LA-UR-12-24915
Approved for public release; distribution is unlimited.

Title: A semi-implicit integration scheme for the stress update of viscoSCRAM

Author(s): Buechler, Miles A.
Luscher, Darby J.

Intended for: Report

A semi-implicit integration scheme for the stress update of viscoSCRAM

Miles A. Buechler[1] and Darby J. Luscher[2]

Los Alamos National Laboratory

Los Alamos, NM 87545

[1]buechler@lanl.gov [2]djl@lanl.gov

September 20, 2012

Abstract

This report documents three improvements made to the implementation of a historically significant material model used in W13 for simulations involving components made of PBX9501. ViscoSCRAM combines linear viscoelasticity with isotropic damage evolution based on fracture mechanics concepts. The original implementation was focused on short duration transient events, thus an explicit update scheme was used. For longer duration simulations which employ relatively larger time step sizes the explicit update scheme is inadequate. This work presents a new semi-implicit update scheme suitable for simulations using relatively large time steps. The algorithm solves a nonlinear system of equations to ensure that the stress, damaged state, and internal (Maxwell) stresses are in agreement with implicit update equations at the end of each increment. The crack growth is advanced in time using a subincremental explicit scheme, thus the entire implementation is semi-implicit. Additionally, modifications to the crack growth kinetics are proposed and implemented obviating the need for an empirical tensile damage growth rate (TDGR) multiplier used in previous implementations. Finally, an automatic time stepping algorithm is implemented to adjust the time step size based on error estimates for the generalized Maxwell model update scheme. Model parameters are fit to existing data and example problems are simulated using both new and legacy implementations to assess comparative advantages and disadvantages.

1 Introduction

This report documents a new implementation of an historically significant constitutive model frequently used to represent the plastic bonded explosive PBX-9501 in finite element simulations of thermomechanical response. PBX-9501 is a particulate composite nominally consisting of 95% HMX crystals bonded together by 2.5% elastomer binder (estane) and 2.5% nitroplasticizer. Mechanical deformation of this material is primarily accommodated by straining of the polymer binder and rearrangement of relatively rigid HMX crystals. Consequently, early work to describe the constitutive behavior of the composite focused on phenomenological linear viscoelastic models known to reasonably characterize many polymers. However, the material stress strain behavior departs from a purely viscoelastic response and this difference was largely attributed to brittle damage of the material realized by the nucleation and growth of microscopic cracks.

ViscoSCRAM is a phenomenological constitutive model developed to represent the combined viscoelastic and brittle damage response of PBX-9501. While this constitutive model has some utility in representing the rate and temperature dependent nature over time scales covering short duration impact events to long term creep behavior, there are several limitations both in the theoretical aspects of the model as well as its early implementation. Never-the-less, viscoSCRAM has been used extensively in many engineering assessments and will continue to be used until the next generation of constitutive models is developed to overcome limitations of the *theory*. In the meantime, it is prudent to address limitations associated with the implementation of the theory so that the model is usable over the range of intended applications. In a colloquial sense, the work detailed in this report is intended to "shore up" previous existing versions of viscoSCRAM. The work does not propose a new theory to overcome model limitations, but does offer a few small enhancements over the original theory.

Originally, viscoSCRAM was developed to model stress strain response over strain rates ranging from approximately 100 to 10,000 s^{-1} for short duration events resulting in mechanical impact. Numerical codes used to solve this class of problems are typically Lagrange finite element codes (e.g. Abaqus Explicit, EPIC) employing explicit time integration of the equations of motion. Stability requirements associated with explicit

time integration of the global equations of motion dictate the use of relatively small time steps. While the damage evolution equations within viscoSCRAM are nonlinear and tightly coupled with the viscoelastic response, the small time steps required for global stability ensure reasonable accuracy of the constitutive update employing explicit (forward Euler) integration of these coupled differential equations. However, viscoSCRAM has also been adapted for modeling the coupled long term creep and damage evolution of PBX-9501, for example under storage conditions prior to a transient mechanical insult. Such simulations typically employ implicit integration over long simulated time periods (e.g. 20 years) prior to handing off to an explicit code for assessing the transient impact event. Simulating such long duration events necessitates relatively large time steps, thus explicit integration within the constitutive update introduces unacceptable error. A fully implicit constitutive update would require solving the strongly nonlinear system of equations and, while possible, would present an unnecessary computational burden. Here, a semi-implicit constitutive update scheme is presented as an alternative strategy which treats the strain decomposition and viscoelastic update in a fully implicit manner over the full time step, but integrates the damage evolution using a higher ordered explicit method over several subincrements of the full step.

The report is organized as follows. Section 2 gives a brief history of viscoSCRAM theory and lays out the constitutive equations. A new modification of the internal frictional dependence of the crack growth kinetics is discussed in Section 3. Section 4 discusses one common constitutive update algorithm and its limitations. A new semi-implicit constitutive update scheme developed in this work is presented in Section 5. Results from example simulations used to compare performance of the new implementation with the previous implementation are presented in Section 7.

For clarity, indicial notation is frequently used in this report. In some cases, the presentation is made cleaner by using direct notation. For example, when dealing with multiple solution iterations for discrete time increments using separate Maxwell elements, it is more clear to represent a second order tensor in direct notation and reserve sub- and superscripts to distinguish the increment, iterate, and element of concern. In direct notation a tensor is represented in bold, e.g., $\mathbf{A}$, while in indicial notation the coefficients comprising the tensor are indicated, e.g., A_{ij}. The tensor order is made clear upon first introduction when using direct

notation, while tensoral order is clear from the number of indices in indicial notation. Standard summation convention applies, i.e., summation over three Cartesian indices is implied by repeated indices. For example, $\texttt{Trace}\,[\mathbf{A}] = A_{kk} = A_{11} + A_{22} + A_{33}$. The 4th order identity tensor is represented in direct notation by $\mathbb{I}$ and indicial notation as $\frac{1}{2}\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right)$. The second order identity tensor is represented in direct notation as $\mathbf{i}$ and in indicial notation by coefficients of the Dirac delta, $\delta_{ij} = 1$ for $i = j$ and 0 otherwise. The scalar product between two second order tensors is denoted by the operator :, i.e., $\mathbf{A} : \mathbf{B} = A_{ij}B_{ij}$.

2 ViscoSCRAM: Brief history and theory

A brief history of this model is discussed here in order to draw attention to the original intentions for the theory, as well as, features that have been omitted and could be reintroduced to overcome some of the limitations of viscoSCRAM. The original model formulation is based on the concept that rate dependence of PBX-9501 is attributable to viscoelasticity of the binder and the rate process of microcracking damage manifest as degradation of stress carrying ability of the composite. Viscoelasticity of the binder is represented by a generalized Maxwell model and the breakage of bonds both within the binder and between binder and particulates is modeled using a statistical representation of cracks evolving according to brittle fracture mechanics. The model for statistical crack mechanics (SCRAM) used within viscoSCRAM was originally developed by Dienes (1985) for application to dynamic problems involving brittle crack growth and coalescence, in particular blasting of oil shale and the sensitivity of propellants to impact. The kinematic basis was a decomposition of the relevant strain rates to reflect deformation attributable to separate mechanisms such as, for example, elasticity, deformation due to opening of existing cracks, shearing of closed cracks, and subsequent growth of cracks. Their work assumed nine possible crack orientations in a two-dimensional plane and an exponential distribution of crack radius associated with each orientation separately. Thus the original model provided for a large degree of initial and evolving anisotropy. It should be noted that the theory is based on an assumption of non-interacting cracks, an assumption that the original author thought to be a limitation in applications to compressive failure. For a more recent review of the evolution of SCRAM, its application

to high-explosive materials, and extensions to address such limitations of the early theory cf. Dienes et al. (2006).

