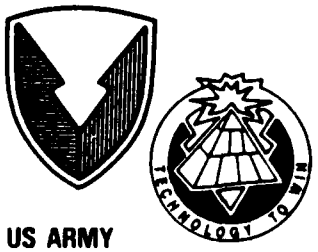
(2)

MTL TR 91-20

EXPERIMENTAL EVALUATION OF THE TAYLOR-TYPE POLYCRYSTAL MODEL FOR THE FINITE DEFORMATION OF AN FCC METAL (OFHC COPPER)

TUSIT WEERASOORIYA and RONALD A. SWANSON
MATERIALS DYNAMICS BRANCH

May 1991



US ARMY
LABORATORY COMMAND
MATERIALS TECHNOLOGY LABORATORY

U.S ARMY MATERIALS TECHNOLOGY LABORATORY
Watertown, Massachusetts 02172-0001

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

Mention of any trade names or manufacturers in this report shall not be construed as advertising nor as an official indorsement or approval of such products or companies by the United States Government.

DISPOSITION INSTRUCTIONS

Destroy this report when it is no longer needed.
Do not return it to the originator

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER MTL TR 91-20	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) EXPERIMENTAL EVALUATION OF THE TAYLOR-TYPE POLYCRYSTAL MODEL FOR THE FINITE DEFORMATION OF AN FCC METAL (OFHC COPPER)		5. TYPE OF REPORT & PERIOD COVERED Final Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Tusit Weerasooriya and Ronald A. Swanson		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U.S. Army Materials Technology Laboratory Watertown, Massachusetts 02172-0001 SLCMT-MRD		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS D/A Project: 611102.H420011
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Laboratory Command 2800 Powder Mill Road Adelphi, Maryland 20783-1145		12. REPORT DATE May 1991 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Polycrystal	Taylor model	Rate dependent
Finite deformation	Constitutive modeling	Texture evaluation
OFHC copper	FCC metal	Torsion

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

(SEE REVERSE SIDE)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

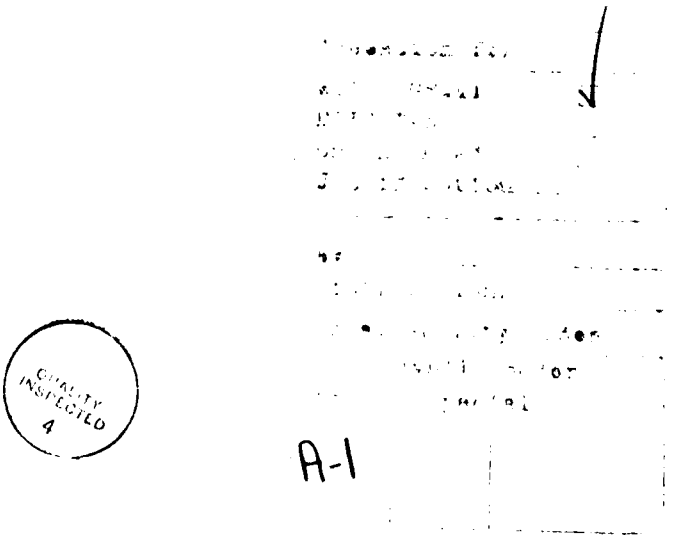
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block No. 20

ABSTRACT

Large deformation uniaxial compression and fixed-end torsion (simple shear) experiments were conducted on annealed OFHC Copper to obtain its stress-strain behavior. This material behavior was also predicted using the Taylor-type rate dependent polycrystal model by Asaro and Needleman. Simulations of the experiments were conducted in personal computers, using a recently developed, highly efficient, fully implicit, time integration scheme by Kalidindi et al. In the initial phase of the simulation, the evolution of the constituent single crystal slip system deformation resistance was estimated using the experimentally determined compressive stress-strain behavior of the polycrystal. With this crystal constitutive behavior, the stress-strain behavior of the polycrystal for simple shear was computed. In addition, the evolution of the crystallographic texture was computed for both compression and simple shear tests. Both the shear and axial stress-strain behaviors for the simple shear test compared reasonably well with the experimental results.

CONTENTS



A-1

INTRODUCTION

In an event such as the penetration of an armor, the armor material undergoes a large deformation at a high strain rate and elevated temperature before failure. When numerically simulating such a rate dependent finite deformation event, predictions significantly depend on the constitutive models used to represent the plastic deformation behavior of the material. Most computer codes contain constitutive models that are extensions of the small deformation representation. In most of these codes, extension to multiaxiality is achieved through the assumption that there exists a unique effective stress-strain behavior for a particular material at a given strain rate and temperature. Finite strain deformation experiments show this assumption is not valid for large deformations. That is, tension, compression, and torsion experimental data for large deformation cannot be correlated by an effective stress-strain representation[1] (also from the experiments reported in this report). The torsional effective stress-strain curve lies significantly below both the tensile and the compressive stress-strain curves at large strains. Therefore to obtain a better prediction of the deformation at finite strain, it is necessary that the constitutive models in computer codes follow the response of the material at finite deformation more accurately.

To improve upon the extension of the small deformation classical model (Prandtl-Reuss), finite-deformation behavior has been modeled in two different ways: 1) Macroscopic phenomenological modeling. Here, functional forms for plastic stretching, plastic spin and back stress tensors are assumed. 2) Microscopic (polycrystal) phenomenological modeling. In this case, the phenomenological representation is of the single crystal slip behavior; the macroscopic plastic constitutive behavior is derived from it. Also the evolution of the crystallographic texture (crystallographic anisotropy) can be deduced from this type of formulation. In addition, with the insight into the micro-mechanics of the plastic deformation process, the polycrystal modeling can lead us to select functional forms for macroscopic modeling that will better represent the physics of microscopic plastic deformation.

In the polycrystal phenomenological plasticity modeling, three methods have been used to deduce the polycrystal constitutive behavior from the deformation of the individual crystals (grains): 1) The Taylor method,[2] where each grain is assumed to deform with the same deformation gradient as the polycrystal, thus satisfying compatibility among the grains, but violating the stress equilibrium across the grains. 2) Sachs method,[3] where each grain is assumed to be subjected to the same stress tensor as the polycrystal, thus satisfying the stress equilibrium condition, but violating the compatibility across the grains. 3) Self-consistent method, where each grain is considered as an inclusion embedded in a homogeneous infinitely extended matrix.[4,5,6] Unfortunately, there are not that many experimental evaluation of these polycrystal schemes in the literature.

1. SEMIATIN, S. L., LAHOTI, G. D. AND JONAS, J. J. Mechanical Testing, Metals Handbook, Ninth Editon, v. 8, p. 164.
2. TAYLOR, G. I. J. *Plastic Strain in Metals.* Inst. Metals, v. 62, 1938, p. 307.
3. SACHS, G., Z. Ver. Dtsch. Ing., v. 72, 1928, p. 734.
4. KRONER, E. *Berechning der elastischen Konstanten des Vielkristalls aus den Konstanten des Einkristalls.* Z. Phys., v. 151, 1958, p.504-518.
5. BUDIANSKY, B. AND WU, T. T. *Theoretical prediction of plastic strains of polycrystals.* In Proc. of the 4th U. S. National Congress of Applied Mechanics, ASME, New York, 1962, p. 1175-1185.
6. HILL, R. *Continuum micro-mechanics of elastoplastic polycrystals.* J. Mech. Phys. Solids, v. 13, 1965, p. 89-101.

For elastic-viscoplastic finite deformation via slip by dislocation glide, Asaro and Needleman[7] have proposed a Taylor-type model to represent the polycrystal behavior from its individual crystals. Harren et al.[8] have conducted a detailed analysis of the finite deformation under shear (torsion) using this model. They have observed that the axial behavior of a single phase polycrystalline material in a torsion test is very sensitive to strain hardening, strain rate sensitivity, and latent hardening behavior of the constituent single crystal. Therefore, finite deformation torsion testing can be used to evaluate and calibrate polycrystal plastic models as well as macroscopic models.

Four types of experiments are commonly used to evaluate plastic constitutive behavior of materials: 1) uniaxial tensile testing, 2) uniaxial compression testing, 3) thin walled torsion testing, and 4) multiaxial testing. With tensile testing, it is not possible to obtain experimental data for large deformations due to the necking instability. Compression testing can be used to obtain large deformation data up to strains of approximately 1.5. Though multiaxial testing is useful to study constitutive models, it is complex to perform, especially to large strains. However, with torsion testing, it is possible to obtain experimental finite deformation data up to very large strains. In addition, axial stress or strain measurement from the torsion tests, depending on whether these tests are conducted under fixed-end or free-end conditions, is a very sensitive measure and can be used to evaluate both polycrystal and macroscopic finite deformation constitutive models. However, there is a lack of torsion test data with axial stress or strain measurements in the literature.

A set of finite deformation experiments (fixed and free-end thin wall torsion and uniaxial compression) were conducted for the experimental evaluation of the finite deformation models. This report gives an evaluation of the Taylor-type rate dependent model proposed by Asaro and Needleman[7] using the experimental stress-strain behavior from these experiments. The predicted textures from the rate dependent Taylor model are also given in this report. Comparison of the predicted textures with the experimental textures will be reported in a future publication.

THE TAYLOR-TYPE POLYCRYSTAL MODEL

An overview of the Taylor-type rate dependent polycrystal model by Asaro and Needleman is given in this section. In our work, we followed the formulation of the time integration of the polycrystal model for various boundary conditions as proposed by Kalidindi et al.[9] The summary of their integration algorithm is also given in this section.

---

7. ASARO, R. J. AND NEEDLEMAN, A. *Texture Development and Strain Hardening in Rate Dependent Polycrystals.* Acta Metall., v. 33, 1985, p. 923.
8. HARREN, S., LOWE, T. C., ASARO, R. J. AND NEEDLEMAN, A. *Analysis of large-strain shear in rate-dependent face-centred cubic polycrystals: correlation of micro- and macromechanics,* Phil. Trans. R. Soc., v. 328, 1989, p. 443-500.
9. KALIDINDI, S. R., BRONKHORST, C. A. AND ANAND, L. *Crystallographic Texture Evolution in Bulk Deformation Processing of FCC Metals.* to appear in Journal of the Mechanics and Physics of Solids.

With this formulation, the problem could be solved with PCs due to the efficiency of the algorithm. The disadvantage of this formulation is that it can be applied only with the prescribed displacement boundary conditions.

Kinematics of the Single Crystal

Let $\mathbf{F}$ be the global deformation gradient of the polycrystalline material. With the Taylor assumption, then the deformation gradient of each crystal of the polycrystal will have the same deformation gradient $\mathbf{F}$. By multiplicative decomposition of the deformation gradient into elastic and plastic components, we obtain:

$$\mathbf{F} = \mathbf{F}^e\mathbf{F}^p \tag{1}$$

where $\mathbf{F}^e$ is the elastic and $\mathbf{F}^p$ is the plastic parts of the deformation gradient. Here $\mathbf{F}^p$ is the deformation gradient due to plastic slip only ($\det \mathbf{F}^p = 1$); it represents an elastically unloaded intermediate relaxed configuration of a stress free state with the orientation of the crystal lattice coinciding with the crystal lattice in the reference configuration. The deformation gradient $\mathbf{F}^e$ represents both rotation and elastic stretching of the crystal lattice ($\det \mathbf{F}^e > 1$). Then the velocity gradient $\mathbf{L}$:

$$\mathbf{L} = \dot{\mathbf{F}}\,\mathbf{F}^{-1} = \dot{\mathbf{F}}^e\,\mathbf{F}^{e-1} + \mathbf{F}^e\,\mathbf{L}^g\,\mathbf{F}^{e-1} \tag{2}$$

where
$$\mathbf{L}^g = \dot{\mathbf{F}}^p\,\mathbf{F}^{p-1} \tag{3a}$$

or
$$\dot{\mathbf{F}}^p = \mathbf{L}^g\,\mathbf{F}^p \tag{3b}$$

Here the notation ($\dot{}$) indicates the derivative with respect to time. Since $\mathbf{L}^g$ is the plastic velocity gradient in the relaxed configuration, $\mathbf{L}^g$ is given by

$$\mathbf{L}^g = \sum_{\alpha=1}^{12} \dot{\gamma}^\alpha\,\mathbf{S}^g, \qquad \mathbf{S}^g = \mathbf{s}^g \otimes \mathbf{n}^g \tag{3c}$$

where $\mathbf{s}^g$ and $\mathbf{n}^g$ are orthonormal unit vectors representing slip directions and slip planes, respectively, for each slip system α in the reference configuration and the $\dot{\gamma}^\alpha$ is the plastic shearing rate in the αth slip system. For an fcc crystal there are twelve slip systems derived from its four {111} slip planes and three <110> slip directions for each slip plane. Consequently, unit vectors $\mathbf{n}^g$ represent four {111} planes and the unit vectors $\mathbf{s}^g$ represent the corresponding three <110> directions of each αth slip system (in the Table of section Model Predictions). From Equation (2),

$$\mathbf{L}^p = \mathbf{D}^p + \mathbf{W}^p = \mathbf{F}^e\,\mathbf{L}^g\,\mathbf{F}^{e-1} = \sum_{\alpha=1}^{12} \dot{\gamma}^\alpha\,\mathbf{S}^\alpha, \quad \mathbf{S}^\alpha = (\mathbf{F}^e\,\mathbf{s}^g) \otimes (\mathbf{n}^g\,\mathbf{F}^{e-1}) \tag{4a}$$

where $\mathbf{D}^p$ and $\mathbf{W}^p$ are plastic stretch and spin tensors, respectively, and are given by

$$\mathbf{D}^p = \sum_{\alpha=1}^{12} \dot{\gamma}^\alpha \ \mathrm{sym}\{\mathbf{S}^\alpha\}, \qquad \text{and} \qquad \mathbf{W}^p = \sum_{\alpha=1}^{12} \dot{\gamma}^\alpha \ \mathrm{asym}\{\mathbf{S}^\alpha\} \tag{4b}$$

The rate of plastic stress power per unit reference volume is (see Anand[10])

$$\mathring{\omega}^p = (\mathbf{C}^e \mathbf{T}^*):\mathbf{L}^p, \qquad \text{with } \mathbf{C}^e = \mathbf{F}^{eT}\mathbf{F}^e \tag{5}$$

and

$$\mathbf{T}^* = \mathbf{F}^{e-1}\{(\det \mathbf{F}e)\,\mathbf{T}\}\,\mathbf{F}^{e-T}. \tag{6}$$

The stress **T** is Cauchy stress of the crystal. The second Piola-Kirchhoff stress, **T***, is the elastic work conjugate of the Lagrangean elastic strain measure

$$\mathbf{E}^e = (1/2)\{\mathbf{C}^e - \mathbf{I}\}, \qquad \text{with } \mathbf{C}^e = \mathbf{F}^{eT}\mathbf{F}^e \tag{7}$$

But the plastic stress power per unit volume is

$$\mathring{\omega}^p = \sum_{\alpha=1}^{12} \tau^\alpha \dot{\gamma}^\alpha \tag{8}$$

where τ^α is the resolved shear stress in the αth slip system. Therefore, from Equations (5) and (8) resolved shear stress

$$\tau^\alpha = (\mathbf{C}^e \mathbf{T}^*):\mathbf{S}^\alpha \tag{9}$$

Crystal Plasticitv Constitutive Model

The relationship between the resolved shear stress, τ^α, and resolved shear strain rate, $\dot{\gamma}_0$, is given by the viscoplastic rate power law

$$\dot{\gamma}^\alpha = \dot{\gamma}_0 \left| \tau^\alpha / s^\alpha \right|^{1/m} \mathrm{sign}(\tau^\alpha) \tag{10}$$

where $\dot{\gamma}_0$ is the reference shear strain rate, s^α is the resistance of the slip system α and m is the strain rate sensitivity parameter (assumed to be the same for all slip systems). Also, the slip resistance s^α can be considered as the slip resistance of the slip system α at the reference shearing rate, $\dot{\gamma}_0$.

The slip resistance s^α is evolved according to

10. ANAND, L. *Constitutive Equations for Hot-Working of Metals.* Int. J. Plast., v. 1, 1985, p. 213-231.

$$\overset{\circ}{s}^{\alpha} = \sum_{\beta=1}^{12} h^{\alpha\beta} \, | \overset{\circ}{\gamma}^{\beta} | \tag{11}$$

where $h^{\alpha\beta}$ is the instantaneous hardening modulus of the slip system α due to a shearing on the slip system β. The form of the hardening moduli (Asaro and Needleman) are given by

$$h^{\alpha\beta} = h(s^{\beta}) \, q^{\alpha\beta} \qquad \text{(no sum of } \beta) \tag{12}$$

where $q^{\alpha\beta}$ is a 12 by 12 matrix describing the relationship between self and latent hardening behavior of the crystal slip systems. The ratio of latent hardening to self hardening is assumed to be unity for coplanar and a value of q for non-coplanar systems. From Kalidindi et al. the single slip hardening modulus, h(s), is assumed to be given by

$$h(s) = h_0 \left(1 - \frac{s}{s_s} \right)^a \tag{13}$$

where constants h_0, s_s and a are assumed to be the same for all slip systems. Here, s_s represents the saturated slip resistance and h_0 is a measure of the initial hardening modulus of the single slip system. From this assumed single slip hardening behavior

$$| \gamma | = \frac{s_s}{(1+a)} \left[\left(1 - \frac{s}{s_s} \right)^{-(a+1)} - \left(1 - \frac{s_0}{s_s} \right)^{-(a+1)} \right] \tag{14}$$

for $a \neq -1$, where s_0 is the initial slip resistance and the $| \gamma |$ is the absolute value of shear. Here, the Bauschinger effect is assumed to be negligible.

From finite elasticity, the reduced constitutive equation for an elastic material in a grain can be written in the form

$$\mathbf{T}^* = \mathcal{L}[\mathbf{E}^e] \tag{15}$$

where $\mathcal{L}$ is the fourth order elastic tensor. Assuming all the grains are of equal volume, with the Taylor assumption that the local deformation gradient of each grain is the same as the global deformation gradient, then the volume average Cauchy stress $\mathbf{T}^G$ is

$$\mathbf{T}^G = \frac{1}{N} \sum_{k=1}^{N} \mathbf{T}^k \tag{16}$$

where $\mathbf{T}^k$ is the Cauchy stress of each grain and N is the number of grains. The volume average stress $\mathbf{T}^G$ is the global stress response of the polycrystal (Asaro and Needleman).

In the reformulation of the Asaro-Needleman polycrystal model for a more efficient time-integration scheme, Kalidindi et al.[9] have assumed the following approximations:

For small elastic stretches:

$$\mathcal{L} \equiv 2\mu \mathbf{I} + (\kappa - \frac{2\mu}{3}) \mathbf{1} \otimes \mathbf{1} \tag{17}$$

and

$$\tau^{\alpha} \approx \mathbf{T}^{*} : \mathbf{S}^{\alpha} \tag{18}$$

Numerical Procedure for Model Predictions

In the simulation of the experiments reported in this report, the time-integration procedure proposed by Kalidindi et al. was used to integrate the above polycrystal model. This formulation of the time-integration procedure is summarized in this section.

The problem of integrating the above polycrystal model can be summarized as follows. Let τ be the time Δt later than the current time t. Then the problem is to find the list $\{\mathbf{F}^{p}(\tau), s^{\alpha}(\tau), \mathbf{T}(\tau)\}$ for each grain at time τ when the current state $\{\mathbf{F}^{p}(t), s^{\alpha}(t), \mathbf{T}(t)\}$ of all the grains and the global deformation gradients at times t and τ are given. Once the stress at each grain is known, global stress behavior at τ can be calculated from Equation (16). The new texture at time τ for each crystal can be computed from (see Equation 4a)

$$\mathbf{s}^{\mathfrak{P}} = \mathbf{F}^{e}(\tau)\, \mathbf{s}^{\mathfrak{G}} \tag{19}$$

$$\mathbf{n}^{\mathfrak{P}} = \mathbf{F}^{e\text{-}T}(\tau)\, \mathbf{n}^{\mathfrak{G}} \tag{20}$$

where $(\mathbf{s}^{\mathfrak{P}}, \mathbf{n}^{\mathfrak{P}})$ is the slip system in the deformed configuration. Since the slip systems in the undeformed configuration $(\mathbf{s}^{\mathfrak{G}}, \mathbf{n}^{\mathfrak{G}})$ are orthonormal, the slip systems in the deformed configuration are also orthonormal. If $\mathbf{F}^{e}(\tau)$ is found, then the slip system in the deformed configuration for each crystal at the time τ can be computed and hence the pole figure can be constructed for the polycrystal.

The starting point for the formulation of the time integration procedure is the result obtained by Weber and Anand[11] in the implicit time-integration of the evolution equation for $\hat{\mathbf{F}}^{p}$ given in the Equation (3b):

$$\mathbf{F}^{p}(\tau) = \exp\{\Delta t\, \mathbf{L}^{p}(\tau)\}\, \mathbf{F}^{p}(t) \tag{21}$$

11. WEBER, G. AND ANAND, L. Finite Deformation Constitutive Equations and a Time Integration Procedure for Isotropic, Hyperelastic-Viscoplastic Solids. Comp. Methods Appl. Mech. Eng., v. 79, 1990, p. 173-202.

