td>
</tr>
</tbody>
</table>

### Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>% symbol</td>
<td>52</td>
</tr>
<tr>
<td>&quot;#&quot; symbol</td>
<td>52</td>
</tr>
</tbody>
</table>

### A

<table>
<thead>
<tr>
<th>Alias</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABAQUS</td>
<td>181</td>
</tr>
<tr>
<td>Adams Bashforth</td>
<td>14</td>
</tr>
<tr>
<td>Adams-Moulton</td>
<td>15</td>
</tr>
<tr>
<td>ANG</td>
<td>63, 69</td>
</tr>
<tr>
<td>Anisotropic Elastic</td>
<td>17</td>
</tr>
</tbody>
</table>

### B

<table>
<thead>
<tr>
<th>Alias</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCN</td>
<td>95</td>
</tr>
<tr>
<td>BCS</td>
<td>95</td>
</tr>
<tr>
<td>BDN</td>
<td>95</td>
</tr>
<tr>
<td>BDS</td>
<td>95</td>
</tr>
<tr>
<td>Biaxial Load Problem</td>
<td>135</td>
</tr>
<tr>
<td>BN</td>
<td>69</td>
</tr>
<tr>
<td>Bodner-Partom</td>
<td>18</td>
</tr>
<tr>
<td>BP</td>
<td>69</td>
</tr>
</tbody>
</table>

### C

<table>
<thead>
<tr>
<th>Alias</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>62</td>
</tr>
<tr>
<td>C2</td>
<td>62</td>
</tr>
<tr>
<td>C3</td>
<td>62</td>
</tr>
<tr>
<td>C4</td>
<td>62</td>
</tr>
</tbody>
</table>

### D

<table>
<thead>
<tr>
<th>Alias</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>60, 68</td>
</tr>
<tr>
<td>Database</td>
<td>72</td>
</tr>
<tr>
<td>DBCH</td>
<td>95</td>
</tr>
<tr>
<td>debonding</td>
<td>42, 43</td>
</tr>
<tr>
<td>Discrete Fiber Breakage Problem</td>
<td>168</td>
</tr>
<tr>
<td>DMAX</td>
<td>60</td>
</tr>
<tr>
<td>double periodicity</td>
<td>63</td>
</tr>
</tbody>
</table>

### E

<table>
<thead>
<tr>
<th>Alias</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effective Thermal Conductivities</td>
<td>219</td>
</tr>
<tr>
<td>Ellipsoidal</td>
<td>86</td>
</tr>
<tr>
<td>Ellipsoidal Inclusion Problem</td>
<td>212</td>
</tr>
<tr>
<td>engineering strains</td>
<td>36</td>
</tr>
<tr>
<td>ERR</td>
<td>66</td>
</tr>
<tr>
<td>ETU</td>
<td>69</td>
</tr>
<tr>
<td>Examples</td>
<td>105</td>
</tr>
<tr>
<td>execution of MAC/GMC</td>
<td>46</td>
</tr>
<tr>
<td>- large finite element problems</td>
<td>49</td>
</tr>
</tbody>
</table>

### F

<table>
<thead>
<tr>
<th>Alias</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>FACE</td>
<td>95</td>
</tr>
<tr>
<td>Fatigue</td>
<td>60</td>
</tr>
<tr>
<td>fatigue damage</td>
<td>30</td>
</tr>
<tr>
<td>Fatigue Damage Analysis</td>
<td>156</td>
</tr>
<tr>
<td>FEAMAC</td>
<td>181</td>
</tr>
<tr>
<td>FEAMAC Analysis Example</td>
<td>181</td>
</tr>
<tr>
<td>FEAMAC Input Deck</td>
<td>49</td>
</tr>
<tr>
<td>FEAMAC Output Files</td>
<td>50</td>
</tr>
<tr>
<td>FG</td>
<td>60</td>
</tr>
<tr>
<td>fiber and matrix constituents</td>
<td>68</td>
</tr>
<tr>
<td>finite element implementation</td>
<td>47</td>
</tr>
<tr>
<td>FL</td>
<td>60</td>
</tr>
<tr>
<td>FN</td>
<td>64</td>
</tr>
<tr>
<td>Forward Euler</td>
<td>14</td>
</tr>
</tbody>
</table>