Addessio and Johnson (1990) simplified the original SCRAM theory into an isotropic theory (ISO-SCRAM) for application to high-velocity impact of ceramic armor where the transition from initial state to fully comminuted material is rapid enough that the details of evolved anisotropy do not influence the process. This argument is essentially one of proportional monotonic loading. Their work averages the contribution of a system of cracks whose orientation distribution is isotropic and size distribution is exponential. Thus the microscopic damage state can be defined by a single parameter, $\bar{c}$, the average crack radius. Addessio and Johnson (1990) also make use of a damage threshold hypersurface in the space of stress and average crack radius. This surface is analogous to a yield surface and the crack growth rate is a nonlinear function of the distance the current material state lies beyond the damage surface and the limiting crack velocity defined by the materials Raleigh wave speed.

Based on elements of the original SCRAM theory and ISO-SCRAM, Bennett et al. (1998) and Hackett and Bennett (2000) developed viscoSCRAM by treating the "nominally undamaged" bulk material as viscoelastic rather than linearly elastic. Like ISO-SCRAM, viscoSCRAM is fully isotropic; however, viscoSCRAM uses crack growth kinetics more similar to SCRAM (Dienes et al., 2006) than the damage surface based kinetics of Addessio and Johnson (1990). It should be noted that viscoSCRAM does not include mechanisms for shear dilatancy. Shear dilatancy can result from opening of cracks with certain orientations under shear, or from joint opening caused by particle interaction. Joint or crack opening as a source of dilatancy has been incorporated into more recent versions of SCRAM (Dienes et al., 2006).

Kinematics

Small strains, ϵ_{ij} are assumed and the total strain tensor is decomposed into volumetric and deviatoric parts according to

$$\epsilon_{ij} = e_{ij} + \frac{1}{3}\epsilon_{kk}\delta_{ij} \tag{1}$$

where e_{ij} is the deviatoric strain tensor and the trace of total strain tensor, ϵ_{kk}, is the volumetric strain. The deviatoric strain is additively decomposed into viscoelastic, e_{ij}^{ve}, and damage e_{ij}^{D} parts according to

$$e_{ij} = e_{ij}^{ve} + e_{ij}^{D} \tag{2}$$

where e^{ve} is the viscoelastic component of the deviatoric strain and e^{D} is the portion of the deviatoric strain due to additional compliance introduced through damage. Thus, viscoSCRAM represents a deviatoric/volumetric decoupling whose deviatoric response reflects the contributions of viscoelasticity and isotropic damage.

Stress strain relations

The Cauchy stress tensor, σ_{ij}, is decomposed into deviatoric and hydrostatic parts, i.e.,

$$\sigma_{ij} = S_{ij} + \delta_{ij}\sigma^{m} \tag{3}$$

where S_{ij} is the deviatoric stress tensor, and $\sigma^{m} = \sigma_{kk}/3$ is the mean stress. The mean stress is related to the volumetric strain in accordance with a linear Hookean bulk response, i.e.,

$$\sigma^{m} = K\epsilon_{kk} \tag{4}$$

although (1) some implementations favor using a separate viscoelastic model for the bulk response, and (2) it is conceivable one may replace this relationship with a more general pressure, specific volume, temperature equation of state. The deviatoric stress directly depends solely upon the viscoelastic part (of the deviatoric part) of strain in accordance with a generalized Maxwell model according to

$$\dot{S}_{ij} = 2G^{\infty}\dot{e}_{ij}^{ve} + \sum_{\kappa=1}^{N}\left(2G^{(\kappa)}\dot{e}_{ij}^{ve} - \frac{S_{ij}^{(\kappa)}}{\tau^{(\kappa)}}\right) \tag{5}$$

where G^{∞} is the stead-state shear modulus, $G^{(\kappa)}$, $\tau^{(\kappa)}$, $S_{ij}^{(\kappa)}$ are the shear modulus, relaxation time, and deviatoric stresses associated with the κ^{th} (of N) Maxwell element and Equation 5 is derived in Appendix C. The damage strain is related to the deviatoric stress through the damage compliance relationship

$$e_{ij}^{D} = \frac{1}{2G_0}\left(\frac{c}{a}\right)^{3} S_{ij} \tag{6}$$

where the instantaneous shear modulus $G_0 = G^\infty + \sum_{\kappa=1}^{N} G^{(\kappa)}$, c is a damage parameter representing the mean microstructural crack size (in units of length) in a phenomenological sense, and a is a normalizing parameter (Addessio and Johnson, 1990).

Kinetics of damage growth

Crack growth is based on the phenomenology of brittle fracture mechanics. There are two distinct regions of crack growth rate, i.e., stable and unstable growth (cf. Dienes et al., 2006). Accordingly, the crack growth rate depends upon an effective stress intensity (Hackett and Bennett, 2000; Bennett et al., 1998; Dienes et al., 2006; Dienes, 1998),

$$K_I = \bar{\sigma}\sqrt{\pi c} \tag{7}$$

where the effective stress, $\bar{\sigma}$, is computed as

$$\bar{\sigma} = \begin{cases} \left(\frac{3}{2} S_{ij} S_{ij}\right)^{\frac{1}{2}}, & \text{for } \sigma_m < 0 \quad (8a) \\ \left(\frac{3}{2} \sigma_{ij} \sigma_{ij}\right)^{\frac{1}{2}}, & \text{otherwise} \quad (8b) \end{cases}$$

The crack growth rate is

$$\dot{c} = \begin{cases} v_{res} \left(\frac{K_I}{K_1}\right)^m, & \text{for } K_I < K' \quad (9a) \\ v_{res} \left[1 - \left(\frac{K_{0\mu}}{K_I}\right)^2\right], & \text{otherwise} \quad (9b) \end{cases}$$

where $v_{res} = v_{max} f(\dot{\bar{e}}, \sigma_m)$ is an effective maximum crack velocity depending upon the theoretical terminal crack speed, v_{max}, (in principle the Raleigh wave speed) and an empirical "knock-down" function of effective strain rate and mean stress, $f(\dot{\bar{e}}, \sigma_m)$, and m is a model parameter. The two distinct regimes of crack growth correspond to stable crack growth (Equation 9a) and unstable growth (Equation 9b), respectively. The stress intensity at which the behavior transitions from Equation 9a to Equation 9b is defined as

$$K' = K_{0\mu} \left(1 + \frac{2}{m}\right)^{1/2} \tag{10}$$

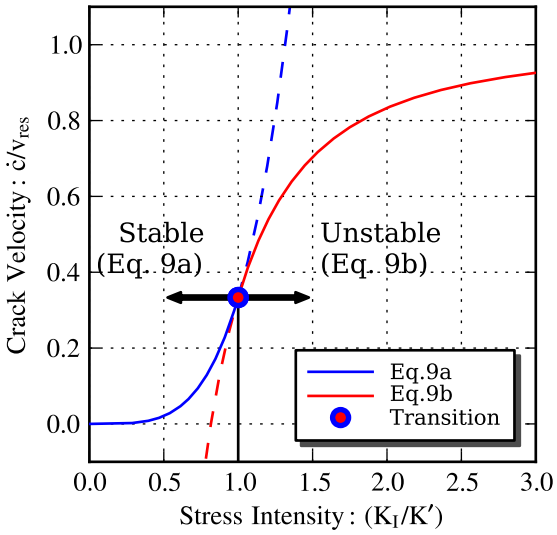


Figure 1: Relationship between normalized crack velocity $\dot{c}/v_{res}$ and normalized stress intensity K_I/K' illustrating transition between stable and unstable crack growth.

where $K_{0\mu}$ is the frictional threshold stress intensity which increases as a function of mean stress σ_m to reflect a frictional resistance to crack growth. The frictional threshold stress intensity is computed by

$$K_{0\mu} = K_0 \left[1 - \frac{\pi\mu'\sigma_m\sqrt{c}}{K_0}\left(1 - \frac{\mu'\sigma_m\sqrt{c}}{K_0}\right)\right]^{\frac{1}{2}} \tag{11}$$

where μ' is a friction coefficient and K_0 is a model parameter representing the non-frictional threshold stress intensity of the material. The normalizing stress intensity in the denominator of Equation 9a is defined as

$$K_1 = K'\left(1 + \frac{m}{2}\right)^{1/m} \tag{12}$$

Figure 1 illustrates the transition from stable to unstable crack growth kinetics based on Equations 9a and 9b. Dependence of the transition threshold stress intensity, K' on the current mean crack size and mean stress present numerical difficulties for implementing a fully implicit update scheme. As discussed in Section 5, these difficulties motivate the semi-implicit scheme presented in this report.