By neglecting higher order terms of $\Delta\gamma^\alpha$ for small $\Delta\gamma^\alpha$ and using Equation (3c)

$$\mathbf{F}^p(\tau) = \{\mathbf{I} + \sum_{\alpha=i}^{12} \Delta\gamma^\alpha \mathbf{S}^\alpha\} \mathbf{F}^p(t) \tag{22a}$$

or

$$\mathbf{F}^{p-1}(\tau) = \mathbf{F}^{p-1}(t) \{\mathbf{I} - \sum_{\alpha=1}^{12} \Delta\gamma^\alpha \mathbf{S}^\alpha\} \tag{22b}$$

where

$$\Delta\gamma^\alpha = \dot{\gamma}^\alpha(\tau)\, \Delta t. \tag{23}$$

From Equations (23), (18) and (10) $\Delta\gamma^\alpha$ is given by

$$\Delta\gamma^\alpha = \dot{\gamma}_o\, \Delta t. \left| \frac{\mathbf{T}^*:\mathbf{S}^\alpha}{s^\alpha} \right|^{1/m} \text{sign}(\mathbf{T}^*:\mathbf{S}^\alpha) \tag{24}$$

Substituting Equations (7) and (1) in the elastic constitutive Equation (15), following equations are obtained (see Kalidindi et al.[7] for details):

$$\mathbf{T}^*(\tau) = \mathbf{T}^{*tr} - \sum_{\alpha=1}^{12} \Delta\gamma^\alpha \mathbf{C}^\alpha \tag{25}$$

$$s^\alpha(\tau) = s^\alpha(t) + \sum_{\beta=1}^{12} h^{\alpha\beta} \left| \Delta\gamma^\beta \right| \tag{26}$$

where

$$\mathbf{T}^{*tr} = \mu\, \mathbf{A} + \frac{3\kappa - 2\mu}{6} (\text{tr}\, \mathbf{A})\, \mathbf{1} - \frac{3\kappa}{2} \mathbf{1}, \tag{27}$$

$$\mathbf{A} = \mathbf{F}^{p-T}(t)\, \mathbf{F}^T(\tau)\, \mathbf{F}(\tau)\, \mathbf{F}^{p-1}(t), \tag{28}$$

$$\mathbf{C}^\alpha = \mu\, \mathbf{B}^\alpha + \frac{3\kappa - 2\mu}{6} (\text{tr}\mathbf{B}^\alpha)\, \mathbf{1}, \tag{29}$$

$$\mathbf{B}^\alpha = \mathbf{A}\, \mathbf{S}^\alpha + \mathbf{S}^\alpha{}^T \mathbf{A}, \tag{30}$$

Equations (25) and (26) are two sets of nonlinear simultaneous equations ($\mathbf{T}^*(\tau)$ - six unknowns - and $s^\alpha(\tau)$ - twelve unknowns) and can be solved using a modified Newton-Raphson type algorithm (see Kalidindi et al.[7] for the details of the algorithm). Once $\mathbf{T}^*(\tau)$

and $s^{\alpha}(\tau)$ are known, $\mathbf{F}^{p}(\tau)$ is obtained using Equation (22a) and $\mathbf{F}^{e}(\tau)$ is obtained from Equation (1). The new texture at time τ can be found by the polar decomposition of $\mathbf{F}^{e}(\tau)$ into its elastic lattice rotation component ($\mathbf{R}^{e}(\tau)$) and elastic stretching component and by using the Equations (19) and (20) to obtain:

$$\mathbf{s}^{\mathfrak{A}} = \mathbf{R}^{e}(\tau)\, \mathbf{s}^{\mathfrak{B}} \tag{31}$$

$$\mathbf{n}^{\mathfrak{A}} = \mathbf{R}^{e\text{-}T}(\tau)\, \mathbf{n}^{\mathfrak{B}}. \tag{32}$$

where, $\mathbf{s}^{\mathfrak{A}}$ (slip direction) and $\mathbf{n}^{\mathfrak{A}}$ (slip plane) are unit vectors representing the slip system α at time τ.

A computer program that solves this nonlinear problem is given in the Appendix.

EXPERIMENTAL STRESS-STRAIN BEHAVIOR

Material

Test specimens were machined from oxygen-free-high-conductivity (OFHC) copper (99.99% Copper). OFHC copper was received as bar stocks in the work hardened condition. Before machining, the copper rods were annealed at 400°C in argon for one hour. This annealing treatment produced an average grain diameter of 45 μm.

Experimental Procedure

Compression Tests

All the compression tests were conducted with cylindrical specimens 0.5 in (12.7 mm) in diameter with a nominal gage length of 0.5 in (12.7 mm) using an Instron screw driven testing machine. The specimen's ends were grooved as shown in the Figure 1 to retain the lubricant for a longer test time. To record the initial 5% strain, all the specimens were strain gaged. The ends of the specimens were lubricated with thin teflon tapes. The relative displacement of the ends of the specimens were monitored using a jig with a DCDT transducer. All the tests were conducted at a nominal engineering strain rate of 5.7×10^{-4} s^{-1}. The stress strain curve for engineering strains greater than 5% was obtained from the DCDT information and using the assumption that the deformation is incompressible. The discontinuity observed at the 5% strain level between the two curves was reconciled by moving the starting point of the stress-strain curve for strains greater than 5% to coincide with the end point of the stress-strain curve for strains less than 5%. This discrepancy arises due to the presence of the grooves at the loading interfaces of the specimen.

Compression Test Results

Stress-strain results for the compression tests are given in Figure 2. These data were obtained using five different specimens. The maximum strain that was obtained was 1.30.

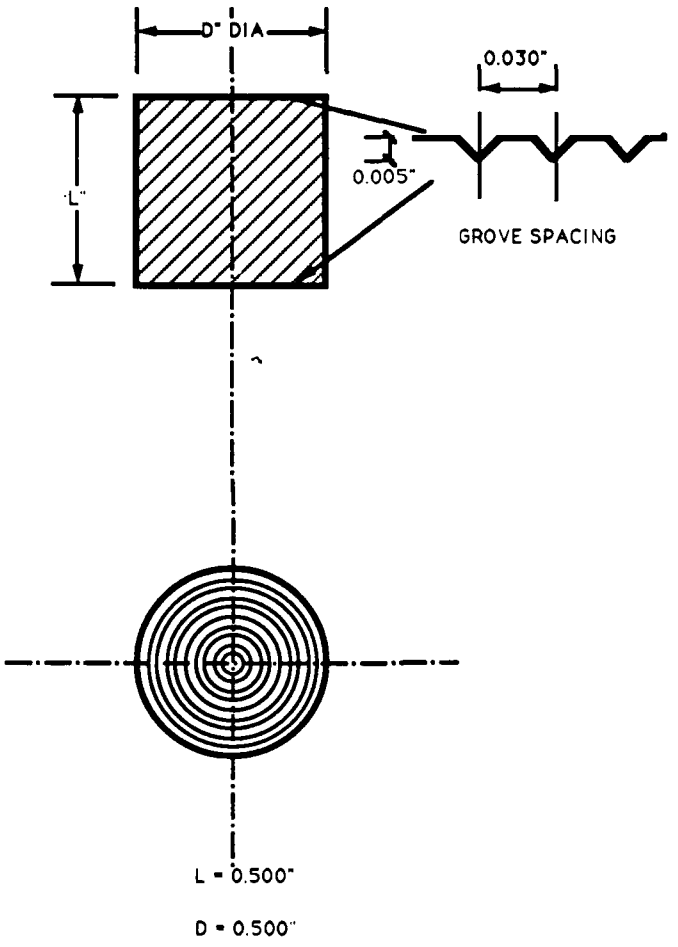


Figure 1. Compression Specimen

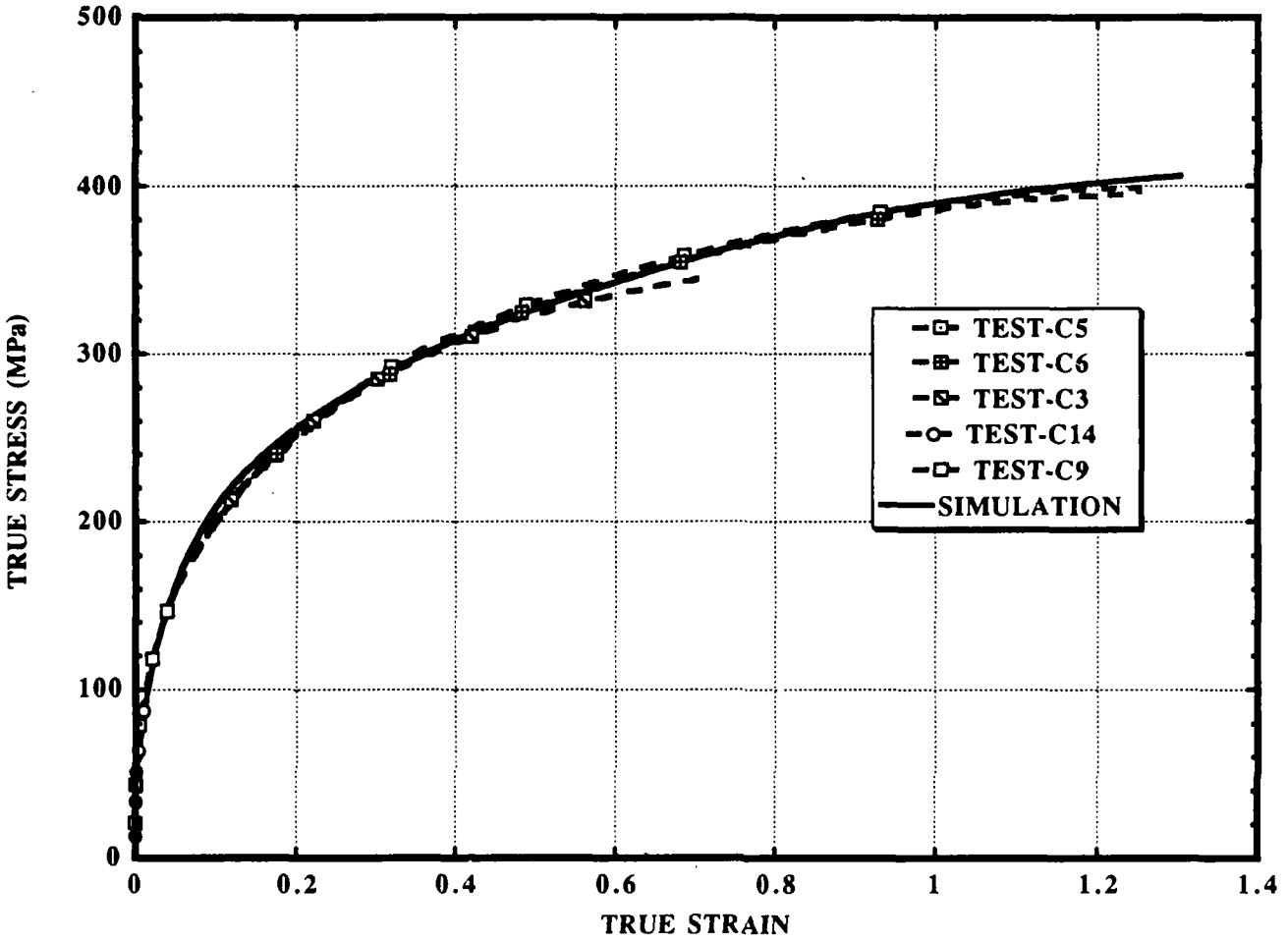


Figure 2. Experimental Uniaxial Compression Stress-Strain Data. Simulated Compression Stress-Strain Curve (for the Single Slip Behavior: s_0 = 16 MPa; h_0 = 700 MPa; s_s = 155 MPa; a = 3.8) is also given.

Torsion Tests

Specimen geometry of the torsion test specimen is given in Figure 3. This specimen is a modified Lindholm-type specimen that was used by White et al.[12] The gage portion of the torsion specimen is a thin wall tube of 0.240 in (6.096mm) length and external and internal diameters 0.810 in (20.574mm) and 0.750 in (19.050mm) respectively (0.030 in wall thickness). The torsion tests were conducted using both free and fixed-end conditions at a shear strain rate of 0.001 s^{-1}. For the tests that were conducted under fixed-end conditions, the axial strain was assumed to be zero. Though the axial strain in the gage sections were not directly measured, axial extension just outside the gage section was monitored for one free-end and one fixed-end test using a specially designed axial extensometer (see Figure 3). All the torsion tests were conducted using a tension-torsion hydraulic test machine. Specimens were attached to the test machine using a pair of hydraulic collet grips.

Shear stress was derived using the expression

$$\text{Shear stress} = \text{Torque} / (2\pi r_m^2 t) \tag{33}$$

where r_m is the mean radius of the gage section and t is the wall thickness of the gage area. The axial stress was obtained by

$$\text{Axial stress} = \text{Axial Force} / (\pi(R^2 - r^2)) \tag{34}$$

where R is the outer radius and r is the inner radius. Axial strain is given by the expression

$$\text{Axial Strain} = \text{Axial Extension/Gage Length} \tag{35}$$

and the shear strain is given by the expression

$$\text{Shear Strain} = (R\theta)/\text{Gage Length} \tag{36}$$

where θ (in radians) is the relative rotation at the grips.

Torsion Test Results

Stress-engineering shear strain curves obtained for fixed-end torsion tests are given in Figure 4. The axial stress is initially compressive and goes through a minimum at an approximate shear strain of 1 with the monotonic increase of shear strain; as the shear strain is increased further, the axial stress becomes tensile at an approximate strain of 2.25. The axial strain that was obtained by monitoring the displacement just outside the end of the shoulder section is also given in the Figure 4. The axial strain monitored this way was

12. WHITE, C. S., BRONKHORST, C. A. AND ANAND, L. *An Improved Isotropic-Kinematic Hardening Model for Moderate Deformation Metal Plasticity*. Mech. Mat., v. 10, 1990, p. 127-147.

approximately zero up to a shear strain of 1.75 and increases after this strain (positive strain indicates an extension in the gage section).

Shear stress-shear strain behavior obtained using free-end tests is given in Figure 5. Also in this figure, the axial strain behavior is given as a function of the shear strain. Axial strains were obtained both from the machine and from the extensometer. As expected the axial strain measured from the extensometer was lower than measured by the machine. Here, positive strain indicates an extension. The axial strain increases and remains approximately constant after a shear strain of about 1.

Both fixed-end and free-end stress-strain behavior is given in Figure 6. The shear stress-shear strain behavior is approximately equal and therefore insensitive to the method of testing - i.e., whether the type of testing is fixed-end or free-end.