### G

<table>
<thead>
<tr>
<th>Alias</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>gauss quadrature points</td>
<td>28</td>
</tr>
<tr>
<td>GCN</td>
<td>95</td>
</tr>
<tr>
<td>MAC/GMC: Micromechanics Analysis Code</td>
<td></td>
</tr>
<tr>
<td>---------------------------------------</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>General Loading Problem</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generalized loading</td>
<td>54</td>
</tr>
<tr>
<td>Generalized method of cells</td>
<td>3, 7</td>
</tr>
<tr>
<td>GVIPS</td>
<td>22</td>
</tr>
</tbody>
</table>

| N | NA | 87 |
|   | NAI | 95 |
|   | NAM | 100, 101 |
|   | NB | 87 |
|   | NBI | 95 |
|   | NC | 100, 101 |
|   |NCY | 60 |
|   | NDPT | 68, 94 |
|   | NF | 68 |
|   | NFIBS | 68 |
|   | NG | 87 |
|   | NGI | 95 |
|   | NI | 94 |
|   | NII | 95 |
|   | NINT | 94 |
|   | NLY | 63 |
|   | NM | 68 |
|   | NMATX | 68 |
|   | NMO | 68 |
|   | NMON | 68 |
| nonisothermal | 73 |
| Non-Symmetric Laminate | 205 |
| NP | 99 |
| NPL | 53 |
| NPRE | 62 |
| NPTS | 66 |
| NPTT | 58 |
| NPTW | 57 |
| NT | 101 |
| NTF | 66 |

numbering scheme 103 |

| O | OMFL | 69 |
|   | OMM | 69 |
|   | OMU | 69 |

| P | packing | 77 |
|   | Plain Weave Composites | 174 |
|   | predictor/corrector | 14 |

| R | R | 77 |
|   | R = X/Y | 78 |
|   | RAD | 77 |
|   | RAD1 | 77 |
|   | RAD2 | 77 |

| M | MACPOST Example | 215 |
|   | MAT | 68, 69, 94 |
| Material Constant Database | 72 |
| matrix | 68 |
| MATSYS | 63 |
| Mechanical Load Problem | 105 |
| MF | 68 |
| MI | 94 |
| micromechanics | 7 |
| MM | 68 |
| MMO | 68 |
| MOD | 63, 85 |
| monolithic material | 68 |
| MS | 68 |

224
MAC/GMC: Micromechanics Analysis Code

<table>
<thead>
<tr>
<th>Term</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>reformulation of GMC</td>
<td>11</td>
</tr>
<tr>
<td>residual stresses</td>
<td>111</td>
</tr>
<tr>
<td>RN</td>
<td>95</td>
</tr>
<tr>
<td>Robinson Viscoplastic</td>
<td>20</td>
</tr>
<tr>
<td>RS</td>
<td>95</td>
</tr>
<tr>
<td>Runge-Kutta</td>
<td>14</td>
</tr>
<tr>
<td>S</td>
<td></td>
</tr>
<tr>
<td>SFL</td>
<td>69</td>
</tr>
<tr>
<td>SK</td>
<td>69</td>
</tr>
<tr>
<td>state variables</td>
<td>183</td>
</tr>
<tr>
<td>STO</td>
<td>64</td>
</tr>
<tr>
<td>STP</td>
<td>64, 66</td>
</tr>
<tr>
<td>SU</td>
<td>69</td>
</tr>
<tr>
<td>Subcell</td>
<td>103</td>
</tr>
<tr>
<td>SYS</td>
<td>63</td>
</tr>
<tr>
<td>T</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>60</td>
</tr>
<tr>
<td>TE</td>
<td>58</td>
</tr>
<tr>
<td>temperature</td>
<td>58</td>
</tr>
<tr>
<td>TGVIPS</td>
<td>24</td>
</tr>
<tr>
<td>thermal conductivities</td>
<td>74</td>
</tr>
<tr>
<td>thermal conductivity</td>
<td>67</td>
</tr>
<tr>
<td>Thermal Load Problem</td>
<td>108</td>
</tr>
<tr>
<td>Thermomechanical Load</td>
<td>111</td>
</tr>
<tr>
<td>THK</td>
<td>63</td>
</tr>
<tr>
<td>TI</td>
<td>57, 58, 95</td>
</tr>
<tr>
<td>TIM</td>
<td>66</td>
</tr>
<tr>
<td>TOLN</td>
<td>95</td>
</tr>
<tr>
<td>Transverse Debonding</td>
<td>114</td>
</tr>
<tr>
<td>Triple Periodic GMC</td>
<td>129</td>
</tr>
<tr>
<td>triple periodicity</td>
<td>63</td>
</tr>
<tr>
<td>U</td>
<td></td>
</tr>
<tr>
<td>user defined architecture</td>
<td>132</td>
</tr>
<tr>
<td>User Defined Example</td>
<td>138</td>
</tr>
<tr>
<td>User Defined RVE Problem</td>
<td>132</td>
</tr>
<tr>
<td>User Supplied Isothermal</td>
<td>73</td>
</tr>
<tr>
<td>User Supplied Nonthermal</td>
<td>73, 74</td>
</tr>
<tr>
<td>USRELS</td>
<td>37</td>
</tr>
<tr>
<td>USRMAT</td>
<td>34, 42</td>
</tr>
<tr>
<td>V</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>60</td>
</tr>
<tr>
<td>VF</td>
<td>77</td>
</tr>
<tr>
<td>VF1</td>
<td>77</td>
</tr>
<tr>
<td>VF2</td>
<td>77</td>
</tr>
</tbody>
</table>