Note that, according to Hackett and Bennett (2000), the effective frictional coefficient, μ', is related to the

internal static coefficient of friction, μ_s, by

$$\mu' = \left[\frac{45}{2\left(3 - 2\mu_s^2\right)}\right]^{\frac{1}{2}} \mu_s \tag{13}$$

however, given the level of empiricism involved with determining μ_s, it is justifiable to instead specify μ' directly.

As stated above, the purpose of Equation 11 is to retard crack growth under compression; however, as discussed in detail in Section 3, this frictional dependence also retards crack growth rate under tension. To overcome this, previous implementations used a tensile damage growth rate scaling factor $C_{TDGR} \gg 1$ to *increase* v_{res} for tensile mean stress σ_m. This issue is addressed by the proposed modifications to the crack growth kinetics presented in Section 3.

Finally, we note a few typographical errors appearing in (Hackett and Bennett, 2000) with the intention of clarifying differences.

- K' erroneously appears in place of K_1 in Equation 9a
- K_1 erroneously appears in place of K_I in Equation 9b
- The power $\frac{1}{2}$ was erroneously left off in Equation 13

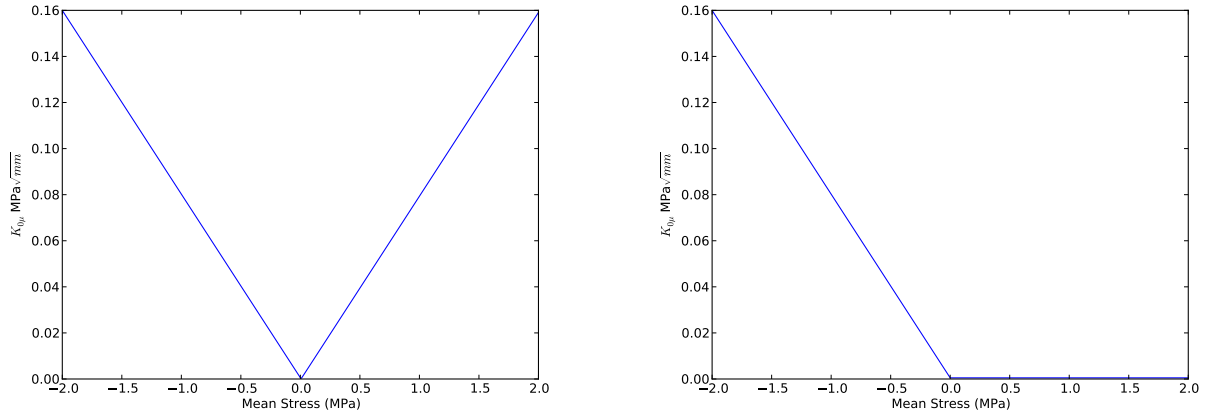
Table 1: Summary of model parameters

Variable	Description	Units
K	Elastic bulk modulus	FL^{-2}
G^{∞}	Long term elastic shear modulus	FL^{-2}
G^{κ}	Viscoelastic shear moduli	FL^{-2}
τ^{κ}	Relaxation time constants	FL^{-2}
c_0	Initial value for mean crack size, c	L
a	Crack normalizing parameter	L
K_0	Threshold stress intensity	$FL^{-3/2}$
μ'	Material frictional parameter	-
m	Crack growth kinetics shape parameter	-
v_{max}	Maximum crack speed	Lt^{-1}

3 Update to crack growth kinetics

In this section, a new modification of the damage evolution law from the original viscoSCRAM theory is presented. As discussed in Section 2 , the effective threshold stress intensity ($K_{0\mu}$) depends on mean stress as a way of retarding the damage evolution rate under compressive loading (cf. Equation 11). Such behavior is physically consistent with the concept of an internal frictional resistance to damage evolution under compression (Bennett et al., 1998; Hackett and Bennett, 2000) and is consistent with experimental data. However, the frictional dependence is symmetric about zero mean stress as shown in Figure 2a. Frictional interaction under tensile loading is inconsistent with the phenomenology leading to Equation 11.

A simple modification to overcome this issue is proposed, i.e., by replacing σ_m with $- < p >$ where the thermodynamic pressure $p = -\sigma_m$ and $< \bullet >$ are Macaulay brackets that return the argument for values



(a) Original formulation for $K_{0\mu}$

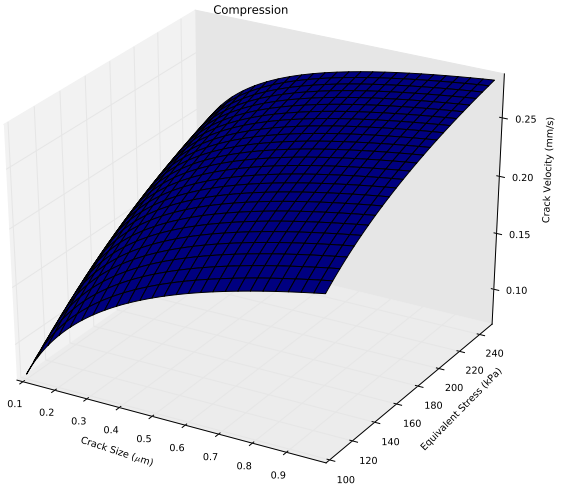
(b) New formulation for $K_{0\mu}$

Figure 2: $K_{0\mu}$ without and with the modifications

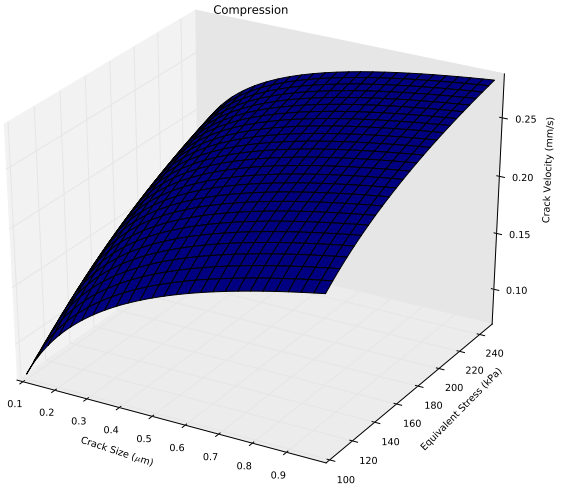
greater than zero, otherwise zero. Thus,

$$K_{0\mu} = K_0 \left[1 + \frac{\pi\mu' < p > \sqrt{c}}{K_0} \left(1 + \frac{\mu' < p > \sqrt{c}}{K_0}\right)\right]^{\frac{1}{2}} \tag{14}$$

The modified $K_{0\mu}$ increases under increasing compressive mean stress and remains equal to K_0 for tensile mean stress. This change in formulation for $K_{0\mu}$ manifests as a more physical damage growth behavior in simulations. To demonstrate this, we simulated uniaxial compression and tension for a simple test problem. States of uniaxial stress are simulated such that the equivalent stress is the absolute value of the applied uniaxial stress, and the mean stress is 1/3 the applied stress. Figure 3 compares the simulated damage growth rate as a function of equivalent stress and crack size under compression using (a) Equation 11 to that using (b) Equation 14. These results show that the proposed change to $K_{0\mu}$ has no impact on the damage growth rate when the mean stress is negative.