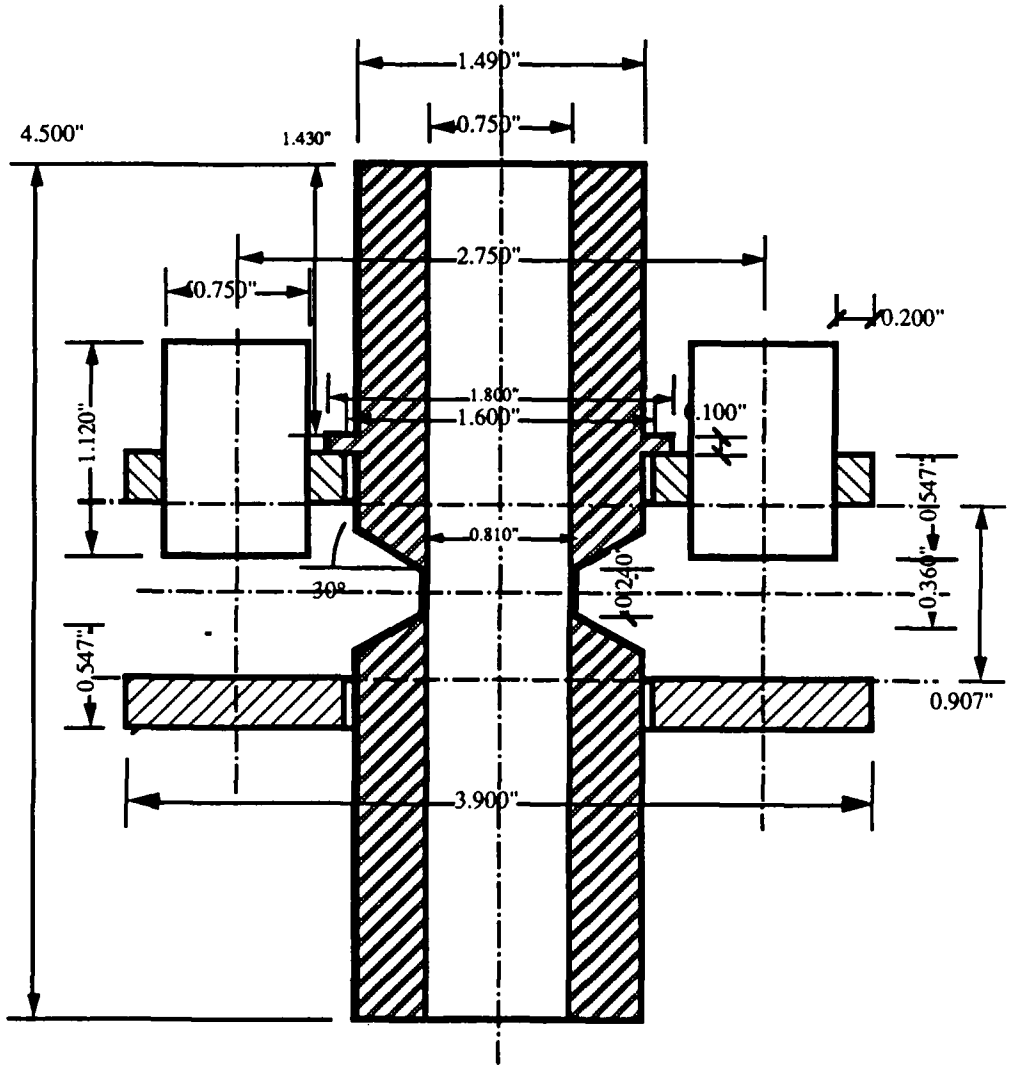


Figure 3. Torsion Specimen with the Extensometer

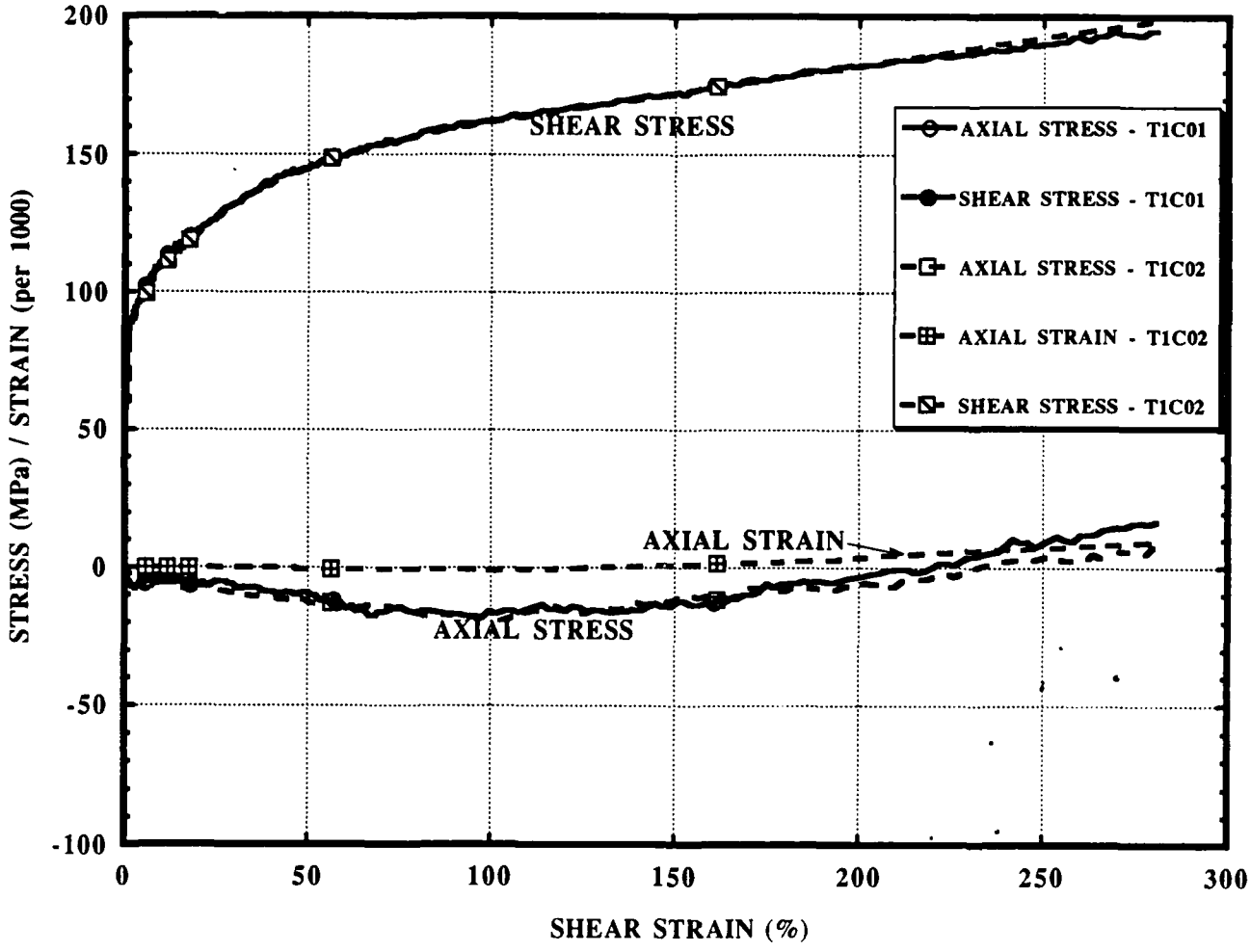


Figure 4. Shear Stress/Axial Strain-Shear Strain from Fixed-End Torsion Tests

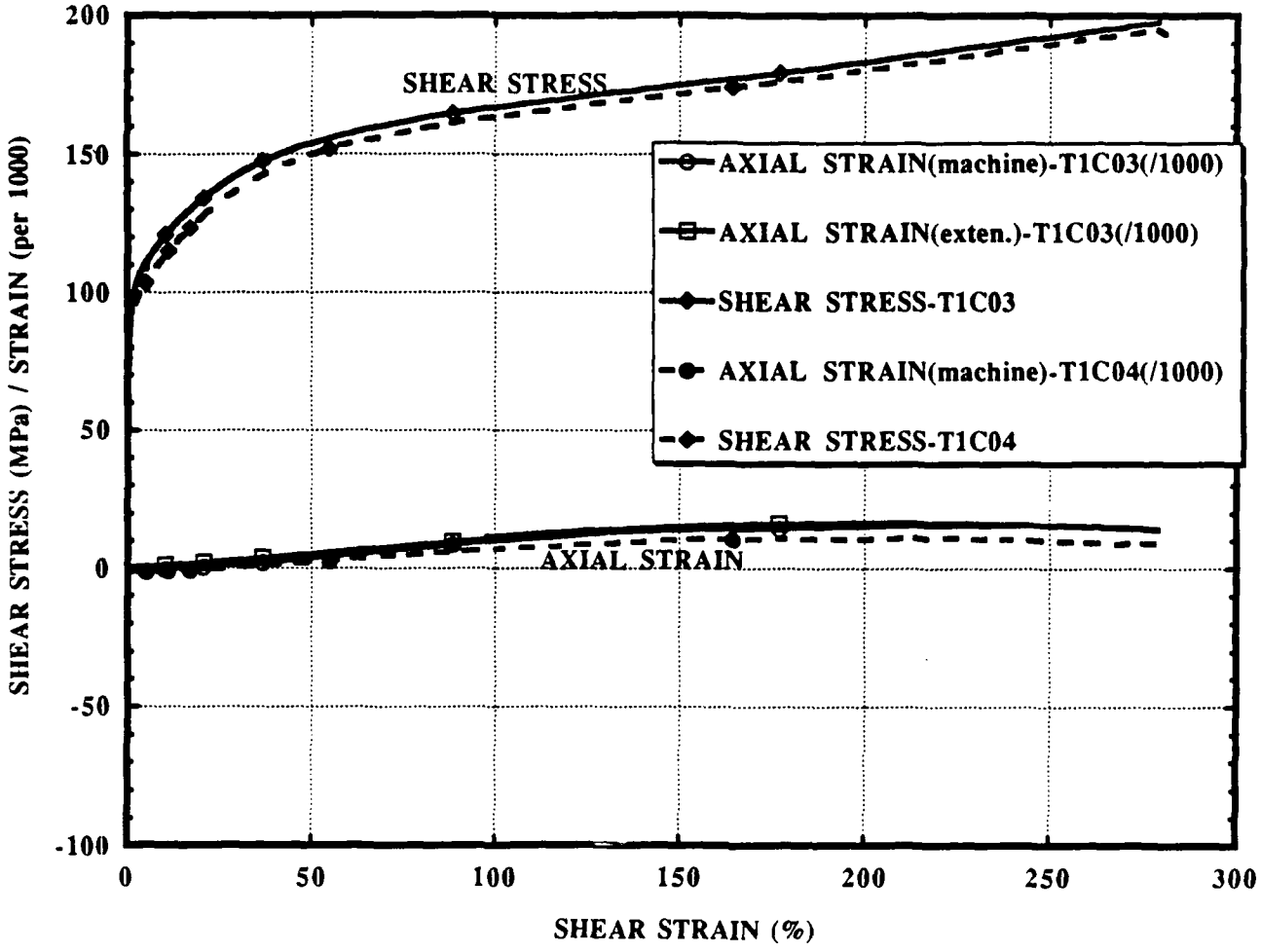


Figure 5. Shear Stress/Axial Strain versus Shear Strain from Free-End Torsion Tests

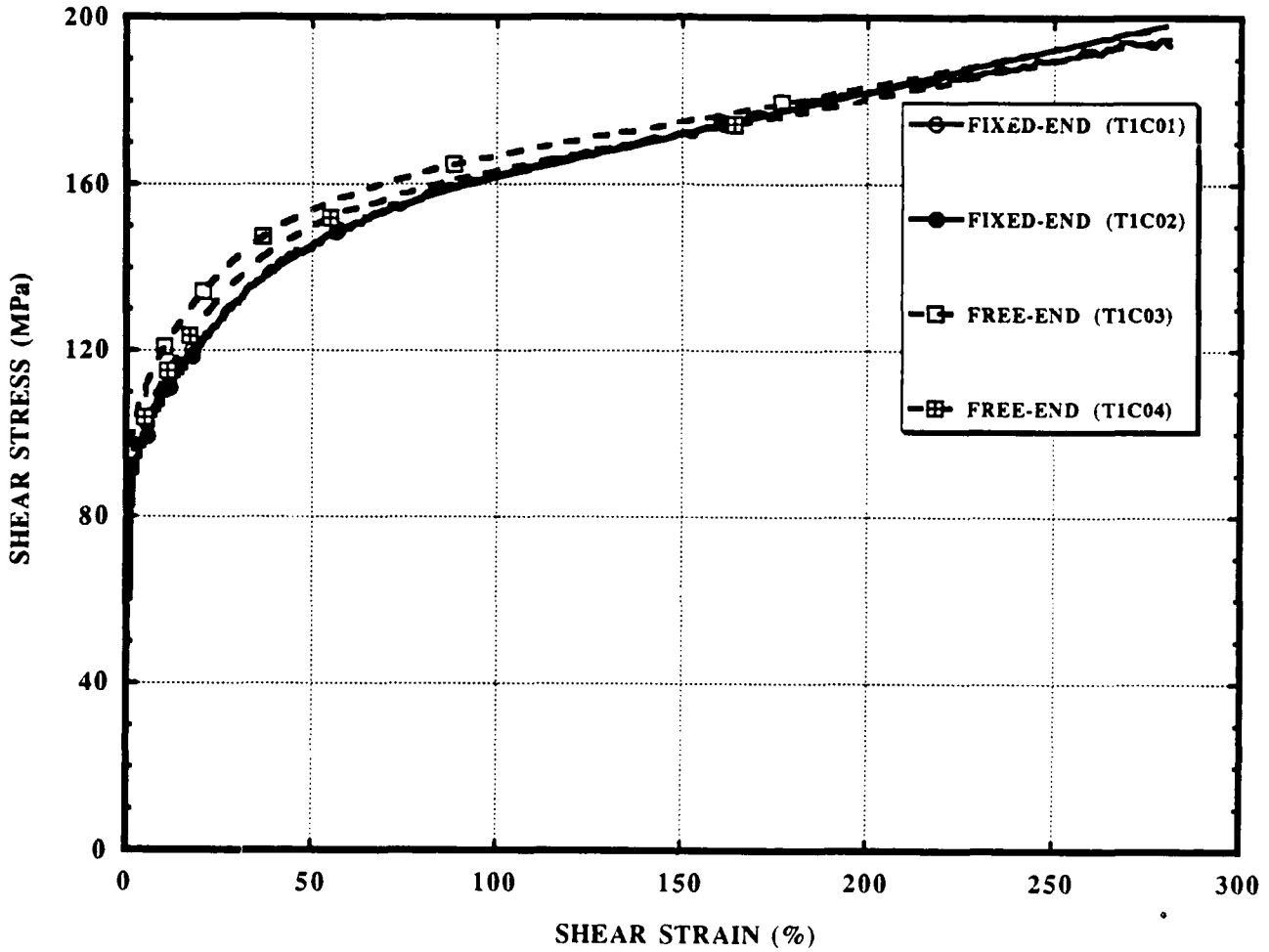


Figure 6. Comparison of Experimental Fixed-End and Free-End Torsion Data

MODEL PREDICTIONS

The computer program given in the Appendix was used to obtain the polycrystal model predictions of the uniaxial compression and fixed-end torsion (simple shear) tests and the evolution of crystallographic texture. Following Kalidindi et al., the strain rate sensitivity parameter, m, for OFHC copper was taken as 0.012. Latent hardening, q, for the copper was assumed to be 1.4 for the computation. Also simulations were conducted with m = 0.04; q = 1.4, and m = 0.012; q = 1.0 to evaluate the effect of higher strain rate sensitivity and lower latent hardening.

Computations were conducted using a 16 MHz IBM compatible 80386 computer with 80387 floating point coprocessor support or a Mac IIcx computer. The initial crystallographic texture was assumed to be isotropic (the grains were randomly oriented); the initial distribution of the crystals were taken from Molinari et al.[13] and was given as 300 sets of Euler angles. Each set of Euler angles would rotate the axes of the corresponding crystal to coincide with the global fixed rectangular Cartesian axes. Schmid tensor, S^{g}, was computed with respect to the global coordinate system for each crystal by

$$S^{g} = Q\, S^{g}\, Q^{T} \tag{37}$$

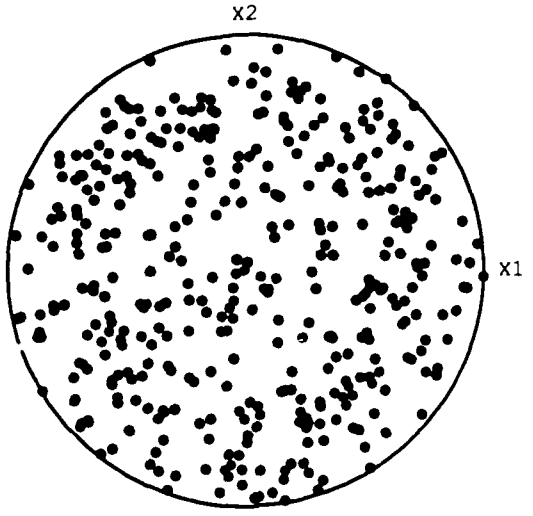
where S^{g} is defined with respect to an orthonormal basis associated with the crystal lattice <100> directions. Twelve slip systems which defines S^{g} for fcc crystal are given in the Table 1.

The orthogonal matrix Q which rotates the crystal basis to coincide with the global fixed basis is given by

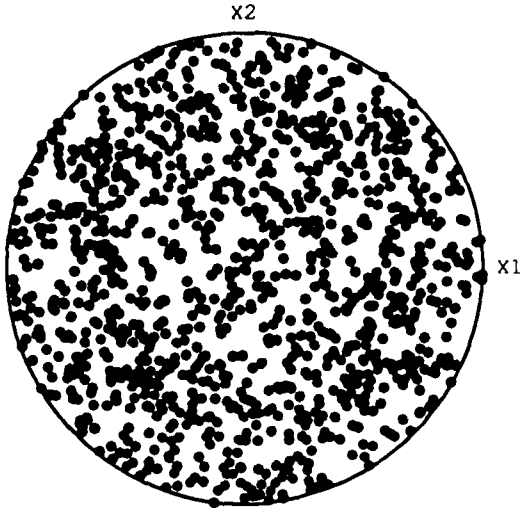
$$Q = \begin{bmatrix} \cos\phi \sin\theta & \sin\phi \cos\omega & \\ -\sin\phi \sin\omega \cos\theta & +\cos\phi \sin\omega \cos\theta & \sin\omega \sin\theta \\ -\cos\phi \sin\omega & -\sin\phi \sin\omega & \\ -\sin\phi \sin\omega \cos\theta & \cos\phi \cos\omega \cos\theta & \cos\omega \sin\theta \\ \sin\phi \sin\theta & -\cos\phi \sin\theta & \cos\theta \end{bmatrix} . \tag{38}$$

where $\{0 \le \phi \le 2\pi, 0 \le \theta \le \pi, 0 \le \omega \le 2\pi\}$ are the three Euler angles that represent this transformation. Most of the simulations were conducted with a subset of 100 crystals after it was determined that there was no appreciable difference in the results whether 300 or 100 crystals were used (see sub-section Fixed-end Torsion Behavior). Both of these crystal distributions are given in Figure 7 as equal area projection {111} pole figures.

13. MOLINARI, A., CANOVA, G. R. AND AHZI, S. *A Seif-Consistent Approach of the Large Deformation Polycrystal Viscoplasticity.* Acta Metall., v. 35, 1987, p. 2983-2994.



(100 crystals)



(300 crystals)

Figure 7. {111} Equal-Area Projection Pole Figures of Initial Randomly Oriented a). 100 and, b). 300 Crystals

Table 1. Twelve slip systems for fcc crystal

slip system α	slip plane $\mathbf{n}^\alpha$	slip direction $\mathbf{s}^\alpha$
1	(1 1 1)/√3	[-1 1 0]/√2
2	(1 1 1)/√3	[-1 0 1]/√2
3	(1 1 1)/√3	[0-1 1]/√2
4	(-1 1 1)/√3	[1 1 0]/√2
5	(-1 1 1)/√3	[1 0 1]/√2
6	(-1 1 1)/√3	[0-1 1]/√2
7	(1-1 1)/√3	[1 1 0]/√2
8	(1-1 1)/√3	[0 1 1]/√2
9	(1-1 1)/√3	[1 0-1]/√2
10	(1 1-1)/√3	[0 1 1]/√2
11	(1 1-1)/√3	[1 0 1]/√2
12	(1 1-1)/√3	[1-1 0]/√2

Uniaxial Compression Behavior

The time integration algorithm discussed earlier requires that the deformation gradient to be prescribed for the deforming body. In a uniaxial compression test, axial strain rate is known; but the lateral strain rates are not known. Following Bronkhorst et al.,[14] for the uniaxial compression test, the material elements are assumed to follow an isochoric motion given by

$$\mathbf{x} = \exp\{\mathring{\varepsilon}t/2\}X_1\,\mathbf{e}_1 + \exp\{\mathring{\varepsilon}t/2\}X_2\,\mathbf{e}_2 + \exp\{-\mathring{\varepsilon}t\}X_3\,\mathbf{e}_3 \qquad (39)$$

where $\mathring{\varepsilon}$ is the strain rate, $\mathbf{X}$ $\{X_1,X_2,X_3\}$ and $\mathbf{x}$ are the original and current positions of a material element respectively and $\mathbf{e}_i$ $\{i = 1, 2, 3\}$ are orthonormal base vectors of a global rectangular Cartesian coordinate system. The response from the Taylor model for this motion is the volume averaged stress deviator $\mathbf{T}' = \mathbf{T} + p\mathbf{I}$, where p an undetermined pressure. In a uniaxial compression test, because the lateral tractions are zero, the approximate value of the axial Cauchy stress can be represented by

$$T_{33} = T'_{33} - p, \qquad \text{where } p = (T'_{11} + T'_{22})/2 \qquad (40)$$

14. BRONKHORST, C. A., KALIDINDI S. R. AND ANAND, L. *Polycrystalline Plasticity and the Evolution of Crystallographic Texture in FCC Metals.* to be published, 1991.

The assumed single slip hardening modulus and resistance of the constitutive behavior of the single crystal is given by the Equations (13) and (14). Here, the unknown parameters are h_o - the initial hardness, s_s - saturated value of the slip resistance, s_o - the initial slip resistance or the initial critical resolved shear stress (crss) and a - the exponent. By systematically iterating these parameters, the calculated macroscopic stress-strain behavior can be adjusted to represent the experimentally observed stress-strain behavior. The saturated stress-strain behavior is dependent on the s_s value, initial slope of the stress-strain curve is dependent on the h_o value, yield point is dependent on the s_o value, and the exponent, a, determines the shape of the stress-strain curve between the yield point and the saturation point. By systematically changing the values of these parameters, a simulated compression stress-strain curve was obtained, which approximately represents the experimental stress-strain behavior. This simulated stress-strain behavior is given in Figure 2 with the experimental stress-strain curves. Corresponding parameters of the single slip behavior are

$$s_o = 16 \text{ MPa}, h_o = 700 \text{ MPa}, s_s = 155 \text{ MPa and } a = 3.8. \quad (41)$$

These parameters for single slip behavior are used for the simulation of the torsion (simple shear) stress-strain behavior in the next section. Thus, by curve fitting the Taylor model to the experimental results of the compression tests, the parameters for simulating the torsion tests are fixed.

For this single slip behavior, simulated stress-strain behaviors for m = 0.04, q = 1.4 and m = 0.012, q = 1.0 are given in the Figure 8 for comparisons. As the strain rate sensitivity increases or latent hardening decreases, for uniaxial compression stress-strain loading, the simulation predicts a lower stress for the same strain.

Fixed-end Torsion (Simple Shear) Behavior

Using the parameters that were determined from the simulation of the compression tests, fixed-end torsion tests were simulated. In the fixed-end torsion tests, the material elements are assumed to follow a motion described by

$$\mathbf{x} = (X_1 + \dot{\gamma} t X_2)\mathbf{e}_1 + X_2 \mathbf{e}_2 + X_3 \mathbf{e}_3. \quad (42)$$

where $\dot{\gamma}$ is the engineering shear strain rate. The simulated shear (T_{12}), axial (T_{22}) and hoop (T_{11}) stresses as functions of shear strain are given in Figure 9 for both 100 and 300 crystals. There is no appreciable difference in the stress-shear strain behavior for 100 or 300 crystals except the axial and hoop stress behavior become smoother for 300 crystals. Therefore, all the other simulations that are reported in this report were carried out with 100 crystals.

Figure 10 gives the simulated stress-shear strain behavior (m = 0.012 and q = 1.4) for simple shear loading with the corresponding fixed-end experimental data from Figure 4. Simulated axial stresses are compressive in the range of shear strains up to 3 and go through a compressive maximum stress as the shear strain is increased. In contrast, the hoop stress is tensile and increases as the shear strain is increased. For comparison with this simulated stress-strain behavior, Figure 11 gives the calculated stress-shear strain behavior for a higher rate sensitive material (m = 0.04; q = 1.4) and for a lower latent

hardening (q = 1.0; m = 0.012) material. In both of these simulations, single slip hardening was assumed to follow the behavior given by the parameters in (41).