Yield Surface Plot 100, 101

225
reformulation of GMC 11
residual stresses 11
RN 95
Robinson Viscoelastic 20
RS 95
Runge-Kutta 14
S
SFL 69
SK 69
state variables 183
STO 64
STP 64, 66
SU 69
Subcell 103
SYS 63
T
T 60
TE 58
temperature 58
TGVIPS 24
thermal conductivities 74
thermal conductivity 67
Thermal Load Problem 108
Thermomechanical Load 111
THK 63
TI 57, 58, 95
TIM 66
TOLN 95
Transverse Debonding 114
Triple Periodic GMC 129
triple periodicity 63
U
user defined architecture 132
User Defined Example 138
User Defined RVE Problem 132
User Supplied Isothermal 73
User Supplied NonIsothermal 73, 74
USRELS 37
USRMAT 34, 42
V
V 60
VF 77
VF1 77
VF2 77
Y
Yield Surface Plot 209
X
X 100, 101
XA 77
XML 69
225
4. TITLE AND SUBTITLE
   Micromechanics Analysis Code With Generalized Method of Cells
   (MAC/GMC)
   User Guide: Version 3.0

6. AUTHOR(S)
   S.M. Arnold, B.A. Bednarcyk, T.E. Wilt, and D. Trowbridge

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
   National Aeronautics and Space Administration
   John H. Glenn Research Center at Lewis Field
   Cleveland, Ohio 44135–3191

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
   National Aeronautics and Space Administration
   Washington, DC 20546–0001

11. SUPPLEMENTARY NOTES
   S.M. Arnold, NASA Glenn Research Center; B.A. Bednarcyk, Ohio Aerospace Institute, 22800 Cedar Point Road, Cleveland, Ohio 44142; T.E. Wilt and D. Trowbridge, The University of Akron, Akron, Ohio 44325–0001. Responsible person, S.M. Arnold, organization code 5920, (216) 433-3334.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
    Unclassified - Unlimited
    Subject Categories: 24 and 39
    Distribution: Nonstandard

13. ABSTRACT (Maximum 200 words)
    The ability to accurately predict the thermomechanical deformation response of advanced composite materials continues to play an important role in the development of these strategic materials. Analytical models that predict the effective behavior of composites are used not only by engineers performing structural analysis of large-scale composite components but also by material scientists in developing new material systems. For an analytical model to fulfill these two distinct functions it must be based on a micromechanics approach which utilizes physically based deformation and life constitutive models and allows one to generate the average (macro) response of a composite material given the properties of the individual constituents and their geometric arrangement. Here the user guide for the recently developed, computationally efficient and comprehensive micromechanics analysis code, MAC, who's predictive capability rests entirely upon the fully analytical generalized method of cells, GMC, micromechanics model is described. MAC/GMC is a versatile form of research software that "drives" the double or triply periodic micromechanics constitutive models based upon GMC. MAC/GMC enhances the basic capabilities of GMC by providing a modular framework wherein 1) various thermal, mechanical (stress or strain control) and thermomechanical load histories can be imposed, 2) different integration algorithms may be selected, 3) a variety of material constitutive models (both deformation and life) may be utilized and/or implemented, and 4) a variety of fiber architectures (both unidirectional, laminate and woven) may be easily accessed through their corresponding representative volume elements contained within the supplied library of RVEs or input directly by the user, and 5) graphical post processing of the macro and/or micro field quantities is made available.

17. SECURITY CLASSIFICATION OF REPORT
    Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE
    Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT
    Unclassified

20. LIMITATION OF ABSTRACT
    Standard Form 298 (Rev. 2-89)
    Prescribed by ANSI Std. 239-18
    298-102