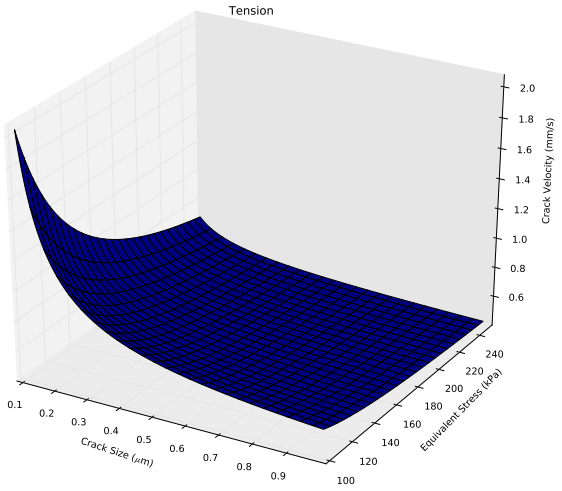


(a) Original formulation for $\dot{c}$

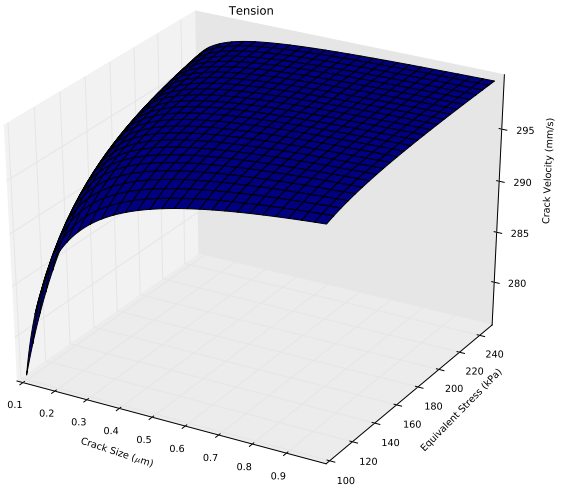


(b) New formulation for $\dot{c}$

Figure 3: Comparison of simulated damage growth rate $\dot{c}$ for uniaxial compression using old (a) versus new (b) kinetics. The results are identical for compressive mean stress.



(a) Original formulation for $\dot{c}$



(b) New formulation for $\dot{c}$

Figure 4: Comparison of simulated damage growth rate $\dot{c}$ for uniaxial tension using old (a) versus new (b) kinetics. Case (b) reflects more physical behavior.

On the other hand, Figure 4 shows the damage growth rate as a function of equivalent stress and crack size under tension. There clearly is a difference in the damage growth rate under tensile mean stress. Using the old

formulation the damage growth rate gets *smaller* with *increasing* damage and *increasing* equivalent stress. This aphysical behavior is caused by the increase in $K_{0\mu}$ for increasing tensile mean stress. Clearly, such behavior is inconsistent with the phenomenology of internal frictional resistance to damage presented by Bennett et al. (1998); Hackett and Bennett (2000). The new formulation is consistent with such phenomenology in that damage grows *faster* for *larger* crack sizes and *larger* equivalent stress. The sensitivity to these variables may be too large in this simulation as indicated by the early saturation of the crack velocity in Figure 4b, which probably indicates an inappropriate selection of model parameters such as the damage growth rate exponent m.

This change in the damage evolution law provides a more physical behavior under uniaxial loading than the previous implementation which relied on a tension/compression switch which abruptly scaled the "knock down" behavior of $f(\dot{\bar{e}})$ by a constant factor $C_{TDGR} \gg 1$ to provide a higher growth rate under tension. Note that the damage parameter is isotropic and grows isotropically, so while this modification is clearly an improvement for pure uniaxial loading, it is not so clear that the formulation is ideal in other more complicated load cases where the mean stress may be positive but there could be one or two small negative principal stresses. This modification requires a new method for parameter estimation. In the past, the damage evolution parameters were fit to compressive data and then C_{TDGR} was determined to best fit the tensile data. We now suggest that the damage parameters be fit to tensile data which is insensitive to μ'. Then, μ' can be fit using compressive data.

4 Historical Implementation

Prior to discussing a few modifications made to the theory of Bennett et al. (1998); Hackett and Bennett (2000) and the new semi-implicit intregration scheme, this section outlines the update scheme typically used. In particular, we address the implementation as discussed in Hackett and Bennett (2000). Differentiating 6 with respect to time results in

$$\dot{\boldsymbol{e}}^D = \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \dot{\boldsymbol{S}} + \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \dot{c}\boldsymbol{S} \tag{15}$$

Substituting the time derivative of Equation 2 along with Equation 15 into Equation 5 gives

$$\dot{\boldsymbol{S}} = 2G_0\dot{\boldsymbol{e}} - 3\left(\frac{c}{a}\right)^2\frac{\dot{c}}{a}\boldsymbol{S} - \left(\frac{c}{a}\right)^3\dot{\boldsymbol{S}} - \sum_{\kappa=1}^{N}\left(\frac{\boldsymbol{S}^{(\kappa)}}{\tau^{(\kappa)}}\right) \tag{16}$$

which is rearranged as

$$\dot{\boldsymbol{S}} = \psi(c)\dot{\boldsymbol{e}} - \theta(c,\dot{c})\left[\boldsymbol{S} - \lambda_{ij}\left(c, \boldsymbol{S}^{(\kappa)}\right)\right] \tag{17}$$

where the *state-dependent* scalar-valued functions $\psi(c)$ and $\theta(c,\dot{c})$ have been introduced and are defined as

$$\psi(c) = \frac{2G_0}{1+\left(\frac{c}{a}\right)^3} \tag{18}$$

and

$$\theta(c,\dot{c}) = \frac{3\left(\frac{c}{a}\right)^2\frac{\dot{c}}{a}}{1+\left(\frac{c}{a}\right)^3} \tag{19}$$

respectively. Additionally, the *state-dependent* tensor-valued function, λ_{ij}, is defined as

$$\lambda_{ij}\left(c, \boldsymbol{S}^{(\kappa)}\right) = \frac{\sum_{\kappa=1}^{N}\left(\frac{\boldsymbol{S}^{(\kappa)}}{\tau^{(\kappa)}}\right)}{3\left(\frac{c}{a}\right)^2\frac{\dot{c}}{a}} \tag{20}$$

We emphasize that the state-dependence of the functions appearing in Equation 17 cause it to be a nonlinear first-order differential equation because the non-constant coefficients depend upon the solution. Hackett and Bennett (2000) propose a central differencing scheme to integrate the differential equation; however, the state-dependence of the coefficient functions in Equation 17 are not handled in a fully consistent manner.

Let t_n be the time at the end of the previous (n^{th}) increment and $t_{n+1} = t_n + \Delta t$ be the time at the end of the current increment, thus defining the time step size, Δt. Likewise, a value of some state variable, A, is labeled with subscript "n" or "$n+1$" if it corresponds to the solution at the end of the previous or current increment respectively. Thus, a second-order approximation to $\dot{A}$ at the *middle* of the current time increment is given by the centered finite divided difference

$$\dot{A}_{n+\frac{1}{2}} = \frac{A_{n+1} - A_n}{\Delta t} \tag{21}$$

Making use of Equation 21 enables us to write a discrete form of Equation 17 evaluated at the *middle* of the current increment, i.e.,

$$\theta_{n+\frac{1}{2}}\mathbf{S}_{n+\frac{1}{2}} + \frac{\mathbf{S}_{n+1} - \mathbf{S}_n}{\Delta t} = \psi_{n+\frac{1}{2}}\frac{\mathbf{e}_{n+1} - \mathbf{e}_n}{\Delta t} - \theta_{n+\frac{1}{2}}\boldsymbol{\lambda}_{n+\frac{1}{2}} \tag{22}$$

Hackett and Bennett (2000) make the reasonable approximation,

$$\mathbf{S}_{n+\frac{1}{2}} \approx \tfrac{1}{2}\left(\mathbf{S}_{n+1} + \mathbf{S}_n\right) \tag{23}$$