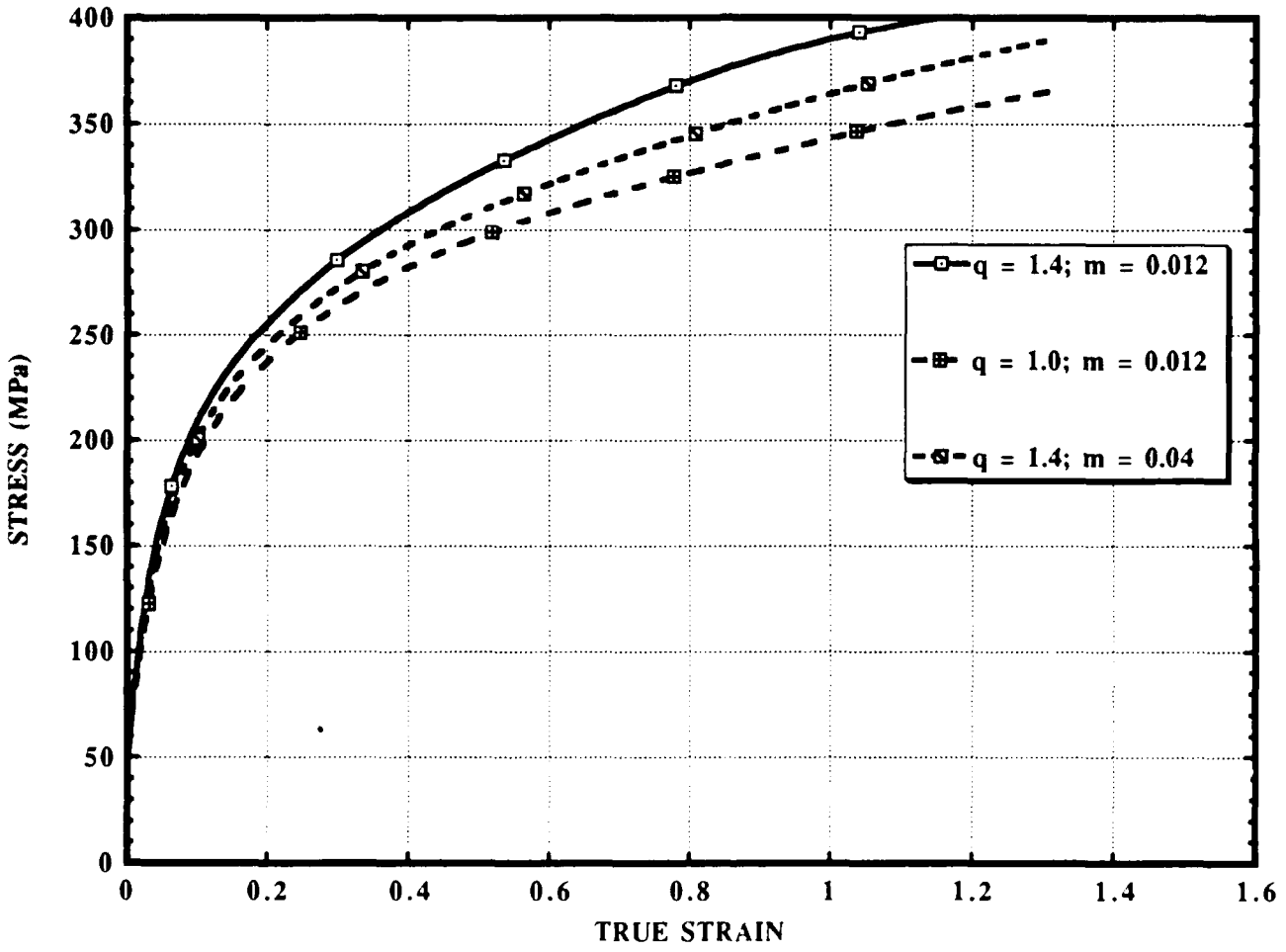


Figure 8. Effect of Strain Rate Sensitivity and Latent Hardening on Simulated Compression Stress-Strain (s_o = 16 MPa; h_o = 700 MPa; s_s = 155 MPa; a = 3.8)

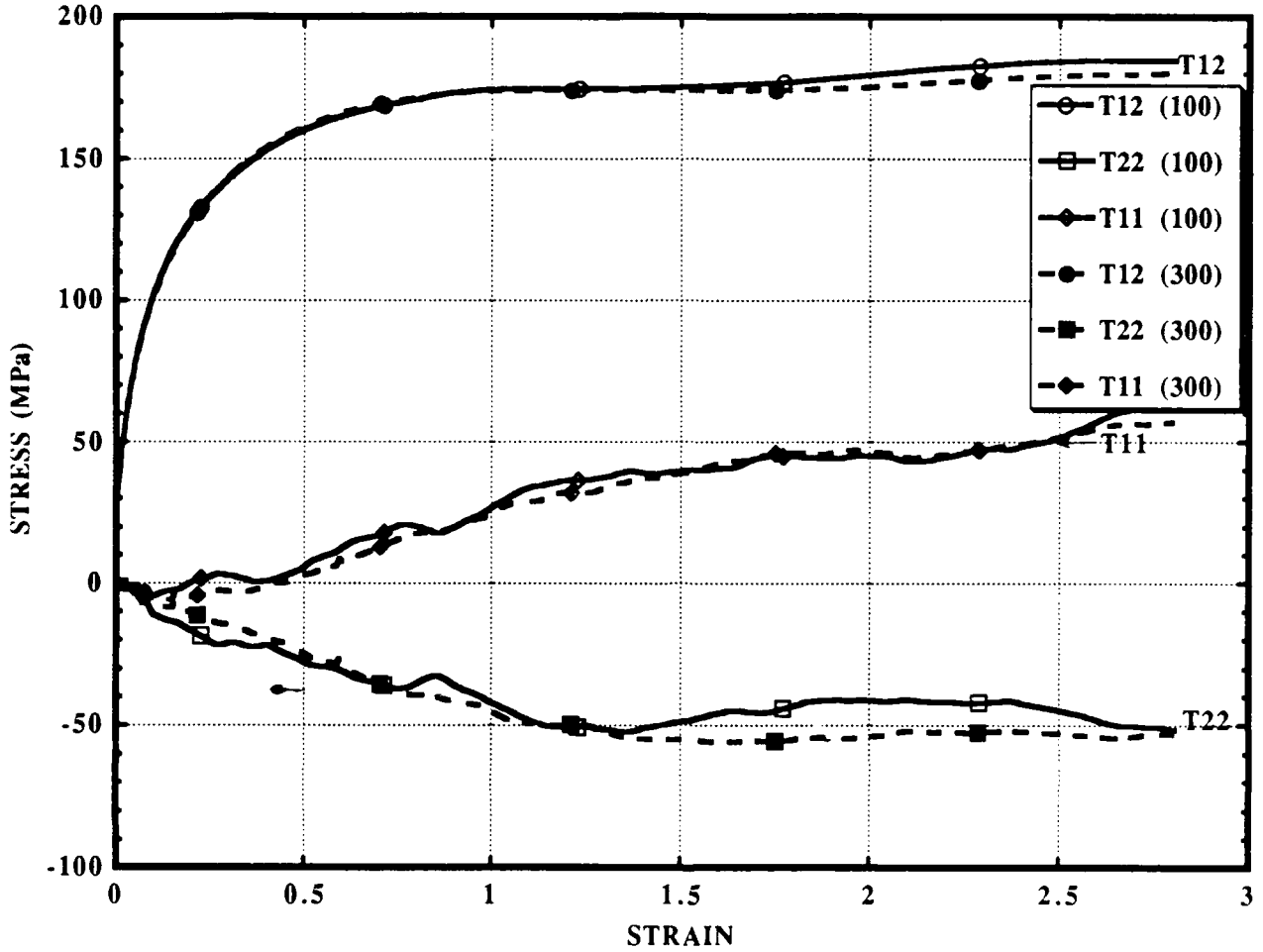


Figure 9. Comparison of Simulations of Simple Shear with 100 versus 300 Crystals

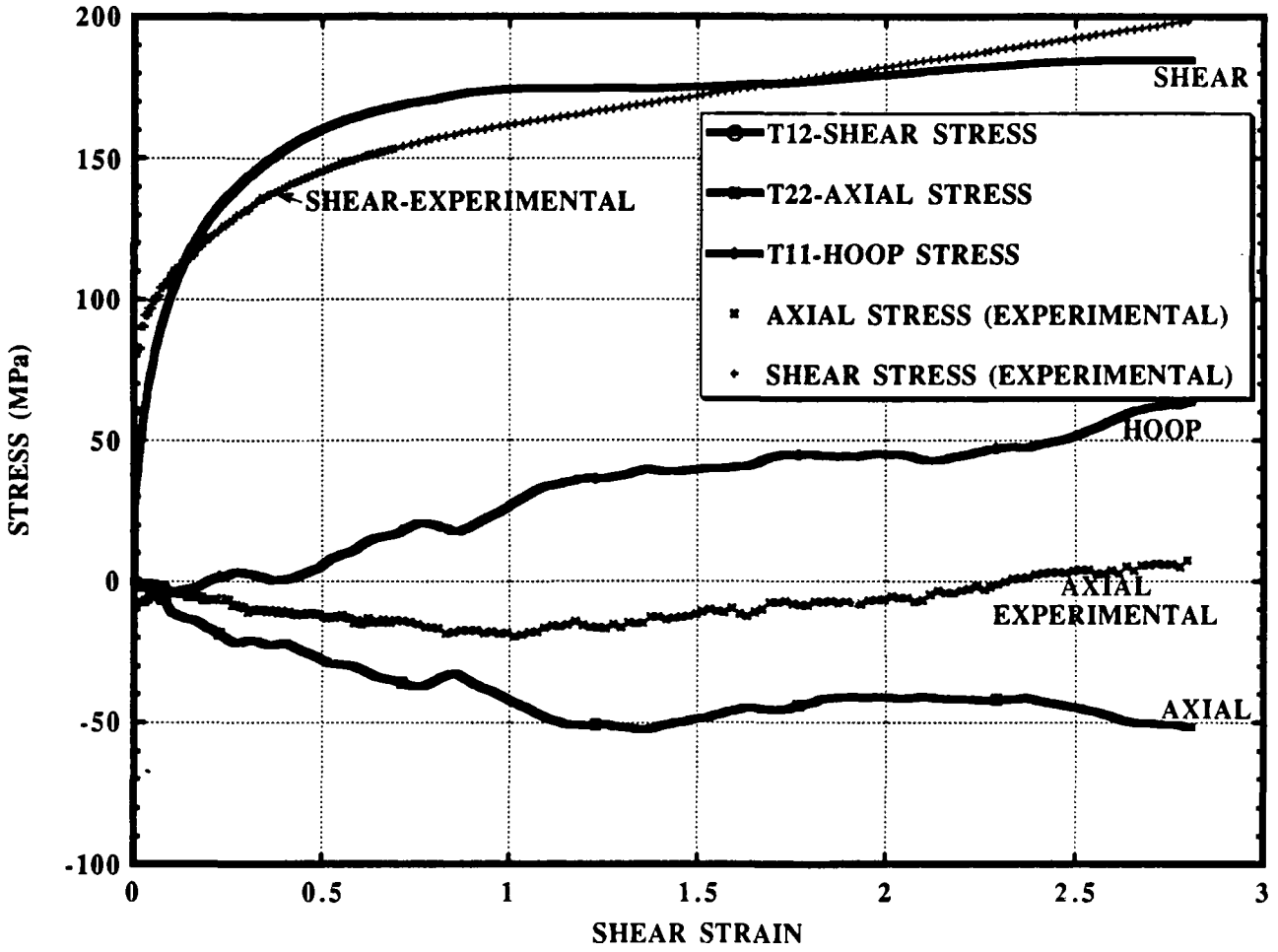


Figure 10. Comparison of Fixed-End Torsion Experimental and Simulated Stress-Strain

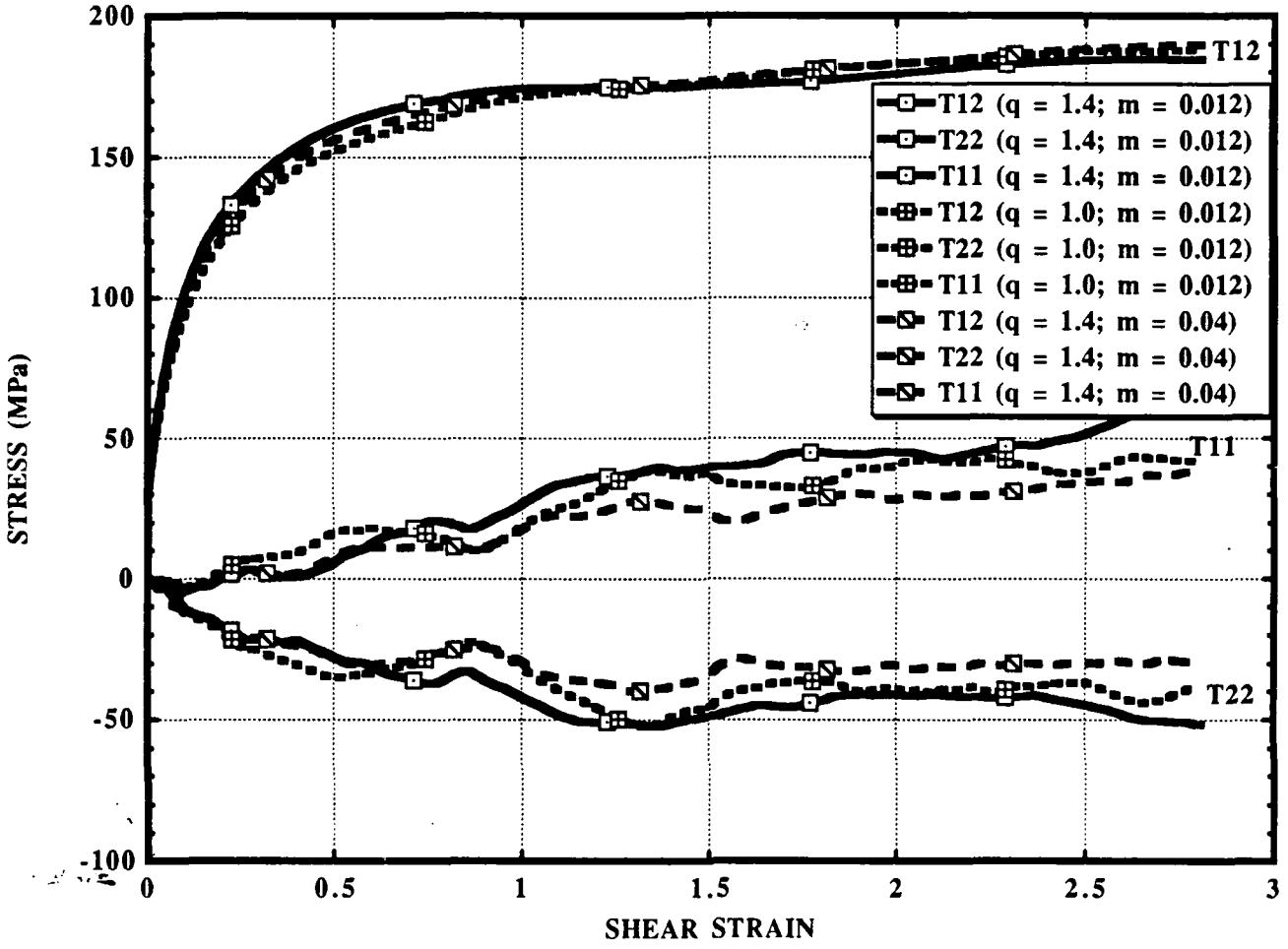


Figure 11. Simulated Simple Shear Stress-Strain as a function of Strain Rate Sensitivity (m= 0.012 and 0.04) and Latent Hardening (q = 1.4 and 1.0) (single slip behavior from experimental compression data with m=0.012 and q= 1.4)

A simulation was also conducted starting from compression data for these two cases: 1) m = 0.04; q = 1.4 - a higher rate sensitive material, for which single slip behavior was found to be given by the Equation 41, except for s_s which was 166 MPa to fit to the experimental data, 2) q = 1.0; m = 0.012 - a lower latent hardening material, for which single slip behavior was found to be given by the Equation 41, except for s_s which was 170 MPa to fit to the experimental data. The corresponding simple shear behavior for these two cases are given in Figure 12 with the behavior from single slip define by the Equation 41 and experimental data.

COMPARISON OF EXPERIMENTAL RESULTS WITH MODEL PREDICTIONS

The suitable selection of the parameters of the single slip behavior, with m = 0.012 and q =1.4, made it possible to obtain a simulated uniaxial compression stress-strain curve that closely follows the experimental results as given in Figure 2. The parameters chosen to fit the compression data remain fixed for the torsion simulations. The predicted stress-strain curve is dependent upon the rate sensitivity as well as the amount of latent hardening: the higher the rate sensitivity, the lower the stress at a give strain value and, the lower the amount of latent hardening, the lower the stress for a given strain value.

The stress-shear strain curve from the simulation of simple shear is given in Figure 10 with the fixed-end torsion stress-strain data. In both simulation and experiment, the axial stresses are compressive, thus qualitatively both have the same results; but the simulation over predicts the measured axial stress. Also, the experimental axial stress becomes tensile at higher shear strains. The simulation does not show this trend up to the maximum simulated shear strain of 2.8. The experimental shear stresses are higher than the simulated values at low and high strains, but the simulated stress is higher than the experimental values for intermediate strain levels. This observation is different from that observed by Bronkhorst et al.[14] They observed a better correlation between results from experiments and simulation. Also we should note that the two materials are different since the annealing temperature was different in both studies. An annealing temperature of 850°C was used in their studies.

The other difference between the two studies is the annealing point in the study: i.e., whether the annealing was performed before or after machining of the specimen. In Bronkhorst study, specimens have been annealed after the machining, but in this study the specimens were machined after annealing so as not to distort the thin wall of the specimen. With the assumption that the specimen may have been work hardened during machining, the following two approaches were taken: 1) a simulation was conducted by increasing the initial single slip resistance, s_0, so that the initial portion of the predicted shear stress-shear strain curve matches the experimental results, and 2) one specimen was annealed before testing. Figure 13 gives a comparison of the simulated results from the first case with the experiment, where the given simulation was conducted with the initial single slip resistance, s_0 = 45 MPa. The discrepancy between the experiment and the simulation at the lower strains disappeared with this simulation, but the differences at the intermediate and large strains remain unchanged. In the second case, where the specimen was annealed

before testing, the experimental results are given with the simulated results in Figure 14. In this case also, the differences in shear stress at lower strains disappeared; however, the differences between the experiment and the simulation increased in the intermediate strain range. At the range of large strains, the experimental shear stress seems to asymptotically approach the simulated results. The new simulated and experimental axial responses did not appreciably deviate from the original simulated and experimental axial responses, respectively.

Figure 11 shows a comparison of the simulated shear and axial stress-shear strain responses for an increase of rate sensitivity (m = 0.012 -> 0.04) and for a decrease of latent hardening (q = 1.4 -> 1.0) parameters. The shear stress response did not change appreciably, but there is a change in the axial and hoop stress responses. As the strain rate sensitivity was increased, the axial and hoop stresses decreased for a given strain. For strains less than 0.6, the axial and hoop stresses are higher with reduced latent hardening, but are lower for reduced latent hardening for strains greater than 0.6.

In contrast to the simulations given in Figure 11, Figure 12 gives the simulated stress-strain response for both higher rate sensitive (m = 0.04) and lower latent hardening (q = 1.0) materials by obtaining the single crystal slip parameters for both cases from the compression data as described in the previous section. These simulations further deviate from the experimental results, therefore indicating that the best simulation is obtained for m = 0.012 (the experimental m from polycrystal jump test by Bronkhorst et al.) and q = 1.4 (the value suggested by Kocks et.al.[15] and Asaro et al.[7] after reviewing a great number of single crystal experiments).

The discrepancies between predicted and experimental stress-strain responses may be due to the followings reasons:

1). The Taylor model does not take into account the interaction between grains. Also the stress equilibrium stress between the grains is violated.

2). The fixed-end torsion tests do not represent simple shear perfectly; this could explain the reduction in the magnitude of the experimental axial stress. Lipkin et al.[16] have experimentally observed the presence of axial strains in the gage section, thus reducing the magnitude of the axial stress due to partial relaxing of the axial constraints. Also finite element calculations by Lipkin et al.[16] and White[17] show the presence of axial strains in the gage section. These finite element calculations also show that the actual shear strain in the gage section is lower than that computed from the relative rotation measured at the testing machine (grips). In addition, the finite element calculations show that even for thin wall specimens, stresses are not uniform through the thickness or close to the shoulder section of the gage area. Also, we should note that these finite element calculations have been conducted with assumed constitutive models.

15. KOCKS, U. F. *The Relation Between Polycrystal Deformation and Single-Crystal Deformation.* Met. Trans., v. 1, 1970, p. 1121-1143.
16. LIPKIN, J., CHIESA, M. L. AND BAMMANN, D. J. *Thermal Softening of 304L Stainless Steel: Experimental Results and Numerical Simulations.* Proc. of IMPACT'87, Bremen, FRG, 1987.
17. WHITE, C. S. *An Analysis of the Thin-Walled Torsion Specimen.* submitted to the ASME J. Eng. Mat. Tech., 1991.