Using this approximation and denoting the value of variables at the middle of the current increment by $A_{n+\frac{1}{2}} = A_\star$ allows Equation 22 to be solved for the updated stresses, i.e.,

$$\mathbf{S}_{n+1} = \mathbf{S}_n + \frac{\psi_\star}{\left(1 + \frac{\Delta t}{2}\theta_\star\right)}\Delta\mathbf{e} - \frac{\theta_\star\,\Delta t}{\left(1 + \frac{\Delta t}{2}\theta_\star\right)}\left(\boldsymbol{\lambda}_\star + \mathbf{S}_n\right) \tag{24}$$

where $\Delta\mathbf{e} = \mathbf{e}_{n+1} - \mathbf{e}_n$. Equation 24 represents a second-order accurate central difference update scheme for the stresses. As written, this is an *implicit* update scheme which requires solution of a nonlinear system of equations because $\theta_\star$,$\psi_\star$, and $\boldsymbol{\lambda}_\star$ depend upon the state variables, c, $\dot{c}$, and $\mathbf{S}^{(\kappa)}$ evaluated at $t = t_n + \frac{1}{2}\Delta t$ and whose evolution depends upon $\mathbf{S}_{n+1}$. Hackett and Bennett (2000) do not implement the implicit solution of Equation 24; instead, their implementation makes the approximations

$$\boldsymbol{\lambda}_\star = \boldsymbol{\lambda}_n \qquad \psi_\star = \psi_n \qquad \theta_\star = \theta_n \tag{25}$$

so that Equation 24 becomes the purely *explicit* update

$$\mathbf{S}_{n+1} = \mathbf{S}_n + \frac{\psi_n}{\left(1 + \frac{\Delta t}{2}\theta_n\right)}\Delta\mathbf{e} - \frac{\theta_n\,\Delta t}{\left(1 + \frac{\Delta t}{2}\theta_n\right)}\left(\boldsymbol{\lambda}_n + \mathbf{S}_n\right) \tag{26}$$

Updates to the mean crack size are provided by *explicit* time integration of Equation 9a (or Equation 9b), i.e.,

$$c_{n+1} = c_n + \dot{c}_n\,\Delta t \tag{27}$$

The time rate of change of stress in each maxwell element is given by combining the time derivative of Equation 2 with Equation 15, and the argument of the summation in Equation 5 resulting in

$$\dot{\boldsymbol{S}}^{(\kappa)} = 2G^{(\kappa)}\dot{\boldsymbol{e}} - \frac{\boldsymbol{S}^{(\kappa)}}{\tau^{(\kappa)}} - \frac{G^{(\kappa)}}{G^0}\left[\frac{3}{a}\left(\frac{c}{a}\right)^2\dot{c}\boldsymbol{S} + \left(\frac{c}{a}\right)^3\dot{\boldsymbol{S}}\right] \tag{28}$$

Hackett and Bennett (2000) mention that they integrated Equation 28 in a variety of methods, although the legacy implementation uses

$$\Delta\mathbf{S}^{(\kappa)} = 2G^{(\kappa)}\Delta\mathbf{e} - \frac{\mathbf{S}_n^{(\kappa)}}{\tau^{(\kappa)}} - \frac{G^{(\kappa)}}{G_0}\left[\frac{3}{a}\left(\frac{c_{n+1}}{a}\right)^2\Delta c\,\mathbf{S}_{n+1} + \left(\frac{c_{n+1}}{a}\right)^3\left(\mathbf{S}_{n+1} - \mathbf{S}_n\right)\right] \tag{29}$$

5 Implementation of a semi-implicit integration scheme

Updating the state from the end of a previous time increment to the end of the current time increment is the central objective of any material model implementation. Because one independent variable of material state evolution is usually time, the process is often referred to as time integration of the constitutive behavior. Ambiguity in context associated with the words implicit and explicit often lead to and perpetuate misconceptions about the role and significance of various schemes in finite element codes. It is the role of the finite element solver to solve the global equations of motion. That task essentially consists of integrating a set of differential equations over the independent variable, time. The manner in which the global equations of motion are integrated forward in time has such impact on certain characteristics of the solution that general finite element solvers are classified according to whether these equations are solved using *implicit* or *explicit* integration in time, e.g. Abaqus/Standard or Abaqus/Explicit, respectively. Within the overall finite-element method, constitutive models are responsible for determining the stress at particular locations within the continuum associated with the deformation history of that location (and possibly others). We restrict attention here to *local* material descriptions, thus the constitutive update can be described in a general form as

$$(\mathbf{F}(\tau)\,\forall\tau \in [0,t]) \longrightarrow \boxed{\text{Constitutive Update}} \longrightarrow \boldsymbol{\sigma}(t) \tag{30}$$

where $\mathbf{F}$ is the deformation gradient, τ is a time variable spanning the entire history of the continuum location. Constitutive models are not typically formulated in such a general manner; instead, one common approach is to represent the essential aspects of the material's deformation history using a set of state variables, $\boldsymbol{\xi}^{(j)}$, to augment the description of material state. Accordingly, the material model then has the task of both computing the stresses and updating the material state from one increment to another. This process is outlined conceptually here, i.e.,

$$\left(\mathbf{F}(t_{n+1}), \mathbf{F}(t_n), \boldsymbol{\xi}^{(j)}(t_n)\right) \longrightarrow \boxed{\text{Constitutive Update}} \longrightarrow \left(\boldsymbol{\sigma}(t_{n+1}), \boldsymbol{\xi}^{(j)}(t_{n+1})\right) \tag{31}$$

although this is not the most general thermodynamically consistent form as one could include rates of state variables and thermodynamically conjugate forces in the arguments of the update scheme. Often the evolution

kinetics for state variables, $\boldsymbol{\xi}^{(j)}$, are prescribed via a set of differential equations. Thus, integration in time of the constitutive differential equations required in Equation 31 is distinct from that of the global equations of motion, the latter providing the classification of the FE code as *Implicit* or *Explicit*. The point is that one may implement an explicit, fully-implicit, or, as presented here, a semi-implicit scheme for updating the constitutive behavior, regardless of the manner in which the global equations of motion are solved.

There are, however, advantages and disadvantages associated with different constitutive integration techniques and these trade-offs are influenced by the size of the time step demanded by the global equation solver and class of problem. Generally, an explicit constitutive update is much faster, but demands smaller time steps to maintain fidelity to the actual constitutive description. Thus, explicit update routines are typically favored for explicit finite element codes, e.g. Abaqus/Explicit. In some cases, a dynamic problem may be solved with an implicit finite element code and the requisite time step size is relatively small in order to capture the transient nature of the problem. In such cases, an explicit constitutive update may provide adequate accuracy and, consequently, would be favored. On the other hand, in problems where the finite element code can make use of a large time step size, it is advantageous to use an implicit update of the constitutive equations to minimize the error propagated through time.

This section details one of the largest contributions of the work in this report, namely, the implementation of a new constitutive update scheme. The new constitutive update scheme treats the decomposition of strain (Equation 2), viscoelastic stress update (Equation 5), and the final damaged state in a implicit manner (Equation 6) . The resulting system of equations is solved using a newton iteration algorithm. The crack growth kinetics proved numerically difficult to solve in a reliable and robust manner within a fully-implicit scheme, thus the update of the crack length is treated explicitly, although we use a higher order scheme to integrate the crack growth kinetics over several subincrements. Accordingly, the total update scheme is semi-implicit, i.e. implicit in the strain decomposition, stress, and damaged state, but explicit in crack growth rate.

Semi-implicit scheme for viscoSCRAM constitutive update

The main constitutive equations consist of the decomposition of strain, Equation 2, the relationship between stress and viscoelastic strain, Equation 5, the relationship between stress, average crack length, and damage strain, Equation 6, and the kinetics governing crack size evolution, Equations 9a and 9b. We first focus attention on the discrete form of Equations 2, 5, and 6. In this section we use the latin subscript n (or $n+1$) to denote the value of a variable from the converged solution at the end of the previous (or current) time increment. Thus, A_n is the value of A for the converged solution at time $t = t_n$ and, likewise, A_{n+1} is the value of A for the converged solution at the end of the current time increment, i.e., $t = t_{n+1}$. Again, the time step size is $\Delta t = t_{n+1} - t_n$. Because the solution scheme is iterative, we represent the *estimate* of the solution variable A_{n+1} for the k^{th} iteration as $A_{k,n+1}$ or, more simply, A_k. Thus, $A_{n+1} \leftarrow A_k$ after the final iteration is complete and the solution has converged for time $t = t_{n+1}$.

First, an algorithm for the implicit update of a generalized Maxwell model (i.e., Equation 5) is summarized in Algorithm 1 (as derived in Abaqus 2010 and Appendix C) which is exact under the condition that the viscoelastic strain rate is constant over the step, i.e., $\dot{e^{ve}_{ij}} = \Delta e^{ve}_{ij}/\Delta t \; \forall \, t \in [t_n, t_{n+1}]$.

Algorithm 1 Maxwell

Input: $\texttt{VARS}: \; \Delta\mathbf{e}^{ve}$, Δt, $\mathbf{e}^{ve}_n$, $\boldsymbol{\alpha}^{(\kappa)}_n \quad \texttt{PARAMS}: G_0, G^{(\kappa)}, \tau^{(\kappa)} \quad \forall\, \kappa \in [1, N]$

Output: $\mathbf{S}_{n+1}$, $\boldsymbol{\alpha}^{(\kappa)}_{n+1}$

1: $\Delta\boldsymbol{\alpha}^{(\kappa)} = \left(1 - \exp\left[\frac{-\Delta t}{\tau^{\kappa}}\right]\right)\left(\mathbf{e}^{ve}_n - \boldsymbol{\alpha}^{(\kappa)}_n\right) + \frac{\tau^{(\kappa)}}{\Delta t}\left(\frac{\Delta t}{\tau^{(\kappa)}} + \texttt{exp}\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right] - 1\right)\Delta\mathbf{e}^{ve}$

2: $\boldsymbol{\alpha}^{(\kappa)}_{n+1} = \boldsymbol{\alpha}^{(\kappa)}_n + \Delta\boldsymbol{\alpha}^{(\kappa)}$

3: $\mathbf{S}_{n+1} = G_0\left(\mathbf{e}_{n+1} - \sum_{\kappa=1}^{N}\frac{G^{(\kappa)}}{G_0}\boldsymbol{\alpha}^{(\kappa)}_{n+1}\right)$

Next we write Equations 2, 5, and 6 in a discrete sense at time $t = t_{n+1}$, i.e.,

$$\mathbf{e}^{D}_{n+1} = \frac{1}{2G_0}\left(\frac{c_{n+1}}{a}\right)^3 \mathbf{S}_{n+1} \tag{32}$$

$$\mathbf{S}_{n+1} = \texttt{Maxwell}\left(\mathbf{e}^{ve}_{n+1}, \mathbb{S}_n\right) \tag{33}$$

$$\mathbf{e}^{ve}_{n+1} = \mathbf{e}_{n+1} - \mathbf{e}^{D}_{n+1} \tag{34}$$

Substitution of Equations 32 and 34 into Equation 33 produces an expression for the implicit update of stress, i.e.,

$$\mathbf{S}_{n+1} = \texttt{Maxwell}\left(\left[\mathbf{e}_{n+1} - \frac{1}{2G_0}\left(\frac{c_{n+1}}{a}\right)^3 \mathbf{S}_{n+1}\right], \mathbb{S}_n\right) \tag{35}$$

While it is the algorithmic objective to solve this nonlinear system of equations, Equation 35 is not in a form useful for developing the algorithm. Instead, we substitute an iterative approximation for the damage strain and stress at time $t = t_{n+1}$ into Equation 32, accordingly

$$\mathbf{e}_k^D \approx \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \mathbf{S}_k \tag{36}$$

which enables the definition of a residual error term, i.e.,

$$\mathbf{R}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \mathbf{S}_k - \mathbf{e}_k^D \tag{37}$$

Algorithm 2 solves for the roots $\mathbf{R}_{n+1}\left(\mathbf{e}_{n+1}^{ve}\right) = \mathbf{0}$ using a Newton iterative solver as explained in detail here. Algorithm 2 requires as input arguments (1) the time step size and increment in total strain, Δt, and $\boldsymbol{\Delta}\mathbf{e}$, respectively, (2) the state of the material at the beginning of the increment, $\boldsymbol{S}_n = \left(\boldsymbol{\sigma}_n, \mathbf{e}_n^{ve}, \mathbf{e}_n^D, \boldsymbol{\alpha}_n^{(\kappa)}, c_n\right)$, and (3) the material model parameters listed in Table 1. The output arguments returned from the algorithm comprise (1) the deviatoric stresses at the end of the current time increment, $\mathbf{S}_{n+1}$ and (2) the state of the material at the end of the increment, $\boldsymbol{S}_{n+1} = \left(\mathbf{e}_{n+1}^{ve}, \mathbf{e}_{n+1}^D, \boldsymbol{\alpha}_{n+1}^{(\kappa)}, c_{n+1}\right)$. Algorithm 2 begins by (Step 1) initializing the iterate of the solution variable, i.e., $\boldsymbol{\Delta}\mathbf{e}^D = \mathbf{0}$ and setting the iteration counter $k = 0$. Steps 2-9 reflect a loop which is repeated until convergence criteria are met or the maximum number of iterations has been exceed in which case the current global FE solution iteration is aborted and repeated with a smaller time step size. Within this loop, Step 3 updates the viscoelastic and damage strains. Using the current approximation of the viscoelastic strain, Step 4 employs Algorithm 1 to compute the corresponding stresses and Maxwell state variables. The resulting stresses are used in an algorithm (UpdateCrack, whose details are addressed below) to update the average crack size consistent with Equations 9a and 9b. The residual error for the current iteration is computed in Step 6 and used in Newton fashion to compute the next iterate of the solution variable in Step 8. The system Jacobian used for a full Newton method to solve Equation 37 is

$$\mathbb{J} = \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{e}^D}, \tag{38}$$

which is derived in Appendix E. The iteration is deemed to have converged when the error norm (Step 9) is less than some predefined allowable error tolerance. Algorithms 1 and 2 are relatively straightforward and complete with the exception of the crack size update calculation (i.e., CrackUpdate) to be discussed in the next section.

Algorithm 2 Semi-Implicit Update Scheme

Input: `VARS` : $\Delta\mathbf{e}$, Δt, $\boldsymbol{\sigma}_n$, $\mathbf{e}_n^{ve}$, $\mathbf{e}_n^D$, $\boldsymbol{\alpha}_n^{(\kappa)}$, c_n `PARAMS` : see Table 1

Output: $\mathbf{S}_{n+1}$, $\mathbf{e}_{n+1}^D$, $\mathbf{e}_{n+1}^{ve}$, $\boldsymbol{\alpha}_{n+1}^{(\kappa)}$, c_{n+1}

1: **initialize:** $k \leftarrow 0$, $\Delta\mathbf{e}^D = \mathbf{0}$

2: **repeat**

3: $\Delta\mathbf{e}^{ve} = \Delta\mathbf{e} - \Delta\mathbf{e}^D, \quad \mathbf{e}_k^D = \mathbf{e}_n^D + \Delta\mathbf{e}^D$

4: **call** `Maxwell` $\longrightarrow \mathbf{S}_k$, $\boldsymbol{\alpha}_k^{(\kappa)}$

5: **call** `RK3` $\longrightarrow c_k$

6: $\mathbf{R}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \mathbf{S}_k - \mathbf{e}_k^D$

7: $\mathbb{J}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \mathbf{C}_k' + \frac{9\Delta t\sqrt{\pi c_k}}{4G_0 a\bar{\sigma}}\left(\frac{c_k}{a}\right)^2 \frac{\partial \dot{c}}{\partial K_I}\mathbf{S}_k \otimes \mathbf{S}_k \cdot \mathbf{C}_k' + \mathbb{I}$

8: $\Delta\mathbf{e}^D \leftarrow \Delta\mathbf{e}^D - \mathbb{J}_k^{-1} : \mathbf{R}_k$

9: **until** $\mathbf{R}_k : \mathbf{R}_k <$ `TOL` $(k \leftarrow k+1)$

10: **update:** $\mathbf{S}_{n+1} = \mathbf{S}_k$, $\mathbf{e}_{n+1}^D = \mathbf{e}_k^D$, $\mathbf{e}_{n+1}^{ve} = \mathbf{e}_n^{ve} + \Delta\mathbf{e}^{ve}$, $\boldsymbol{\alpha}_{n+1}^{(\kappa)} = \boldsymbol{\alpha}_k^{(\kappa)}$, $c_{n+1} = c_k$

Explicit subincrement integration of crack growth kinetics

The integration of crack growth rate Equations 9a and 9b is performed using the subincrement explicit integration scheme described in Algorithm 3. If Algorithm 3 used an implicit scheme, then Algorithm 2 would be fully-implicit; however, the stiff and piecewise nonlinear system of equations presented by Equations 9a and 9b are prohibitive to an efficient, reliable, and robust solution strategy. Even though the integration scheme presented in Algorithm 3 is explicit, collectively Algorithms 1-3 ensure all solution variables are in agreement with their corresponding constitutive equations to within a finite allowable error at the end of the time increment, $t = t_{n+1}$. This is not true for a purely explicit non-iterative update scheme.