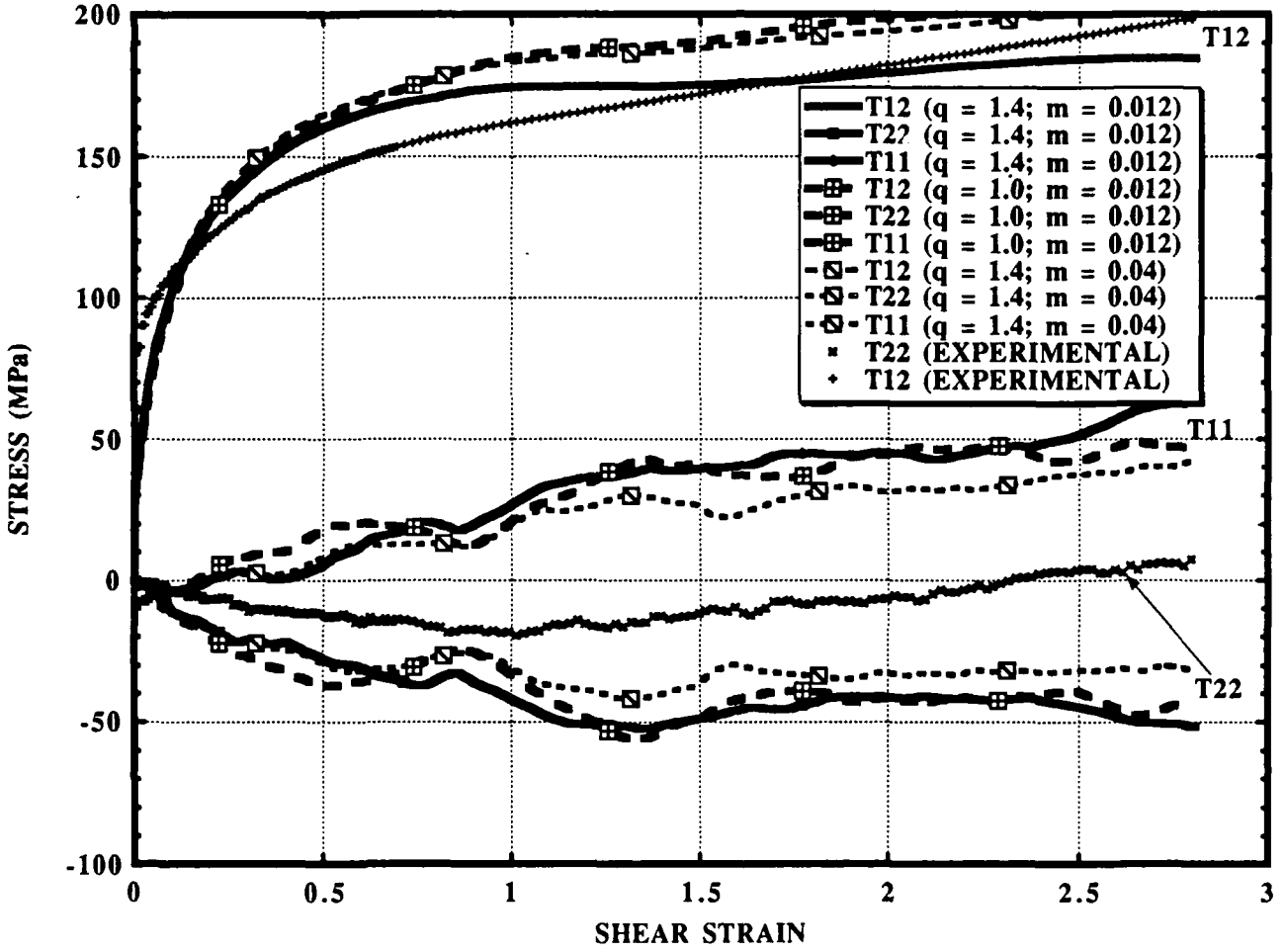


Figure 12. Simulated Simple Shear Stress-Strain as a function of Strain Rate Sensitivity (m = 0.012 and 0.04) and Latent Hardening (q = 1.4 and 1.0) (each single slip behavior was deduced from compression experimental data)

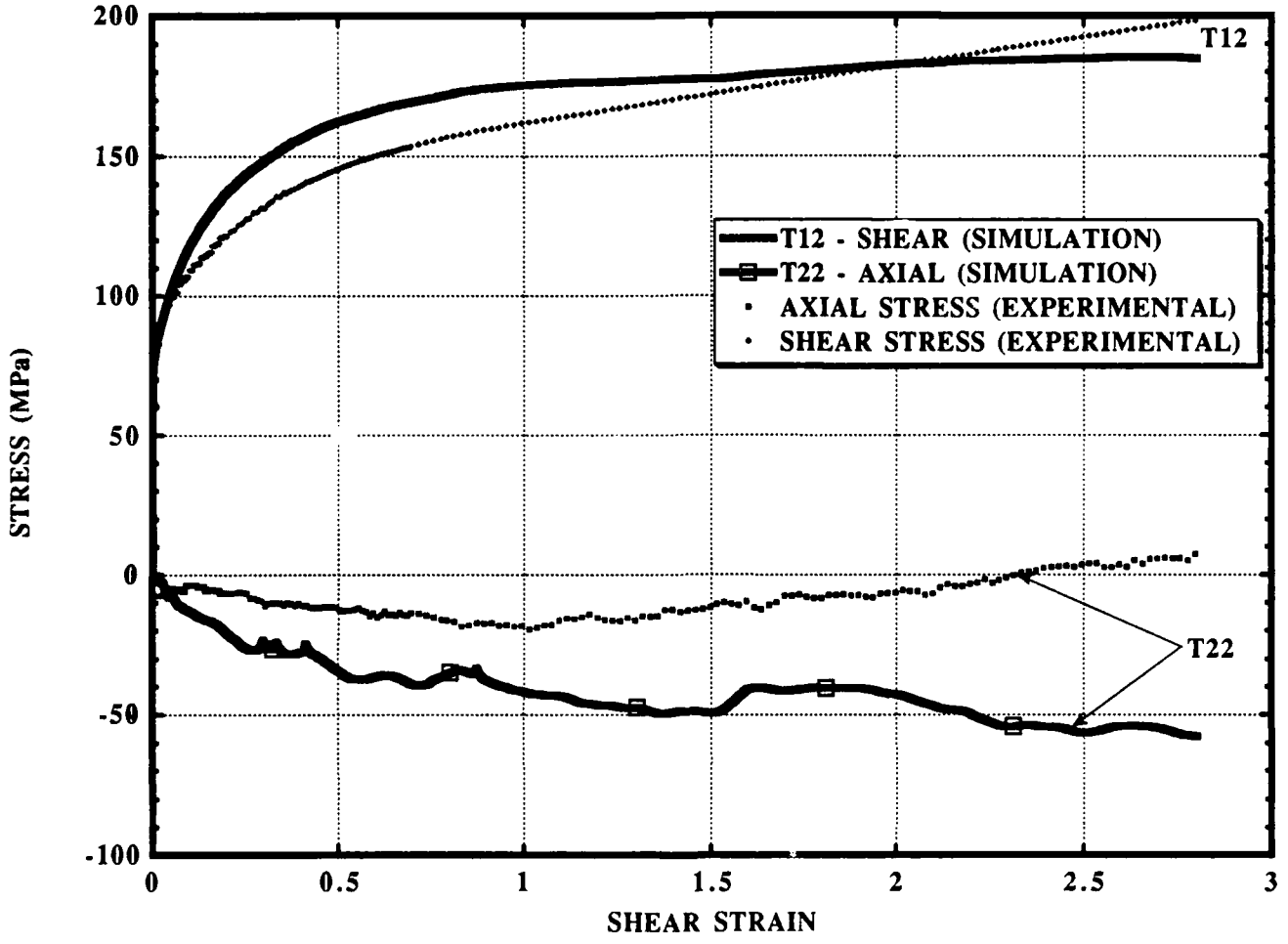


Figure 13. Simulated Stress-Shear Strain with s_0 = 45 MPa versus Experimental Fixed-end Torsion Stress-Shear Strain Data

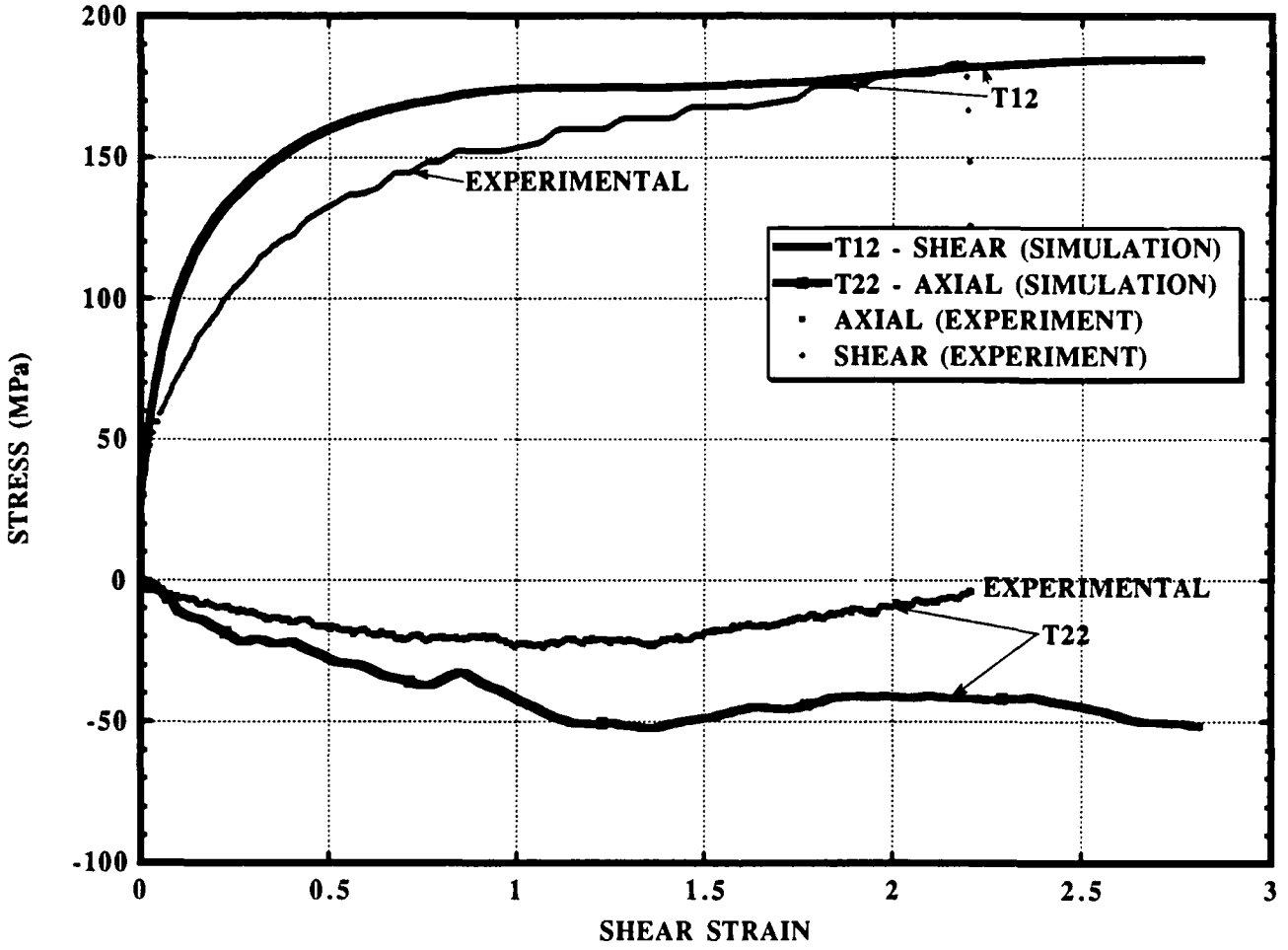


Figure 14. Simulated Stress-Shear Strain versus Experimental Fixed-end Torsion Stress-Shear Strain Data (specimen was heat treated after machining)

PREDICTED TEXTURES

As described in the section entitled The Taylor-type Polycrystal Model, the evolution of the texture can be obtained from the polycrystal simulation. The initially assumed random distribution of the crystals are given in Figure 7 as equal-area projection {111} pole figures. Predicted pole figures for both uniaxial compression and simple shear experiments are discussed and compared with the corresponding ideal textures in Figures 15 and 16 in the following sections. Experimental pole figures are being obtained and will be discussed and compared with the predicted pole figures in a future report.

Uniaxial Compression

At a uniaxial compression strain of 1.3, equal-area projection {111}, {110} and {100} pole figures of the polycrystal are given in Figure 17. From these pole figures, it can be seen that the polycrystal has developed a texture which is symmetric with the loading axis. As seen by others (Bronkhorst et al.[14]) in uniaxial compression tests, the {110} planes have aligned perpendicular to the loading axis (see the {110} pole in the {110} pole figure). The predicted uniaxial compression pole figure is identical to the ideal {110}<*uvw*> fibre texture given in Figure 15, where {110} planes align perpendicular to the loading axis (e_3-direction).

Fixed-end Torsion (Simple Shear)

Figure 18 gives the predicted {111}, {110} and {100} pole figures for a simple shear test at a shear strain of 2.8. These pole figures are similar to the ones predicted by Harren et al.[8] using the time integration algorithm by Asaro and Needleman.

Ideal textures that are formed in shear deformation have been classified in detail by Canova et al.[18], Montheillet et al.[19] and Harren et al.[8] Following the notation by Canova et al., the ideal textures, {*hkl*}<*uvw*>, where {*hkl*} planes aligning with the global shear plane (e_2-direction) and <*uvw*> directions aligning with the global shear direction (e_1-direction), that form under shear can be divided into three categories: A fibre texture - {111}<*uvw*>, B fibre texture - {*hkl*}<110>, and C texture - {001}<110>. These ideal shear textures are given as {111} pole figures in Figure 18. It has been hypothesized by Harren et al. that the B fibre and the C orientation textures do not produce axial stresses, but the A fibre texture produces either compressive or tensile axial stresses depending on whether the A fibre is either A^- partial fibre or A^+ partial fibre (partial A fibres are identified in Figure 16a), respectively. Also in the Figure 16(b), B fibre is divided into two partials, B^y and B^n.

Comparing the predicted and ideal {111} pole figures, prediction shows the presence of a strong A^- fibre and the absence of A^+ fibre. Also the B^y fibre and the C orientation are strongly present in the pole figure; however, B^n fibre is absent in the prediction.

18. CANOVA, G. R., KOCKS, U. F. AND JONAS, J. J. *Theory of Torsion Texture Development.* Acta Metall., v. 32, 1984, p. 211-226.
19. MONTHEILLET, F, COHEN, M, AND JONAS, J. J. *Axial Stresses and Texture Development During the Torsion Testing of Al, Cu and α-Fe.* Acta Metall., v. 32, 1984, p. 2077-2089.

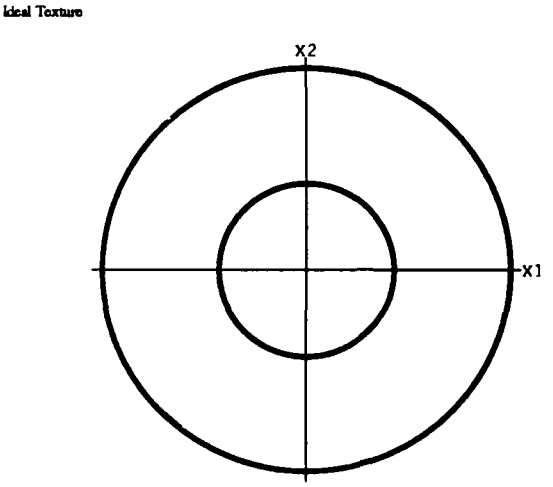
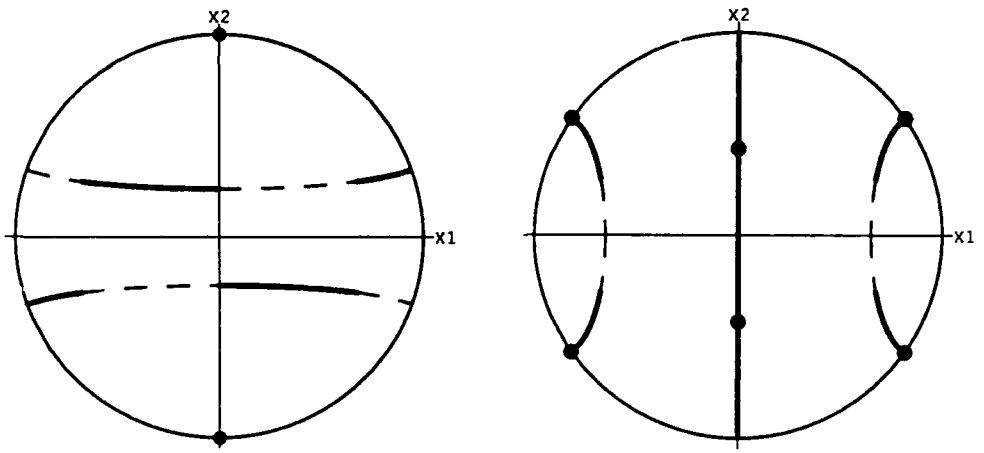


Figure 15. Ideal Texture {111} Pole Figure for Uniaxial Compression



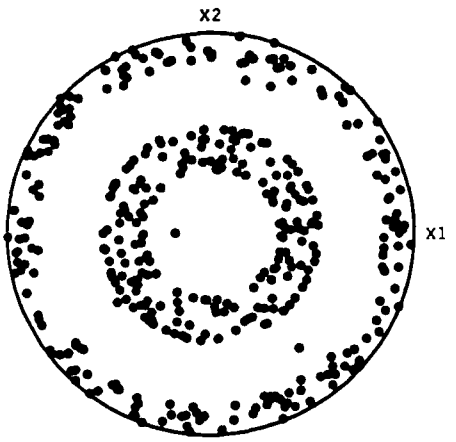
a) A Fibre {111}<uvw> - Ideal Textures in Shear (A- : solid lines, A+ : dashed lines and A : dots)

b) B Fibre {hkl}<110> and C {001}<110> - Ideal Textures in Shear (By : solid lines, Bn : dashed lines and C : dots)

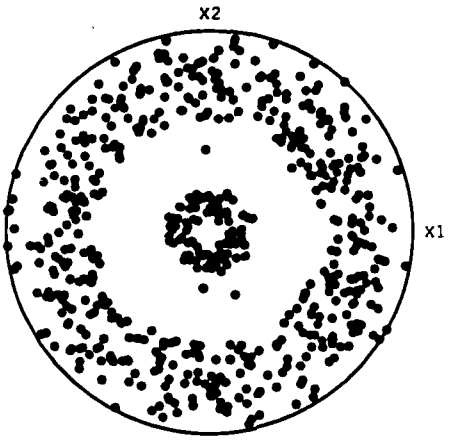
a). A Fibre (A^+ : Solid Lines and A^- : Dashed Lines)

b). B Fibre and C (B^y : Solid Lines, B^n : Dashed Lines and C : Dots)

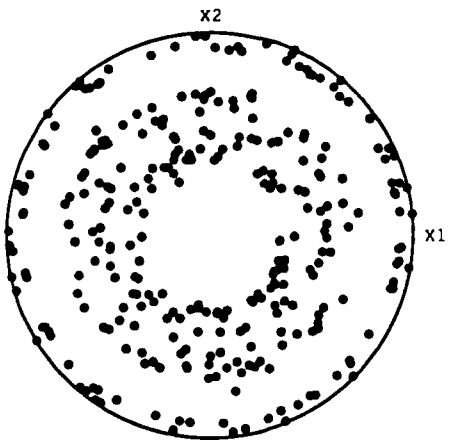
Figure 16. Ideal Texture {111} Pole Figures for Simple Shear



□ (111) Pole figure in uniaxial compression at strain = 1.3
(m=0.012;q=1.4)

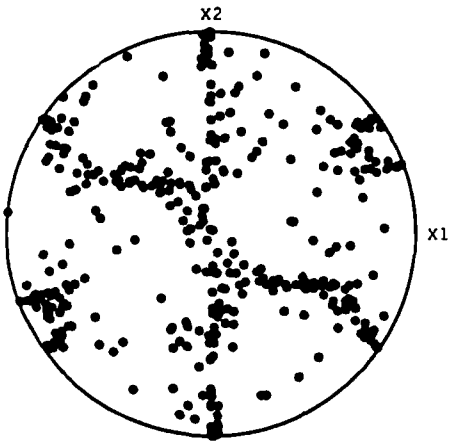


□ (110) Pole figure in uniaxial compression at strain = 1.3
(m=0.012;q=1.4)

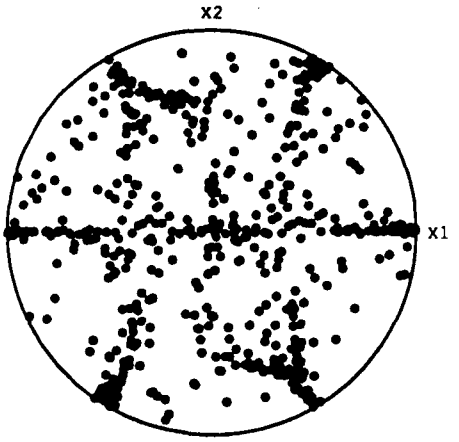


□ (100) Pole figure in uniaxial compression at strain = 1.3
(m=0.012;q=1.4)

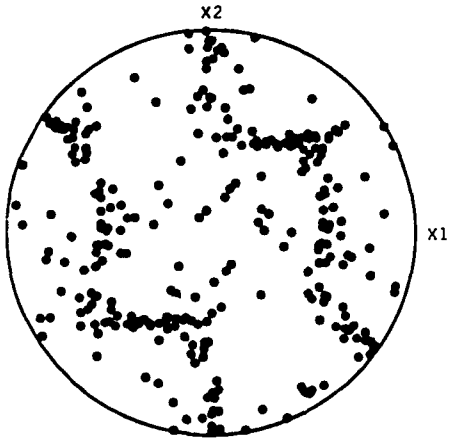
Figure 17. Computed Pole Figures from Rate Dependent Taylor Model for Uniaxial Compression



(111) Pole figure in simple shear at shear strain = 2.8
(m=0.012;q=1.4)



(110) Pole figure in simple shear at shear strain = 2.8
(m=0.012;q=1.4)



(100) Pole figure in simple shear at shear strain = 2.8
(m=0.012;q=1.4)

Figure 18. Computed Pole Figures from Rate Dependent Taylor Model for *Simple Shear*

The presence of A^- fibre is responsible for the compressive axial stresses seen in the predicted as well experimental fixed-end torsion tests. The perturbations in the compressive axial stress with increasing strain is due to the migration of orientations from/to A^- to/from B^y fibre and C orientation.[8]

SUMMARY

Large deformation uniaxial compression and fixed-end and free-end torsion (simple shear) experiments were conducted for annealed OFHC Copper to obtain stress-strain behavior of the material. The stress-strain behavior for these experiments were also predicted using a Taylor type rate dependent polycrystal model by Asaro and Needleman[7]. Simulation of these experiments was conducted using a recently developed, highly efficient, fully implicit time integration scheme by Kalidindi et al.[9] In the initial phase of the simulation, the evolution of the constituent single crystal slip system deformation resistance was estimated using the experimentally determined compressive stress-strain behavior of the polycrystal. With this crystal constitutive behavior, the stress-strain behavior of the polycrystal for simple shear was computed. In addition, evolution of the crystallographic texture was computed for both compression and simple shear tests.