Algorithm 3 is an automatic time stepping implementation of a 3rd order embedded Runge-Kutta algorithm described Algorithm 4. The arguments provided as input to Algorithm 3 are (1) the stresses at the end of the current (previous) time increment, $\boldsymbol{\sigma}_{n+1}$ ($\boldsymbol{\sigma}_n$), (2) the mean crack size at the end of the previous increment c_n, (3) the time step size Δt, and (4) the crack rate model parameters. The algorithm returns the crack size at the end of the current increment, i.e., c_{n+1} as well as the algorithmically consistent tangents $\frac{\partial c}{\partial \bar{\sigma}}$, $\frac{\partial c}{\partial P}$, and $\frac{\partial c}{\partial V_{res}}$.

In Step, the average stress rates over the incriment are calculated. In Step 2, the algorithm tries to complete the integration in one time step by setting the subincrement $\Delta t^* = \Delta t$, and the total time increment completed $\Delta t_{total} = 0$ and the subincrement initial stress $\boldsymbol{\sigma}_k$, and initial crack size c_k are set to their associated values at the beginning of the total increment. The tangents are also set to zero. Steps 3-16 reflect a loop which is performed until total of the subincrements Δt_{total} is equal to the total increment Δt. Assuming the recommended subincrement size is bigger than the current subincrement size then the next subincriment size is increased by a fraction of the difference between the recommended subincriment size and the current subincrement size in Step 10. If the recomended subincrement size is smaller than the current subincriment then none of the calculations are stored and a new subincriment is tried in Step 12. Steps 14-16 assures that the sum of the subincrements does not exceed the total increment.

One additional step which might be prudent is to set a maximum number of cutbacks allowed in Algorithm 3. If the max is exceeded then maybe a global cutback would be recommended and could be incorporated in the automatic timesteping algorithm of the next section.

Algorithm 3 RK_MULTISTEP, 3rd order subincrimentation embedded Runge-Kutta crack update

Input: `Vars` : $\boldsymbol{\sigma}_{k+1}$, $\boldsymbol{\sigma}_k$, c_k, Δt, `Params` : $K_0, \mu', m, v_{res}, b[:], b^*[:], d[:], a[:,:]$

Output: c_{k+1}, $\frac{\partial c}{\partial \bar{\sigma}}$, $\frac{\partial c}{\partial p}$, $\frac{\partial c}{\partial V_{res}}$

$\dot{\bar{\sigma}} = \frac{\bar{\sigma}_{k+1} - \bar{\sigma}_k}{\Delta t}$, $\dot{P} = \frac{P_{k+1} - P_k}{\Delta t}$

initialize: $\Delta t^* = \Delta t, \quad \Delta t_{complete} = 0, \quad \frac{\partial c}{\partial \bar{\sigma}} = 0\ , \quad \frac{\partial c}{\partial P} = 0, \quad \frac{\partial c}{\partial V_{res}} = 0$

$c_j = c_k, \quad \bar{\sigma}_j = \bar{\sigma}_k, \quad P_j = P_k$

repeat

  $\bar{\sigma}_{j+1} = \bar{\sigma}_j + \dot{\bar{\sigma}}\Delta t^*$, $P_{j+1} = P_j + \dot{P}\Delta t^*$

  call `RK3` $\longrightarrow c_{j+1}, \frac{\partial c_{j+1}}{\partial \bar{\sigma}}, \frac{\partial c_{j+1}}{\partial P}, \frac{\partial c_{j+1}}{\partial V_{res}}, \Delta t_{rec}$

  if $\Delta t^* \leq \Delta t_{rec}$ **then**

    $\Delta t_{complete} = \Delta t_{complete} + \Delta t^*$

    $\frac{\partial c}{\partial \bar{\sigma}} = \frac{\partial c}{\partial \bar{\sigma}} + \frac{\partial c_{j+1}}{\partial \bar{\sigma}}\Delta t^*, \quad \frac{\partial c}{\partial P} = \frac{\partial c}{\partial P} + \frac{\partial c_{j+1}}{\partial P}\Delta t^*, \quad \frac{\partial c}{\partial V_{res}} = \frac{\partial c}{\partial V_{res}} + \frac{\partial c_{j+1}}{\partial V_{res}}\Delta t^*$

    $c_j = c_{j+1} \quad \bar{\sigma}_j = \bar{\sigma}_{j+1}, \quad P_j = P_{j+1}$

    $\Delta t^* = \Delta t^* + 0.9\,(\Delta t_{rec} - \Delta t^*)$

  else

    $\Delta t^* = 0.9\Delta t_{rec}$

  end if

  if $\Delta t^* < \Delta t - \Delta t_{complete}$ **then**

    $\Delta t^* = \Delta t - \Delta t_{complete}$

  end if

until $\Delta t_{complete} = \Delta t$

$c_{k+1} = c_{j+1}, \quad \frac{\partial c}{\partial \bar{\sigma}} = \frac{1}{\Delta t}\frac{\partial c}{\partial \bar{\sigma}}, \quad \frac{\partial c}{\partial p} = \frac{1}{\Delta t}\frac{\partial c}{\partial P}, \frac{\partial c}{\partial V_{res}} = \frac{1}{\Delta t}\frac{\partial c}{\partial V_r es}$

Automatic time stepping algorithm

An automatic time stepping algorithm was implemented to inform the "parent code" to determine the appropriate time step and allow it to grow as the loading and solution approach steady state. We use parent code here to refer to the finite element code responsible for solving the global equations of motion and calling the constitutive update routine, e.g. *Abaqus/Standard*. The integration algorithm for the viscoelastic stresses developed in Appendix C is exact if the strain rate is constant over the integral in Equation 66. A constant strain rate assumption is convenient because the parent code typically only passes the strain and time increments, thus a constant strain rate is often inferred from those quantities. Note, of course, that integration of the global equations of motion includes an inertial term reflecting acceleration of continuum material points, thus the velocities are not constant over an increment and, consequently, neither is strain rate in a strict algorithmic sense. Moreover, the strain rate in the viscous sub-elements of the generalized Maxwell model are not generally constant. The automatic time stepping algorithm presented here ensures that the change in sub-elemental strain rate over an increment is small.

The viscous strain rate in each element is given by Equation 46 but is reproduced here in deviatoric form for convenience.

$$\dot{\boldsymbol{\alpha}}^{(\kappa)} = \frac{\mathbf{S}^{(\kappa)}}{\eta^{(\kappa)}} \tag{39}$$

where $\dot{\boldsymbol{\alpha}}^{(\kappa)}$ is the viscous strain rate, $\mathbf{S}^{(\kappa)}$ is the deviatoric stress, and $\eta^{(\kappa)}$ is the viscous coefficient for the κ^{th} Maxwell element. Alternatively, the deviatoric stress in the Maxwell element can be expressed in terms of the elastic (recoverable) strain in the element, i.e.,

$$\mathbf{S}^{(\kappa)} = G^{(\kappa)} \left(\mathbf{e}^{ve} - \boldsymbol{\alpha}^{(\kappa)} \right) \tag{40}$$

where $G^{(j)}$ is the shear modulus corresponding to the κ^{th} Maxwell element. Note that $\boldsymbol{\alpha}^{(\kappa)}$ are stored by the subroutine as state variables for the viscoelastic calculations. Combining Equations 39 and 40 with $\tau = \eta/G$ results in

$$\dot{\boldsymbol{\alpha}}^{(\kappa)} = \frac{1}{\tau^{(\kappa)}} \left(\mathbf{e}^{ve} - \boldsymbol{\alpha}^{(\kappa)} \right) \tag{41}$$