Experimental shear stress was insensitive to changing from fixed-end to free-end condition of testing. The predicted shear stress-strain behavior for simple shear test compared reasonably well with the experimental results. The simulated axial stress response agrees qualitatively (compressive) with the experimental observations, but over estimates the experimentally observed axial compressive stress.

In uniaxial compression tests, the simulated stress-strain behavior decreases with an increase of strain rate sensitivity or decrease of the latent hardening for the same strain. In simple shear, shear stress response is not sensitive to changes in either strain rate sensitivity or latent hardening. The axial and hoop stresses decrease with the increase in strain rate sensitivity. With the decrease of latent hardening, the axial and hoop stresses increase at lower shear strains, but decrease at higher shear strains.

The predicted texture of grains for compression tests assumes an orientation given by the ideal {110}<*uvw*> fibre where {110} planes become perpendicular to the loading axis. For simple shear, predictions show the presence of the ideal A^- ({111}<*uvw*> partial fibre) and B^y ({*hkl*}<110> partial fibre) fibres and C ({001}<110>) orientation, where in this notation, planes are parallel to the shear plane and directions are along the shear direction.

ACKNOWLEDGMENT

The authors are grateful for the Composite Laboratory of MTL for allowing us to conduct the torsion tests and especially Gary Forley of the Composite Group for the help given in conducting these tests. We highly appreciate all the encouragement and resources given by Tony Chou, Chief, Material Dynamic Branch, during this work. Also, one of the authors (TW) appreciates the discussions he had with Suriya Kalidindi of MIT about his algorithm. We were able to debug the computer program in a very short time using the Stephen Wolfram's *Mathematica*, a System for doing Mathematics by Computer, symbolic computation program.

APPENDIX

Computer Program PolyCrystal-MTL (PC-MTL-Ver: 01).

A C version of the program will be available in future.

```
program Tpolycrytal;

{** written by Tusit Weerasooriya of Army Material Technology Lab **}
{**                December 1990                **}

{This program computes the stress history for a given history (including   }
{reverse loading) of deformation gradient tensor (F). Continuum is }
{considered as a poly-crystal of n number of crystals with initially known  }
{orientation (ex. randomly). The program also tracks the evolution of      }
{anisotrophy (crystalographic-texture) with the deformation. State variables}
{- crystal slip resistance and Fp - for each crystal are tracked and are    }
{saved in a restart file. Using the restart file, deformation can be further}
{extended}

const
  ncrystals = 100;
  startcrystal = 1;
  ntimesteps = 400;
  lastcrystal = startcrystal + ncrystals - 1;
  totcrystals = 100;

  qq = 1.0;                          {mult for latent_h for noncoplanar}

  maxstrain = 2.80;                  {max stop strain}
  minstrain = 0.000;                 {min stop strain for reverse}
  startstrain = 0.000;               {starting strain before reverse}
  sdgama = 0.01;                     {maximum delta_gama for iteration}

  ho = 700.0;                        {s_dot=ho(1-s/ss)^aa}
  ss = 155.0;
  so = 16.0;
  aa = 3.80;

  gamadoto = 0.001;                  {reference strain rate}
  m = 0.012;                         {strain rate sensitivity}
  straindot = 0.001;                 {strain rate of the test}
  mu = 44000.0;

type
  table66 = array[1..6, 1..6] of extended;
  table33 = array[1..3, 1..3] of extended;
  rtable33 = array[1..3, 1..3] of real;
  vector3 = array[1..3] of extended;
  vector6 = array[1..6] of extended;
  vector12 = array[1..12] of extended;
  table1212 = array[1..12, 1..12] of real;
  Sc12 = array[1..12] of table33;
  Qvector = array[1..totcrystals] of table33;
  svector = array[1..totcrystals] of vector12;
```

```
  Tvector = array[1..totcrystals] of vector6;
  rtstepv = array[1..ntimesteps] of real;
  tstepv = array[1..ntimesteps] of extended;
  p111 = array[1..8] of vector3;
  p110 = array[1..12] of vector3;
  p100 = array[1..6] of vector3;

var
  answer: string[3];
  restart: string[2];
  testtype, testdirn, strainlimit: char;
  crystal, i1, ltimestep, n, i, j, itemp: integer;
  qab: table1212;
  k, r3km2mud6, r3kd2, sod3, temp, scaler: real;
  Rt, cdgamamax, newdgamamax, ptsdgamamax, dgamamax: real;
  p111c, p111g: p111;
  p110c, p110g: p110;
  p100c, p100g: p100;
  Sc, Sa: Sc12;
  Q: table33;
  eulera: vector3;
  I66, null66: table66;
  I33, null33, temp33, temp133, temp233, Feinter: table33;
  nullv6: vector6;
  nullv12, sov12: vector12;
  Qptr: ^Qvector;
  Fpiptr, nFpiptr: ^Qvector;
  sptr, nsptr: ^svector;
  Tsvptr, nTsvptr: ^Tvector;
  x, y, msdot: real;
  time, strain: rtstepv;
  T11avptr, T22avptr, T33avptr, T12avptr: ^tstepv;
  deltat: real;
  Tfile, pfile, p0file, p00file, pifile, pi0file, pi00file: text;

var
  l, iterate: integer;
  alpha, beta: integer;
  F, Fe, Fpi, Fpinew, R, InU: table33;
  A, Ts, Tstr, Tcauchy: table33;
  Tsnewvec, Tvecerror, Tstrvec, Tfn, temp6: vector6;
  pole, newpole: vector3;
  Ba, Ca: Sc12;
  Savec, Cavec: array[1..12] of vector6;
  Jacobian, IJac, temp66: table66;
  s, snew, serror, st, tau, deltagama, temp12: vector12;
  gamopdelt, s1, s2, s3, s4, s5: extended;
  Tout  sout, iterout: boolean;

{-------------------------------------------------------}

procedure mprint (am: table33;
                  n: integer);
```

```
{print 3x3 in MatrixForm}
  var
    i, j: integer;

  begin
    for i := 1 to n do begin
      writeln;
      for j := 1 to n do
        write(am[i, j] : 10 : 3);
    end; (*for i*)
    writeln;
  end;
{-------------------------------------------------------------}

procedure vprint (av: vector12;
                  n: integer);
{print a 12 element vector}
  var
    i, j: integer;

  begin
    for i := 1 to n do
      write(av[i] : 12 : 6);
    writeln;
  end;{vprint}
{-------------------------------------------------------------}

procedure m6print (am: table66;
                  n: integer);
{print a 6x6 in MatrixForm}

  var
    i, j: integer;

  begin
    for i := 1 to n do begin
      writeln;
      for j := 1 to n do
        write(am[i, j] : 10 : 4);
    end; (*for i*)
    writeln;
  end;
{-------------------------------------------------------------}

procedure v6print (av: vector6;
                  n: integer);
{print vector of 6 elements}
  var
    i, j: integer;

  begin
    for i := 1 to n do begin
      write(av[i] : 12 : 6);
```

```
    end; (*for i*)
    writeln;
  end;{vprint}
{-------------------------------------------------------------}

procedure inverse (am: table33;
                   var bm: table33;
                   n: integer);
{inverse a 3x3 matrix by back substitution}
  var
    i, j, k: integer;
    p: extended;
    s: extended;

  begin
    for i := 1 to n do
      for j := 1 to n do
        bm[i, j] := am[i, j];

    for i := 1 to n do begin
      p := bm[i, i];
      p := 1.0 / p;
      bm[i, i] := 1.0;
      for j := 1 to n do
        bm[i, j] := bm[i, j] * p;
      for k := 1 to n do
        if (k <> i) then begin
          s := bm[k, i];
          bm[k, i] := 0.0;
          for j := 1 to n do
            bm[k, j] := bm[k, j] - s * bm[i, j];
        end;(*if*)
    end;{for i}
  end;{inverse}
{-------------------------------------------------------------}

procedure inverse6 (am: table66;
                    var bm: table66;
                    n: integer);
 {inverse a 6x6 matrix by back substitution}
  var
    i, j, k: integer;
    p: extended;
    s: extended;

  begin
    for i := 1 to n do
      for j := 1 to n do
        bm[i, j] := am[i, j];
    for i := 1 to n do begin
      p := bm[i, i];
      p := 1.0 / p;
      bm[i, i] := 1.0;
```

```
      for j := 1 to n do
        bm[i, j] := bm[i, j] * p;
      for k := 1 to n do
        if (k <> i) then begin
          s := bm[k, i];
          bm[k, i] := 0.0;
          for j := 1 to n do
            bm[k, j] := bm[k, j] - s * bm[i, j];
        end;(*if*)
    end;{for i}
  end;{inverse6}
{-------------------------------------------------------------}

procedure aikbkj (am: table33;
                  bm: table33;
                  var cm: table33);
 {Aik Bkj tensor multiplication}
  var
    i, j, k: integer;

  begin
    for i := 1 to 3 do
      for j := 1 to 3 do begin
        cm[i, j] := 0.0;
        for k := 1 to 3 do
          cm[i, j] := cm[i, j] + am[i, k] * bm[k, j];
      end;{for k}
  end;{aikbkj}
{-------------------------------------------------------------}

procedure aikbjk (am: table33;
                  bm: table33;
                  var cm: table33);
 {Aik Bjk tensor multiplication - A (B)T}
  var
    i, j, k: integer;

  begin
    for i := 1 to 3 do
      for j := 1 to 3 do begin
        cm[i, j] := 0.0;
        for k := 1 to 3 do
          cm[i, j] := cm[i, j] + am[i, k] * bm[j, k];
      end;{for k}
  end;{aikbjk}
{-------------------------------------------------------------}

procedure mplusm (am: table33;
                  bm: table33;
                  var cm: table33);
 {Aij+E ij}
  var
    i, j: integer;
```

```
  begin
    for i := 1 to 3 do
      for j := 1 to 3 do
        cm[i, j] := am[i, j] + bm[i, j];
  end;{mplusm}
{-------------------------------------------------------------}

procedure mdotv (am: table33;
                 av: vector3;
                 var cv: vector3);
 {Aij Bj}
  var
    i, k: integer;

  begin
    for i := 1 to 3 do begin
      cv[i] := 0.0;
      for k := 1 to 3 do
        cv[i] := cv[i] + am[i, k] * av[k];
    end;{for i}
  end;{mdotv}
{-------------------------------------------------------------}

procedure smulm (scaler1: real;
                 am: table33;
                 var bm: table33);
 {s*Aij}
  var
    i, j: integer;

  begin
    for i := 1 to 3 do
      for j := 1 to 3 do
        bm[i, j] := scaler1 * am[i, j];
  end;{smulm}
{-------------------------------------------------------------}

function absmax (av: vector12): real;
{abs(max(Aij))}
  var
    i: integer;

  begin
    temp := abs(av[1]);
    for i := 2 to 12 do
      if (abs(av[i])) > temp then
        temp := abs(av[i]);
    absmax := temp;
  end;{absmax}
{-------------------------------------------------------------}

function det33 (am: table33): real;
```

```
  var
    i: integer;

  begin

    det33 := am[1, 1] * (am[2, 2] * am[3, 3] - am[3, 2] * am[2, 3]) - am[2, 1] * (am[1, 2] * am[3, 3] - am[3, 2] * am[1,
       3]) + am[3, 1] * (am[1, 2] * am[2, 3] - am[2, 2] * am[1, 3]);

  end;{det33}
{-------------------------------------------------------------}

procedure vouterprod (var av, bv: vector3;
                      var cm: table33);
{Ai Bj tensor product}
  var
    i, j: integer;
  begin
    for i := 1 to 3 do
      for j := 1 to 3 do
        cm[i, j] := av[i] * bv[j];
  end;{vouterprod}
{-------------------------------------------------------------}

procedure genSc (var Sc: Sc12;
                 var p111c: p111);

  var
    sqrt2, sqrt3: real;
    i, j: integer;
    s11, s12, s13, s21, s22, s23, s31, s32, s33, s41, s42, s43: vector3;
  begin
    sqrt3 := sqrt(3.0);
    sqrt2 := sqrt(2.0);
    for i := 1 to 3 do begin
      p111c[1, i] := 1.0 / sqrt3;              { 1, 1, 1}
      p111c[2, i] := 1.0 / sqrt3;
      p111c[3, i] := 1.0 / sqrt3;
      p111c[4, i] := 1.0 / sqrt3;
    end;
    p111c[2, 1] := -p111c[2, 1];               {-1, 1, 1}
    p111c[3, 2] := -p111c[3, 2];               { 1,-1, 1}
    p111c[4, 3] := -p111c[4, 3];               { 1, 1,-1}
    for i := 1 to 3 do begin
      p111c[5, i] := -p111c[1, i];             {-1,-1,-1}
      p111c[6, i] := -p111c[2, i];             { 1,-1,-1}
      p111c[7, i] := -p111c[3, i];             {-1, 1,-1}
      p111c[8, i] := -p111c[4, i];             {-1,-1, 1}
    end;
    s11[1] := -1.0 / sqrt2;
    s11[2] := 1.0 / sqrt2;
    s11[3] := 0.0;                             {-1, 1, 0}
    s12[1] := -1.0 / sqrt2;
    s12[2] := 0.0;
```

```
    s12[3] := 1.0 / sqrt2;                  {-1, 0, 1}
    s13[1] := 0.0;
    s13[2] := -1.0 / sqrt2;
    s13[3] := 1.0 / sqrt2;                  { 0,-1, 1}

    s21[1] := 1.0 / sqrt2;
    s21[2] := 1.0 / sqrt2;
    s21[3] := 0.0;                          { 1, 1, 0}
    s22[1] := 1.0 / sqrt2;
    s22[2] := 0.0;
    s22[3] := 1.0 / sqrt2;                  { 1, 0, 1}
    s23[1] := 0.0;
    s23[2] := -1.0 / sqrt2;
    s23[3] := 1.0 / sqrt2;                  { 0,-1, 1}

    s31[1] := 1.0 / sqrt2;
    s31[2] := 1.0 / sqrt2;
    s31[3] := 0.0;                          { 1, 1, 0}
    s32[1] := 0.0;
    s32[2] := 1.0 / sqrt2;
    s32[3] := 1.0 / sqrt2;                  { 0, 1, 1}
    s33[1] := 1.0 / sqrt2;
    s33[2] := 0.0;
    s33[3] := -1.0 / sqrt2;                 { 1, 0,-1}

    s41[1] := 0.0;
    s41[2] := 1.0 / sqrt2;
    s41[3] := 1.0 / sqrt2;                  { 0, 1, 1}
    s42[1] := 1.0 / sqrt2;
    s42[2] := 0.0;
    s42[3] := 1.0 / sqrt2;                  { 1, 0, 1}
    s43[1] := 1.0 / sqrt2;
    s43[2] := -1.0 / sqrt2;
    s43[3] := 0.0;                          { 1,-1, 0}

    for i := 1 to 3 do
      for j := 1 to 3 do begin
        Sc[1, i, j] := s11[i] * p111c[1, j];
        Sc[2, i, j] := s12[i] * p111c[1, j];
        Sc[3, i, j] := s13[i] * p111c[1, j];
        Sc[4, i, j] := s21[i] * p111c[2, j];
        Sc[5, i, j] := s22[i] * p111c[2, j];
        Sc[6, i, j] := s23[i] * p111c[2, j];
        Sc[7, i, j] := s31[i] * p111c[3, j];
        Sc[8, i, j] := s32[i] * p111c[3, j];
        Sc[9, i, j] := s33[i] * p111c[3, j];
        Sc[10, i, j] := s41[i] * p111c[4, j];
        Sc[11, i, j] := s42[i] * p111c[4, j];
        Sc[12, i, j] := s43[i] * p111c[4, j];
      end;{for j}
  end;{genSc}
{-------------------------------------------------------}
```

```
procedure genQ (var Q: table33;
                eulerangle: vector3);

  var
    phi, omega, theta: real;
    cphi, sphi, comega, somega, ctheta, stheta: real;
  begin
    phi := eulerangle[1];     {0=<phi=<360}
    theta := eulerangle[2];     {0=<theta=<180}
    omega := eulerangle[3];     {0=<omega=<360}
    cphi := cos(phi);
    sphi := sin(phi);
    ctheta := cos(theta);
    stheta := sin(theta);
    comega := cos(omega);
    somega := sin(omega);
    Q[1, 1] := cphi * comega - sphi * somega * ctheta;
    Q[1, 2] := sphi * comega + cphi * somega * ctheta;
    Q[1, 3] := somega * stheta;
    Q[2, 1] := -cphi * somega - sphi * comega * ctheta;
    Q[2, 2] := -sphi * somega + cphi * comega * ctheta;
    Q[2, 3] := comega * stheta;
    Q[3, 1] := sphi * stheta;
    Q[3, 2] := -cphi * stheta;
    Q[3, 3] := ctheta;
  end;{genQ}
{-------------------------------------------------------------}

procedure eulerread (totcrystals: integer);

  var
    i: integer;
    eafile: text;

  begin
    open(eafile, 'text.300');
    reset(eafile);

    temp := 3.14159265 / 180.0;
    for i := 1 to totcrystals do begin
      readln(eafile, eulera[2], eulera[1], eulera[3]);
      eulera[1] := temp * eulera[1];  {phi}
      eulera[2] := temp * eulera[2];  {theta}
      eulera[3] := temp * eulera[3];  {omega}

      genQ(Qptr^[i], eulera);

    end;{for i}
    close(eafile);
  end;{eulerread}
{-------------------------------------------------------------}

function trace (am: table33): real;
```

```
{calculate the trace of a matrix - Akk}
  var
    i: integer;
  begin
    scaler := 0.0;
    for i := 1 to 3 do
      scaler := scaler + am[i, i];
    trace := scaler;
  end;{trace}
{-------------------------------------------------------------}
procedure symmtov (am: table33;
                   var avec: vector6);
{vectorise a symmetric matrix}
  var
    i: integer;
  begin
    for i := 1 to 3 do
      avec[i] := am[i, i];
    avec[4] := am[1, 2];
    avec[5] := am[1, 3];
    avec[6] := am[2, 3];
  end;{symmtov}
{-------------------------------------------------------------}

procedure symmtopv (am: table33;
                    var avec: vector6);
{vectorise a matrix}
  var
    i, j: integer;
  begin
    for i := 1 to 3 do
      for j := 1 to 3 do
        temp33[i, j] := 0.5 * (am[i, j] + am[j, i]);


    for i := 1 to 3 do
      avec[i] := temp33[i, i];
    avec[4] := 2.0 * temp33[1, 2];
    avec[5] := 2.0 * temp33[1, 3];
    avec[6] := 2.0 * temp33[2, 3];
  end;{symmtopv}
{-------------------------------------------------------------}

procedure vtosymm (avec: vector6;
                   var am: table33);
{convert the 6 element vector to the corresponding symmetric 3x3 matrix}
  var
    i, j: integer;
  begin

    for i := 1 to 3 do
      am[i, i] := avec[i];
    am[1, 2] := avec[4];
```

```
    am[1, 3] := avec[5];
    am[2, 3] := avec[6];
    am[2, 1] := am[1, 2];
    am[3, 1] := am[1, 3];
    am[3, 2] := am[2, 3];
  end;{vtosymm}
{-------------------------------------------------------------}

procedure eapolexy (hkl: vector3;
                    var x, y: real);
(* equal area projection pole figure generation *)
  begin

    {equal area-projection pole figure}
    temp := sqrt(1.0 + hkl[3]);
    x := hkl[1] / temp;
    y := hkl[2] / temp;

    {stereographic-projection pole figure}
    {temp:=1.0+hkl[3];}
    {x:=hkl[1]/(1.0+hkl[3]);}
    {y:=hkl[2]/(1.0+hkl[3]);}

  end;{eapolexy}
{-------------------------------------------------------------}

procedure RUdecomp (F: table33;
                    var R, InU: table33);
{polar decomposition of F = RU}
  var
    c, c2: table33;
    i, j, k: integer;
    ic, iic, iiic, iu, iiu, iiiu, ic2, ic3, iic2, iunew: extended;
    a0, a1, a2, a3, a4, iu3, srtiiic: extended;
  begin
    c := null33;
    for i := 1 to 3 do
      for j := i to 3 do
        for k := 1 to 3 do
          c[i, j] := c[i, j] + F[k, i] * F[k, j];
    c[2, 1] := c[1, 2];
    c[3, 1] := c[1, 3];
    c[3, 2] := c[2, 3];
    {c^2}
    c2 := null33;
    for i := 1 to 3 do
      for j := i to 3 do
        for k := 1 to 3 do
          c2[i, j] := c2[i, j] + c[i, k] * c[k, j];
    c2[2, 1] := c2[1, 2];
    c2[3, 1] := c2[1, 3];
    c2[3, 2] := c2[2, 3];
    {calculate inavriants of c}
```

```
    ic := c[1, 1] + c[2, 2] + c[3, 3];
    iic := (ic * ic - (c2[1, 1] + c2[2, 2] + c2[3, 3])) / 2.0;
    iiic := det33(c);
    {calculate iiiu=sqrt(iiic)}
    iiiu := sqrt(iiic);
    {calculate iu}
    iunew := ic;
    repeat
      iu := iunew;
      iu3 := iu * iu * iu;
      srtiiic := sqrt(iiic);
      iunew := iu - (iu * iu3 - 2.0 * ic * iu * iu - 8.0 * iu * srtiiic + ic * ic - 4.0 * iic) / (4.0 * iu3 - 4.0 * ic * iu - 8.0 *
        srtiiic);
    until abs(iunew - iu) < 0.000001;
    iu := iunew;
    iiu := sqrt(iic + 2.0 * iu * sqrt(iiic));
    {U^-1}
    for i := 1 to 3 do
      for j := 1 to 3 do begin
        temp33[i, j] := c[i, j] + (iiiu / iu) * I33[i, j];
        temp233[i, j] := c[i, j] + iiu * I33[i, j];
      end;{for j}
    inverse(temp33, temp133, 3);
    aikbkj(temp133, temp233, InU);
    for i := 1 to 3 do
      for j := 1 to 3 do
        InU[i, j] := InU[i, j] / iu;
    aikbkj(F, InU, R);
{    mprint(R,3);}

  end;{RUdecomp}
{-------------------------------------------------------}

procedure initialize;
  var
    crystal, i, j: integer;
    sqrt2: real;
    rsFpiptr: file of Qvector;
    rsTsvptr: file of Tvector;
    rssptr: file of svector;
    rsreal: file of real;

  begin

    for i := 1 to 3 do
      for j := 1 to 3 do
        if (i = j) then begin
          I33[i, j] := 1.0;
          null33[i, j] := 0.0;
        end
        else begin
          I33[i, j] := 0.0;
          null33[i, j] := 0.0;
```

```
    end;
for i := 1 to 6 do
  for j := 1 to 6 do
    if (i = j) then begin
      I66[i, j] := 1.0;
      null66[i, j] := 0.0;
    end
    else begin
      I66[i, j] := 0.0;
      null66[i, j] := 0.0;
    end;{else}
for i := 1 to 6 do
  nullv6[i] := 0.0;
for i := 1 to 12 do begin
  nullv12[i] := 0.0;
  sov12[i] := so;
end;{i}

sqrt2 := sqrt(2.0);
for j := 1 to 3 do
  for i := 1 to 3 do
    p110c[j, i] := 1.0 / sqrt2;

p110c[1, 1] := 0.0;                    { 0, 1, 1}
p110c[2, 2] := 0.0;                    { 1, 0, 1}
p110c[3, 3] := 0.0;                    { 1, 1, 0}

for j := 4 to 6 do
  for i := 1 to 3 do
    p110c[j, i] := p110c[j - 3, i];
p110c[4, 2] := -p110c[4, 2];           { 0,-1, 1}
p110c[5, 1] := -p110c[5, 1];           {-1, 0, 1}
p110c[6, 1] := -p110c[6, 1];           {-1, 1, 0}

for j := 7 to 12 do                    { 0,-1,-1}
                                       {-1, 0 -1}
                                       {-1,-1, 0}
  for i := 1 to 3 do                   { 0, 1,-1}
                                       { 1, 0,-1}
                                       { 1,-1, 0}
    p110c[j, i] := -p110c[j - 6, i];

for j := 1 to 6 do
  for i := 1 to 3 do
    p100c[j, i] := 0.0;
p100c[1, 1] := 1.0;                    { 1, 0, 0}
p100c[2, 2] := 1.0;                    { 0, 1, 0}
p100c[3, 3] := 1.0;                    { 0, 0, 1}
p100c[4, 1] := -1.0;                   {-1, 0, 0}
p100c[5, 2] := -1.0;                   { 0,-1, 0}
p100c[6, 3] := -1.0;                   { 0, 0,-1}

for i := 1 to 12 do
```

```
  for j := 1 to 12 do
    if (((i >= 1) and (i <= 3)) and ((j >= 1) and (j <= 3))) or (((i >= 4) and (i <= 6)) and ((j >= 4) and (j <= 6))) or
    (((i >= 7) and (i <= 9)) and ((j >= 7) and (j <= 9))) or (((i >= 10) and (i <= 12)) and ((j >= 10) and (j <= 12)))
    then
      qab[i, j] := 1.0 * ho
    else
      qab[i, j] := qq * ho;

if (restart = 'nn') then begin

  open(pifile, 'pi111.pr2');
  rewrite(pifile);
  open(pi0file, 'pi110.pr2');
  rewrite(pi0file);
  open(pi00file, 'pi100.pr2');
  rewrite(pi00file);
  rewrite(Tfile);
  rewrite(pfile);
  rewrite(p0file);
  rewrite(p00file);

  time[1] := 0.0;
  strain[1] := 0.0;
  T12avptr^[1] := 0.0;
  T11avptr^[1] := 0.0;
  T22avptr^[1] := 0.0;
  T33avptr^[1] := 0.0;

  writeln(Tfile, '    STRAIN     T12av     T33av     T22av     T11av     T33av-p ');
  writeln(Tfile, strain[1] : 12 : 4, T12avptr^[1] : 12 : 4, T33avptr^[1] : 12 : 4, T22avptr^[1] : 12 : 4, T11avptr^[1] : 12
   : 4, -(T33avptr^[1] - (T11avptr^[1] + T22avptr^[1]) / 2.0) : 12 : 4);

  for crystal := startcrystal to lastcrystal do begin
    sptr^[crystal] := sov12;
    Fpiptr^[crystal] := I33;
    Tsvptr^[crystal] := nullv6;

    writeln(pifile, '');
    for i := 1 to 8 do begin
      mdotv(Qptr^[crystal], p111c[i], p111g[i]);
      if p111g[i, 3] >= 0.0 then begin
        eapolexy(p111g[i], x, y);
        write(pifile, x : 10 : 4, y : 10 : 4);
      end;{if}
    end;{for i}

    writeln(pi0file, '');
    for i := 1 to 12 do begin
      mdotv(Qptr^[crystal], p110c[i], p110g[i]);
      if p110g[i, 3] >= 0.0 then begin
        eapolexy(p110g[i], x, y);
        write(pi0file, x : 10 : 4, y : 10 : 4);
      end;{if}
```

```
        end;{for i}

        writeln(pi00file, '');
        for i := 1 to 6 do begin
          mdotv(Qptr^[crystal], p100c[i], p100g[i]);
          if p100g[i, 3] >= 0.0 then begin
            eapolexy(p100g[i], x, y);
            write(pi00file, x : 10 : 4, y : 10 : 4);
          end;{if}
        end;{for i}
      end;{crystal}
      close(pifile);
      close(pi0file);
      close(pi00file);

    end{if}

    else if (restart = 'yy') or (restart = 'ry') then begin

      open(rsFpiptr, 'rsFpptr');
      open(rssptr, 'rssptr');
      open(rsTsvptr, 'rsTsvptr');
      open(rsreal, 'rsreal');

      reset(rsFpiptr);
      reset(rssptr);
      reset(rsTsvptr);
      reset(rsreal);

      read(rsFpiptr, Fpiptr^);
      read(rssptr, sptr^);
      read(rsTsvptr, Tsvptr^);
      read(rsreal, time[1], strain[1], deltat);

      close(rsFpiptr);
      close(rssptr);
      close(rsTsvptr);
      close(rsreal);

{   append(Tfile); for PC}
      seek(Tfile, maxlongint);        {for Mac}
      rewrite(pfile);
      rewrite(p0file);
      rewrite(p00file);
    end;{else}

  end;{initialize}
{-------------------------------------------------------------}

procedure saversfiles;
 {save intermediate data for restarting the calculation}
  var
    rsFpiptr: file of Qvector;
```

```
    rsTsvptr: file of Tvector;
    rssptr: file of svector;
    rsreal: file of real;

  begin

    open(rsFpiptr, 'rsFpptr');
    open(rssptr, 'rssptr');
    open(rsTsvptr, 'rsTsvptr');
    open(rsreal, 'rsreal');

    rewrite(rsFpiptr);
    rewrite(rssptr);
    rewrite(rsTsvptr);
    rewrite(rsreal);

    write(rsFpiptr, Fpiptr^);
    write(rssptr, sptr^);
    write(rsTsvptr, Tsvptr^);
    write(rsreal, time[ltimestep], strain[ltimestep], deltat);

    close(rsFpiptr);
    close(rssptr);
    close(rsTsvptr);
    close(rsreal);
  end;{saversfiles}
{-------------------------------------------------------}

function sign (etemp: extended): real;
 {if negative return -1.0 and otherwise return 1.0}
  var
    s1: real;
  begin
    if etemp < 0.0 then
      s1 := -1.0
    else
      s1 := 1.0;
    sign := s1;
  end;{sign}
{-------------------------------------------------------}

procedure tau_dgama;
 {calculate tau and delta_gama for each slip system}
  var
    i, alpha: integer;
    s1, s2, s3, s4, s5: extended;
  begin

    tau := nullv12;
    for alpha := 1 to 12 do begin
      for i := 1 to 6 do
        tau[alpha] := tau[alpha] + nTsvptr^[crystal][i] * Savec[alpha, i];
```

```
      if (abs(tau[alpha]) < so / 100000.0) then
        deltagama[alpha] := 0.0
      else begin
        s3 := abs(tau[alpha] / nsptr^[crystal, alpha]);
        s4 := ln(s3) / m;
        deltagama[alpha] := gamopdelt * sign(tau[alpha]) * exp(s4);
      end;{if}
    end;{for alpha}

  end;{tau_dgama}
{-------------------------------------------------------------}

  label
    10, 20;

{*************end of procedures, functions and labels************}

begin

  k := 3.115 * mu;                          {k - bulk modulus}
  r3km2mud6 := (3.0 * k - 2.0 * mu) / 6.0;
  r3kd2 := (3.0 * k) / 2.0;
  sod3 := so / 3.0;
  ltimestep := ntimesteps;                  {last time step}

  dgamamax := 0.0;                          {maximum delta_gama for iteration}

  writeln('start');

  new(Fpiptr);
  new(sptr);
  new(Qptr);
  new(Tsvptr);

  new(nFpiptr);
  new(nsptr);
  new(nTsvptr);

  new(T11avptr);
  new(T22avptr);
  new(T33avptr);
  new(T12avptr);

  write('Restart yy/nn/ry :');
  readln(restart);

  write('TestType s/c/t :');
  readln(testtype);

  write('TestDirn f/r :');
  readln(testdirn);

  strainlimit := 'n';
```

```
msdot := 1.0;
if testdirn = 'r' then
  msdot := -1.0;

if testtype = 's' then begin
  open(Tfile, 'TS.pr2');
  open(pfile, 'p0s.pr2');
  open(p0file, 'p1s.pr2');
  open(p00file, 'p2s.pr2');
end {if}
else if (testtype = 'c') or (testtype = 't') then begin
  open(Tfile, 'TC.pr2');
  open(pfile, 'p0c.pr2');
  open(p0file, 'p1c.pr2');
  open(p00file, 'p2c.pr2');
end;

eulerread(totcrystals);

genSc(Sc, p111c);

initialize;

if (restart = 'nn') or (restart = 'ry') then begin
  deltat := 0.1;{2;}
  for i1 := 2 to ntimesteps do begin
    if i1 > 2 then
      deltat := 0.2;
    if i1 > 5 then
      deltat := 0.3;
    if i1 > 7 then
      deltat := 0.4;
    if i1 > 10 then
      deltat := 0.6;
    if i1 > 15 then
      deltat := 1.0;
    if i1 > 20 then
      deltat := 2.0;
    if i1 > 30 then
      deltat := 4.0;
    if i1 > 50 then
      deltat := 7.0;
    if i1 > 80 then
      deltat := 8.0;
    if i1 > 100 then
      deltat := 10.0;
    if i1 > 120 then
      deltat := 15.0;
    time[i1] := time[i1 - 1] + (deltat / abs(straindot)) * 0.001;
    strain[i1] := 2.0 * startstrain + msdot * straindot * time[i1];
{     writeln(time[i1]);}
  end;{for i1}
  writeln('Max. Strain = ', strain[ntimesteps]);
```

```
  end;{if}

  for i1 := 2 to ntimesteps do begin
10:
    if (i1 <= 10) and ((restart = 'nn') or (restart = 'ry')) then begin
      deltat := time[i1] - time[i1 - 1];
      strain[i1] := 2.0 * startstrain + msdot * straindot * time[i1];
    end {if}
    else begin
      time[i1] := time[i1 - 1] + deltat;
      strain[i1] := 2.0 * startstrain + msdot * straindot * time[i1];
    end;{else}

    ptsdgamamax := dgamamax;
    dgamamax := 0.0;

    s1 := strain[i1]; {straindot * time[i1];}

    if testtype = 's' then begin
      F := I33;
      F[1, 2] := s1;
    end
    else begin
      if testtype = 'c' then begin
        F := null33;
        F[1, 1] := exp(s1 / 2.0);
        F[2, 2] := F[1, 1];
        F[3, 3] := exp(-s1);
      end
      else begin
        F := null33;
        F[1, 1] := exp(-s1 / 2.0);
        F[2, 2] := F[1, 1];
        F[3, 3] := exp(s1);
      end;{else}
    end;{else}

    mprint(F, 3);

    nTsvptr^ := Tsvptr^;
    nsptr^ := sptr^;

    cdgamamax := 0.0;
    for crystal := startcrystal to lastcrystal do begin
      writeln;
      writeln('** Crystal = ', crystal : 4, ' **');

      for alpha := 1 to 12 do begin
      {write('Sc', alpha);}
      {mprint(Sc[alpha], 3);}
        aikbkj(Qptr^[crystal], Sc[alpha], temp33);

        Sa[alpha] := null33;
```

```
      for i := 1 to 3 do
        for j := 1 to 3 do
          for l := 1 to 3 do
            Sa[alpha, i, j] := Sa[alpha, i, j] + temp33[i, l] * Qptr^[crystal, j, l];
      symmtopv(Sa[alpha], Savec[alpha]);
    {v6print(Savec[alpha],6);}
    end;{for alpha}


    writeln('Crystal=', crystal : 4, ' : time[', i1, ']=', time[i1] : 8 : 2, ' : strain[', i1, ']=', strain[i1] : 8 : 3, ' : dt=', deltat :
     8 : 4);

    aikbkj(F, Fpiptr^[crystal], Feinter);
  {A = Transpose[Feinter].Feinter}
    A := null33;
    for i := 1 to 3 do
      for j := 1 to 3 do
        for l := 1 to 3 do
          A[i, j] := A[i, j] + Feinter[l, i] * Feinter[l, j];
{   writeln('A : ');}
{    mprint(A, 3);}
    s1 := r3km2mud6 * trace(A);
    s3 := s1 - r3kd2;
    for i := 1 to 3 do
      for j := 1 to 3 do
        Tstr[i, j] := mu * A[i, j] + s3 * I33[i, j];
{    mprint(Tstr, 3);}
    symmtov(Tstr, Tstrvec);

    for alpha := 1 to 12 do begin
      Ba[alpha] := null33;
      for i := 1 to 3 do
        for j := 1 to 3 do
          for l := 1 to 3 do
            Ba[alpha, i, j] := Ba[alpha, i, j] + A[i, l] * Sa[alpha, l, j] + Sa[alpha, l, i] * A[l, j];
{     writeln('Ba[',alpha,']');}
{     mprint(Ba[alpha],3);}
      s1 := trace(Ba[alpha]) * r3km2mud6;
      for i := 1 to 3 do
        for j := 1 to 3 do
          Ca[alpha, i, j] := mu * Ba[alpha, i, j] + s1 * I33[i, j];

{     writeln('Ca[',alpha,']');}
{     mprint(Ca[alpha],3);}
      symmtov(Ca[alpha], Cavec[alpha]);
    end;{for alpha}
    st := nsptr^[crystal];

    gamopdelt := gamadoto * deltat;
    iterout := false;
    iterate := 1;
        {for iterate := 1 to 20 do begin}
    repeat
```

```
        tau_dgama;
{               vprint(tau, 12);}
{               vprint(deltagama, 12);}
        newdgamamax := absmax(deltagama);
{       writeln('ndgamax, dgamax ,ptsdgamax = ',}
{          newdgamamax:10:4,' : ',dgamamax:10:4,' : ',ptsdgamamax:10:4);}
        if (newdgamamax > dgamamax) then
          dgamamax := newdgamamax;
        temp6 := nullv6;
        for i := 1 to 6 do begin
          for alpha := 1 to 12 do
            temp6[i] := temp6[i] + deltagama[alpha] * Cavec[alpha, i];
          Tfn[i] := nTsvptr^[crystal][i] - Tstrvec[i] + temp6[i];
        end;{for i}
        writeln('C = ', crystal, ' : Tstep = ', i1, ' : iterate = ', iterate);
{        v6print(Tfn,6);}

        Jacobian := I66;
        for i := 1 to 6 do
          for j := 1 to 6 do
            for alpha := 1 to 12 do
              if (abs(tau[alpha]) > (so / 100000.0)) then begin
                s1 := deltagama[alpha] / (m * tau[alpha]);
                Jacobian[i, j] := Jacobian[i, j] + s1 * Cavec[alpha, i] * Savec[alpha, j];
              end;{if}
{         writeln('J : ');}
{         m6print(Jacobian,6);}
        inverse6(Jacobian, IJac, 6);
{         writeln('IJ : ');}
{         m6print(IJac,6);}
        temp6 := nullv6;
        for i := 1 to 6 do begin
          for j := 1 to 6 do
            temp6[i] := temp6[i] + IJac[i, j] * Tfn[j];
          Tsnewvec[i] := nTsvptr^[crystal][i] - temp6[i];
          Tvecerror[i] := Tsnewvec[i] - nTsvptr^[crystal][i];
        end;{for i}

        Tout := true;
        for i := 1 to 6 do begin
          if ((abs(Tvecerror[i])) > so / 100000.0) then begin
            Tout := false;
            if (abs(Tvecerror[i]) > 2.0 * sod3) then
{                       Tsnewvec[i] := Tsnewvec[i] + nTsvptr^[crystal][i])/2.0}
{             else}
              Tsnewvec[i] := nTsvptr^[crystal][i] + sod3 * sign(Tvecerror[i]);
          end;{if}
        end;{for i}
        nTsvptr^[crystal] := Tsnewvec;
        {v6print(Tsnewvec,6);}
{         writeln('Tout : ',Tout);}

        if Tout then begin
```

```
            tau_dgama;
            temp12 := nullv12;
            for alpha := 1 to 12 do begin
              s3 := 0.0;
              for beta := 1 to 12 do begin
                s1 := 1.0 - nsptr^[crystal, beta] / ss;
                if (s1 > 0.0) then begin
                  s2 := aa * ln(s1);
                  s1 := exp(s2);  {ho embedded in qab}
                end
                else
                  s1 := 0.0;
                temp12[alpha] := temp12[alpha] + qab[alpha, beta] * s1 * abs(deltagama[beta]);
              end;{for beta}
              snew[alpha] := st[alpha] + temp12[alpha];
              serror[alpha] := snew[alpha] - nsptr^[crystal, alpha];
            end;{for alpha}
            nsptr^[crystal] := snew;
            sout := true;
            s1 := absmax(serror);
            if s1 > so / 100.0 then begin
              sout := false;
            end;{if}
            write('s :');
            vprint(nsptr^[crystal], 6);
          end;{if Tout}
{           writeln('sout : ',sout);}
          iterate := iterate + 1;
          if (Tout and sout) then
            iterout := true;

        until (iterout or (iterate > 80));
          {end;for iterate}

        writeln('ndgamax, dgamax ,ptsdgamax = ', newdgamamax : 10 : 4, ' : ', dgamamax : 10 : 4, ' : ', ptsdgamamax : 10
          : 4);

        if (newdgamamax > cdgamamax) then
          cdgamamax := newdgamamax;

{for Mac}
{       if (i1>10) or (restart='y') then}
{         if (iterate>=60) or (dgamamax>100000000.0) then begin}
{           writeln('**** deltat is large - unstable - reduce!! ****');}
{           deltat := 0.75*deltat;}
{            goto 10:}
{       end;}
{if}

        temp233 := I33;
        for alpha := 1 to 12 do begin
          smulm(deltagama[alpha], Sa[alpha], temp133);
          for i := 1 to 3 do
```

```
        for j := 1 to 3 do
          temp233[i, j] := temp233[i, j] - temp133[i, j];
      end;{for alpha}
      aikbkj(Fpiptr^[crystal], temp233, temp133);
      s1 := det33(temp133);
      s2 := exp(ln(s1) / 3.0);
      s1 := 1 / s2;
      smulm(s1, temp133, Fpinew);
{     writeln('Fpinew : ');}
{     mprint(Fpinew,3);}
      nFpiptr^[crystal] := Fpinew;

      aikbkj(F, Fpinew, Fe);
{     writeln('Fe : ');}
{     mprint(Fe,3);}
      vtosymm(nTsvptr^[crystal], Ts);
      aikbkj(Fe, Ts, temp33);
      s1 := det33(Fe);
      Tcauchy := null33;
      for i := 1 to 3 do
        for j := 1 to 3 do begin
          for l := 1 to 3 do
            Tcauchy[i, j] := Tcauchy[i, j] + temp33[i, l] * Fe[j, l];
          Tcauchy[i, j] := Tcauchy[i, j] / s1;
        end;{for j}
      writeln('Tcauchy : ');
      mprint(Tcauchy, 3);

      if (crystal = startcrystal) then begin
        T12avptr^[i1] := Tcauchy[1, 2];
        T11avptr^[i1] := Tcauchy[1, 1];
        T22avptr^[i1] := Tcauchy[2, 2];
        T33avptr^[i1] := Tcauchy[3, 3];
      end {if}
      else begin
        itemp := crystal - startcrystal + 1;
        T12avptr^[i1] := T12avptr^[i1] + (Tcauchy[1, 2] - T12avptr^[i1]) / itemp;
        T11avptr^[i1] := T11avptr^[i1] + (Tcauchy[1, 1] - T11avptr^[i1]) / itemp;
        T22avptr^[i1] := T22avptr^[i1] + (Tcauchy[2, 2] - T22avptr^[i1]) / itemp;
        T33avptr^[i1] := T33avptr^[i1] + (Tcauchy[3, 3] - T33avptr^[i1]) / itemp;
      end; {else}

      if testdirn = 'f' then
        if (strain[i1] > maxstrain) then
          strainlimit := 'y';
      if testdirn = 'r' then
        if (strain[i1] < minstrain) then
          strainlimit := 'y';

      if (i1 = ntimesteps) or (strainlimit = 'y') then begin
        RUdecomp(Fe, R, InU);
        aikbkj(R, Qptr^[crystal], temp33);
```

```
      writeln(pfile, '');
      for i := 1 to 8 do begin
        mdotv(temp33, p111c[i], newpole);
        if (newpole[3] >= 0.0) then begin
          eapolexy(newpole, x, y);
          write(pfile, x : 10 : 4, y : 10 : 4);
        end;{if}
      end;{for i}

      writeln(p0file, '');
      for i := 1 to 12 do begin
        mdotv(temp33, p110c[i], newpole);
        if (newpole[3] >= 0.0) then begin
          eapolexy(newpole, x, y);
          write(p0file, x : 10 : 4, y : 10 : 4);
        end;{if}
      end;{for i}

      writeln(p00file, '');
      for i := 1 to 6 do begin
        mdotv(temp33, p100c[i], newpole);
        if (newpole[3] >= 0.0) then begin
          eapolexy(newpole, x, y);
          write(p00file, x : 10 : 4, y : 10 : 4);
        end;{if}
      end;{for i}

    end;{if}

  end;{for crystal}

  writeln('dt= ', deltat : 8 : 3, ' time[', i1, ']= ', time[i1] : 8 : 2, ' strain=', strain[i1] : 8 : 4);
  Rt := cdgamamax / sdgama;

  {auto time increment selection}
  if dgamamax < 1000000.0 then begin
    if Rt <= 0.8 then
      deltat := 1.25 * deltat
    else if (Rt > 0.8) and (Rt < 1.25) then
      deltat := deltat / Rt
    else begin
      writeln('**** deltat is large - reduce! ****');
      deltat := 0.75 * deltat;
    end;{else}
  end {if}
  else begin
    writeln('**** deltat is large - unstable - reduce!! ****');
    deltat := 0.75 * deltat;
  end;{else}

  writeln('cdgamamax, Rt :', cdgamamax : 10 : 4, Rt : 10 : 4);
  writeln('newdeltat = ', deltat : 10 : 4);
```

```
      Fpiptr^ := nFpiptr^;
      sptr^ := nsptr^;
      Tsvptr^ := nTsvptr^;

      writeln(Tfile, strain[i1] : 12 : 4, T12avptr^[i1] : 12 : 4, T33avptr^[i1] : 12 : 4, T22avptr^[i1] : 12 : 4, T11avptr^[i1] :
          12 : 4, -(T33avptr^[i1] - (T11avptr^[i1] + T22avptr^[i1]) / 2.0) : 12 : 4);

      if strainlimit = 'y' then begin
        ltimestep := i1;
        goto 20;
      end;{if}

    end;{for i1}
20:

    writeln(i1 : 5, crystal : 5);

    close(Tfile);
    close(pfile);
    close(p0file);
    close(p00file);

    saversfiles;

end.
```

DISTRIBUTION LIST

No. of Copies	To
1	Office of the Under Secretary of Defense for Research and Engineering, The Pentagon, Washington, DC 20301
	Commander, U.S. Army Materiel Command, 5001 Eisenhower Avenue, Alexandria, VA 22333-0001
1	ATTN: AMCLD
	Commander, U.S. Army Laboratory Command, 2800 Powder Mill Road, Adelphi, MD 20783-1145
1	ATTN: AMSLC-IM-TL
1	AMSLC-CT
	Commander, Defense Technical Information Center, Cameron Station, Building 5, 5010 Duke Street, Alexandria, VA 22304-6145
2	ATTN: DTIC-FDAC
1	MIAC/CINDAS, Purdue University, 2595 Yeager Rd., West Lafayette, IN 47905
	Commander, Army Research Office, P.O. Box 12211, Research Triangle Park, NC 27709-2211
1	ATTN: Information Processing Office
	Commander, U.S. Army Electronics Technology and Devices Laboratory, Fort Monmouth, NJ 07703-5000
1	ATTN: SLCET-DT
	Commander, U.S. Army Missile Command, Redstone Arsenal, AL 35898-5247
1	ATTN: AMSMI-RD-CS-R/Doc
1	Technical Library
	Commander, U.S. Army Armament, Munitions and Chemical Command, Dover, NJ 07801
2	ATTN: SMCAR-TDC
	Commander, U.S. Army Natick Research, Development and Engineering Center, Natick, MA 01760
1	ATTN: Technical Library
	Commander, U.S. Army Tank-Automotive Command, Warren, MI 48397-5000
1	ATTN: AMSTA-R
	Commander, U.S. Army Engineer Waterways Experiment Station, P.O. Box 631, Vicksburg, MS 39180
1	ATTN: Research Center Library
	Director, U.S. Army Ballistic Research Laboratory, Aberdeen Proving Ground, MD 21005
1	ATTN: SLCBR-DD-T (STINFO)
1	SLCBR-IV-M, Dr. W. H. Drysdale
1	SLCBR-TB-W, Dr. T. Wright
1	SLCBR-TB-W, Dr. N. J. Huffington
1	SLCBR-TB-W, Dr. J. Walter
	Director, Benet Weapons Laboratory, LCWSL, USA AMCCOM, Watervliet, NY 12189
1	ATTN: AMSMC-LCB-TL
	Commander, U.S. Army Foreign Science and Technology Center, 220 7th Street, N.E., Charlottesville, VA 22901-5396
3	ATTN: AIFRTC, Applied Technologies Branch, Gerald Schlesinger
	Commander, U.S. Army Aviation Systems Command, Aviation Research and Technology Activity, Aviation Applied Technology Directorate, Fort Eustis, VA 23604-5577
1	ATTN: SAVDL-E-MOS
	NASA - Langley Research Center, Hampton, VA 23665
1	ATTN: Aerostructures Directorate
	Naval Research Laboratory, Washington, DC 20375
1	ATTN: Code 5830
	Office of Naval Research, 800 North Quincy Street, Arlington, VA 22217-5000
1	ATTN: Mechanics Division, Code 1132-SM
	Naval Air Development Center, Warminster, PA 18974-5000
1	ATTN: Code 6064
1	AVCSTD/6043

No. of Copies	To
	Commander, David Taylor Naval Ship Research & Development Center, Bethesda, MD 20084
1	ATTN: Code 172
	U.S. Air Force Office of Scientific Research, Bolling Air Force Base, Washington, DC 20332
1	ATTN: Mechanics Division
	Commander, U.S. Air Force Wright Research & Development Center, Wright-Patterson Air Force Base, OH 45433-6523
1	ATTN: WRDC/MLLN, Dr. T. Nicholas
	NASA - Marshall Space Flight Center, Huntsville, AL 35812
1	ATTN: EH01, Dir, M&P Lab
1	Committee on Marine Structures, Marine Board, National Research Council, 2101 Constitution Avenue, N.W., Washington, DC 20418
1	Prof. Elias Aifantis, ME-EM Department, Michigan Technological University, Houghton, MI 49931
1	D. H. Allen, Texas A&M University, Aerospace Engineering Dept., College Station, TX 77843
1	Prof. Lallit Anand, Massachusetts Institute of Technology, Department of Mechanical Engineering, Cambridge, MA 02139
1	Prof. Robert Asaro, Dept. of Applied Mechanics and Engineering Science, R-011, University of California, San Diego, La Jolla, CA 92093
1	Prof. S. N. Atluri, Center for the Advancement of Computational Mechanics, Georgia Institute of Technology, Mail Code 0356, Atlanta, GA 30332
1	Douglas Baumann, Sandia Laboratories, Div. 8123, P.O. Box 969, Livermore, CA 94305
1	Prof. Romesh Batra, University of Missouri at Rolla, Department of Engineering Mechanics Rolla, MO 65401-0249
1	Dr. Steve Bless, University of Dayton Research Institute, Impact Physics - KLA 14, 300 College Park, Dayton, OH 45469
1	Prof. Stuart Brown, Massachusetts Institute of Technology, Department of Materials Science, Cambridge, MA 02139
1	Robert J. Bucci, ALCOA Laboratories, ALCOA Center, Pittsburgh, PA 15069
1	S. T. Buljan, GTE Laboratories, Inc., 40 Sylvan Road, Waltham, MA 02254
1	Prof. B. Budiansky, Harvard University, Pierce Hall, Cambridge, MA 02138
1	Mr. Steven Bullis, GE Armament Systems Department, Burlington, VT 05401-4985
1	Dr. Peter C. T. Chen, Benet Weapons Laboratory, B-115, Watervliet, NY 12189-4050
1	Dr. Shin-Chi Chu, ARDEC, ATTN: SMCAR-SCA-TT, Dover, NJ 07801-5001
1	Prof. R. Clifton, Brown University, Division Of Engineering, Providence, RI 02912
1	Prof. Yannis F. Dafalias, University of California, Department of Civil Engineering, Davis, CA 95616
1	Prof. Paul Dawson, Cornell University, Department of Mechnical and Aerospace Engineering, Ithaca, NY 14853
1	Prof. J. Duffy, Brown University, Division of Engineering, Providence, RI 02912
1	Prof. G. J. Dvorak, Rensselaer Polytechnic Institute, Department of Civil Engineering, Troy, NY 12180-3590
1	Prof L. B. Freund, Brown University, Division of Engineering, Providence, RI 02912
1	Dr. L. M. Gold, Raytheon Company, 528 Boston Post Road, Sudbury, MA 01776
1	Prof. E. Hart, Cornell University, Department of Theoretical and Applied Mechanics, Ithaca, NY 14853
1	Dr. S. S. Hecker, Los Alamos National Laboratories, Los Alamos, NM 87545
1	Dr. Peter Hilton, Arthur D. Little, Inc., 15L/155B Acorn Park, Cambridge, MA 02140

No. of Copies	To
1	Prof. J. W. Hutchinson, Harvard University, Pierce Hall, Cambridge, MA 02138
1	Gordon R. Johnson, Honeywell Defense Systems Division, 5901 South County Road 18, Edina, MN 55436
1	Prof. S. Kobayashi, University of California, Department of Mechaical Engineering, Berkeley, CA 94720
1	Prof. Erhardt Krempl, Rensselaer Polytechnic Institute, Department of Mechanical Engineering, Troy, NY 12180-3590
1	Raymond Krieg, Sandia National Laboratories, Div. 5521, Kirtland Base East, Albuquerque, NM 87185
1	Dr. J. Lankford, Southwest Resarch Institute, 6220 Culebra Road, San Antonio, TX 78284
1	Dr. Ulric Lindholm, Southwest Research Institues, P.O. Drawer 28510, San Antonio, TX 78026
1	Prof. F. A. McClintock, Massachusetts Institute of Technology, Department of Mechanical Engineering, Cambridge, MA 02139
1	R. M. McMeeking, University of California at Santa Barbara, Department of Material Science, Santa Barbara, CA 93111
1	Prof. Alan Miller, Stanford Univerrsity, Department of Materials Science Engineering, Stanford, CA 94035
1	Prof. Alan Needleman, Brown University, Division of Engineering, Providence, RI 02912
1	Prof. Sia Nemat-Nasser, University of California, San Diego, Department of Applied Mechanics and Engineering Sciences, Mail Code B-010, La Jolla, CA 92093
1	Prof. James Rice, Harvard University, Pierce Hall, Cambridge, MA 02138
1	Dr. Owen Richmond, ALCOA Laboratories, ALCOA Center, Pittsburgh, PA 15069
1	Prof. R. O. Ritchie, University of California, Department of Engineering, Berkely, CA 94720
1	Prof. John N. Rossetos, Northeastern University, Boston, MA 02115
1	Prof. John Rudnicki, Northwestern University, Dept. of Civil Engineering, Evanston, IL 60201
1	Dr. L. Seaman, Stanford Research Institute, 333 Ravenswood Avenue, Menlo Park, CA 94025
1	Dr. D. A. Shockey, Stanford Research Institute International, 333 Ravenswood Avenue, Menlo Park, CA 94025
1	Prof. Floyd Tuler, Worcester Polytechnic Institute, Mechanical Engineering Department, 100 Institute Road, Worcester, MA 01609-2280
1	Dr. Kirk Valanis, Endocronics, 8605 Lakecrest Court, Vancouver, WA 98665
	Laboratoire de Physique, et Méchanique des Materiaux, Faculte' des Sceinces, Ile du Saulcy, 574045, Metz Cedex 1, France
1	ATTN: G. R. Canova
1	J. R. Klepaczko
1	A. Molinari
1	Dr. F. Montheillet, Department Materiaux, Ecole de Mines, 158 Cours Fauriel, 42023 Saint-Etienne, France
	Los Alamos National Laboratories, Materials Science Division, Los Alamos, NM 87545
1	ATTN: G765 Dr. U. F. Kocks
1	G770 Dr. A. D. Rollet
1	G730 Dr. Mike Stout
1	G756 Dr. P. S. Follansbee
	Director, U.S. Army Materials Technology Laboratory, Watertown, MA 02172-0001
2	ATTN: SLCMT-TML
2	Authors

U.S. Army Materials Technology Laboratory,
Watertown, Massachusetts 02172-0001
EXPERIMENTAL EVALUATION OF THE TAYLOR-TYPE POLYCRYSTAL MODEL FOR THE FINITE DEFORMATION OF AN FCC METAL (OFHC COPPER) -
Tusit Weerasooriya and Ronald A. Swanson

Technical Report MTL TR 91-20, May 1991, 61 pp - illus-tables; D/A Project: 611102.H420011

AD____________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

Key Words

Polycrystal
Finite deformation
OFHC copper

Large deformation uniaxial compression and fixed-end torsion (simple shear) experiments were conducted on annealed OFHC copper to obtain its stress-strain behavior. This material behavior was also predicted using the Taylor-type rate dependent polycrystal model by Asaro and Needleman. Simulations of the experiments were conducted in personal computers, using a recently developed, highly efficient, fully implicit, time integration scheme by Kalidindi et al. In the initial phase of the simulation, the evolution of the constituent single crystal slip system deformation resistance was estimated using the experimentally determined compressive stress-strain behavior of the polycrystal. With this crystal constitutive behavior, the stress-strain behavior of the polycrystal for simple shear was computed. In addition, the evolution of the crystallographic texture was computed for both compression and simple shear tests. Both the shear and axial stress-strain behaviors for the simple shear test compared reasonably well with the experimental results.

U.S. Army Materials Technology Laboratory,
Watertown, Massachusetts 02172-0001
EXPERIMENTAL EVALUATION OF THE TAYLOR-TYPE POLYCRYSTAL MODEL FOR THE FINITE DEFORMATION OF AN FCC METAL (OFHC COPPER) -
Tusit Weerasooriya and Ronald A. Swanson

Technical Report MTL TR 91-20, May 1991, 61 pp - illus-tables; D/A Project: 611102.H420011

AD____________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

Key Words

Polycrystal
Finite deformation
OFHC copper

Large deformation uniaxial compression and fixed-end torsion (simple shear) experiments were conducted on annealed OFHC copper to obtain its stress-strain behavior. This material behavior was also predicted using the Taylor-type rate dependent polycrystal model by Asaro and Needleman. Simulations of the experiments were conducted in personal computers, using a recently developed, highly efficient, fully implicit, time integration scheme by Kalidindi et al. In the initial phase of the simulation, the evolution of the constituent single crystal slip system deformation resistance was estimated using the experimentally determined compressive stress-strain behavior of the polycrystal. With this crystal constitutive behavior, the stress-strain behavior of the polycrystal for simple shear was computed. In addition, the evolution of the crystallographic texture was computed for both compression and simple shear tests. Both the shear and axial stress-strain behaviors for the simple shear test compared reasonably well with the experimental results.

U.S. Army Materials Technology Laboratory,
Watertown, Massachusetts 02172-0001
EXPERIMENTAL EVALUATION OF THE TAYLOR-TYPE POLYCRYSTAL MODEL FOR THE FINITE DEFORMATION OF AN FCC METAL (OFHC COPPER) -
Tusit Weerasooriya and Ronald A. Swanson

Technical Report MTL TR 91-20, May 1991, 61 pp - illus-tables; D/A Project: 611102.H420011

AD____________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

Key Words

Polycrystal
Finite deformation
OFHC copper

Large deformation uniaxial compression and fixed-end torsion (simple shear) experiments were conducted on annealed OFHC copper to obtain its stress-strain behavior. This material behavior was also predicted using the Taylor-type rate dependent polycrystal model by Asaro and Needleman. Simulations of the experiments were conducted in personal computers, using a recently developed, highly efficient, fully implicit, time integration scheme by Kalidindi et al. In the initial phase of the simulation, the evolution of the constituent single crystal slip system deformation resistance was estimated using the experimentally determined compressive stress-strain behavior of the polycrystal. With this crystal constitutive behavior, the stress-strain behavior of the polycrystal for simple shear was computed. In addition, the evolution of the crystallographic texture was computed for both compression and simple shear tests. Both the shear and axial stress-strain behaviors for the simple shear test compared reasonably well with the experimental results.

U.S. Army Materials Technology Laboratory,
Watertown, Massachusetts 02172-0001
EXPERIMENTAL EVALUATION OF THE TAYLOR-TYPE POLYCRYSTAL MODEL FOR THE FINITE DEFORMATION OF AN FCC METAL (OFHC COPPER) -
Tusit Weerasooriya and Ronald A. Swanson

Technical Report MTL TR 91-20, May 1991, 61 pp - illus-tables; D/A Project: 611102.H420011

AD____________

UNCLASSIFIED
UNLIMITED DISTRIBUTION

Key Words

Polycrystal
Finite deformation
OFHC copper

Large deformation uniaxial compression and fixed-end torsion (simple shear) experiments were conducted on annealed OFHC copper to obtain its stress-strain behavior. This material behavior was also predicted using the Taylor-type rate dependent polycrystal model by Asaro and Needleman. Simulations of the experiments were conducted in personal computers, using a recently developed, highly efficient, fully implicit, time integration scheme by Kalidindi et al. In the initial phase of the simulation, the evolution of the constituent single crystal slip system deformation resistance was estimated using the experimentally determined compressive stress-strain behavior of the polycrystal. With this crystal constitutive behavior, the stress-strain behavior of the polycrystal for simple shear was computed. In addition, the evolution of the crystallographic texture was computed for both compression and simple shear tests. Both the shear and axial stress-strain behaviors for the simple shear test compared reasonably well with the experimental results.