The change in internal strain rate over a time increment is

$$\Delta\dot{\boldsymbol{\alpha}}^{(\kappa)} = \dot{\boldsymbol{\alpha}}_{n+1}^{(\kappa)} - \dot{\boldsymbol{\alpha}}_{n}^{(\kappa)} = \frac{\Delta\mathbf{e}^{ve} - \Delta\boldsymbol{\alpha}^{(\kappa)}}{\tau^{(\kappa)}} \tag{42}$$

where $t^{(k+1)}$ and $t^{(k)}$ are the times at the end and beginning of the increment respectively. A weighted effective change in strain rate over an increment is defined as

$$\Delta\dot{\boldsymbol{\alpha}} = \sum_{\kappa=1}^{N} \frac{G^{(\kappa)}}{G_0} \Delta\dot{\boldsymbol{\alpha}}^{(\kappa)} \tag{43}$$

and a heuristic dimensionless metric that reflects the effective change in internal strain over an increment due to a non-constant strain rate, which is computed as the scalar product of $\Delta\dot{\boldsymbol{\alpha}}$ with itself multiplied by Δt for convenience, i.e.,

$$\Delta\alpha_{eq} = \Delta t \sqrt{\frac{2}{3}\Delta\dot{\boldsymbol{\alpha}} : \Delta\dot{\boldsymbol{\alpha}}} \tag{44}$$

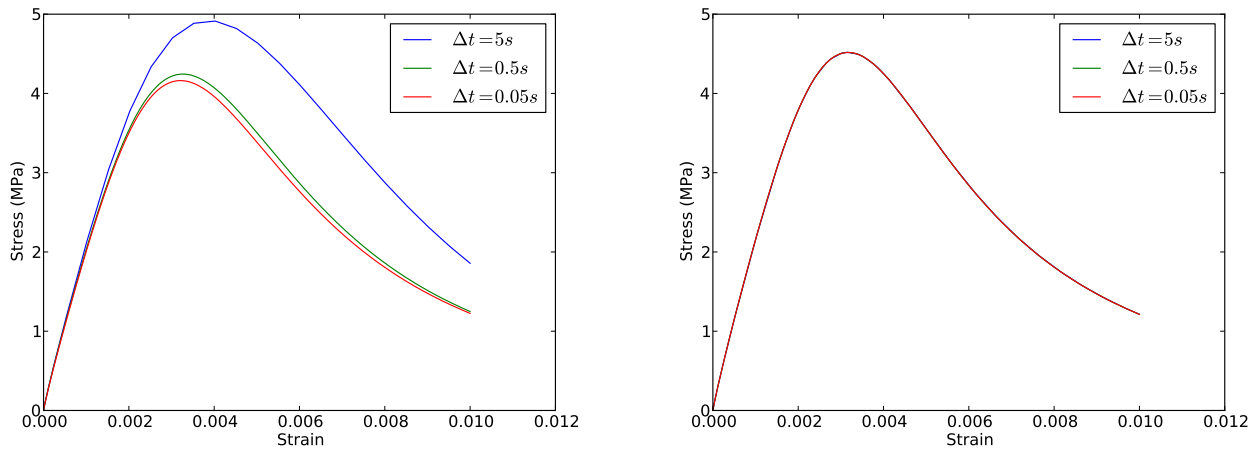
A limit on the maximum acceptable value of $\Delta\alpha_{eq}$ is set *a priori*, which can be conceptually related to a maximum acceptable error in strain. If the calculated value for $\Delta\alpha_{eq}$ exceeds the limit, then the user subroutine terminates the current global solution increment and requests a reduced time step size be used for the next attempted increment. This implementation enables the user to allow the global finite element code to attempt larger time steps and automatically reduce the time step size if necessary.

6 Comparison with historical implementation

In this section a historical explicitly integrated implementation is compared with the proposed implementation. We simply apply the material parameters provided with the original implementation to compare numerical performance of the two algorithms. The parameters are listed in Appendix I

Because in the new implementation damage growth is not retarded by tensile mean pressure the two implementations are not expected to give consistent results so the comparisons are only carried out in compression. First, a 3D multi-element simulation of a compression experiment at a constant strain rate of 1e-4 s^{-1} is performed for both implementations. The maximum allowed time step is varied from 0.05s to 5s. The results of these simulations are shown in Figure 5. The old formulation exhibits great dependance on time step in

the behavior once damage starts to accumulate, though the dependance appears convergent. The dependance on time step in the behavior of the new formulation is insignificant.



(a) Old formulation

(b) New formulation

Figure 5: Compression simulations at $\dot{\epsilon} = 1e - 4s^{-1}$

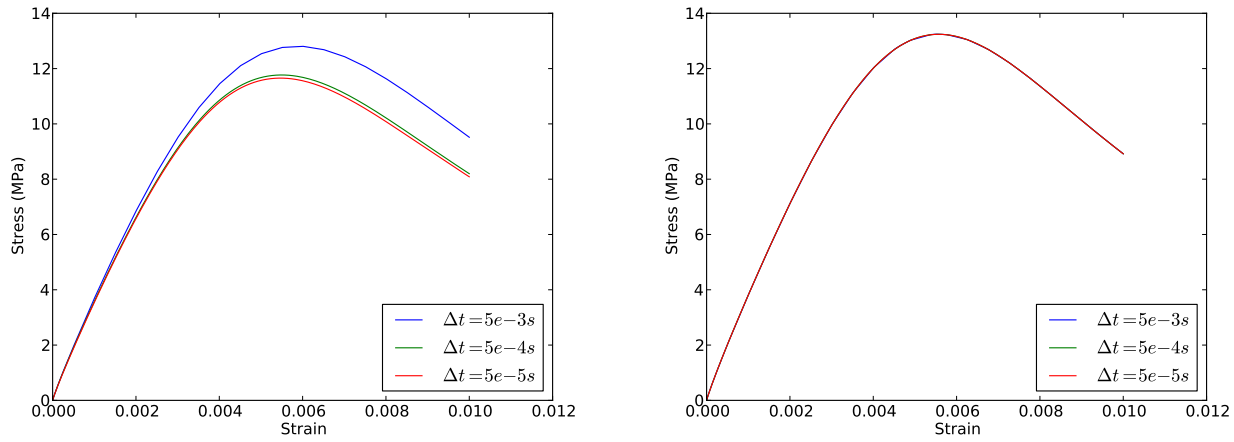
While accuracy is the major motivation for the new implementation excessive computation time is also undesirable. The computation time for the compression simulations is listed in Table 2. There are several interesting observations that can be made. First, the time required to perform the simulations with a maximum time step of 5 seconds and 0.05 seconds is longer for the new formulation than the old formulation but with a maximum time step of 0.5 seconds the new formulation seems to perform better. The slower performance of the new formulation with a 5 second time step is attributed to the auto time stepping algorithm which didn't allow the simulation to run at 5 second intervals. As the time step gets small the explicit algorithm of the old formulation should perform better and it does, but we should consider that the new formulation achieved convergence at a set time step of 5 seconds It may be better to compare the 36 second completion of the converged new formulation to the 717 seconds required to complete the nearly converged simulation using the old formulation.

The old formulation should perform better at relatively high rate so we studied the efficiency of the two

Table 2: Simulation Time

$\dot{\epsilon} = 1\text{e-}4s^{-1}$			$\dot{\epsilon} = 0.1s^{-1}$		
Δt_{max} (s^{-1})	Old Form (s)	New Form (s)	Δt_{max} (s^{-1})	Old Form (s)	New Form (s)
5.0	31.0	36.3	5e-3	23.5	22.6
0.5	166	128	5e-4	153	121
0.05	717	832	5e-5	677	811

formulation at a strain rate of 0.1 s^{-1}. The results are shown in Figure 6 and the computation times are shown in the right half of Table 2. Here the new formulation does appear to be slower than the old formulation at a given maximum time step but again the new formulation seems to exhibit convergence at the largest time step size of 5e-3 seconds. Conversely, the old formulation may have converged at a time step size of 5e-5 so we are comparing 22.6 seconds for the new formulation to 677 seconds for the old formulation.



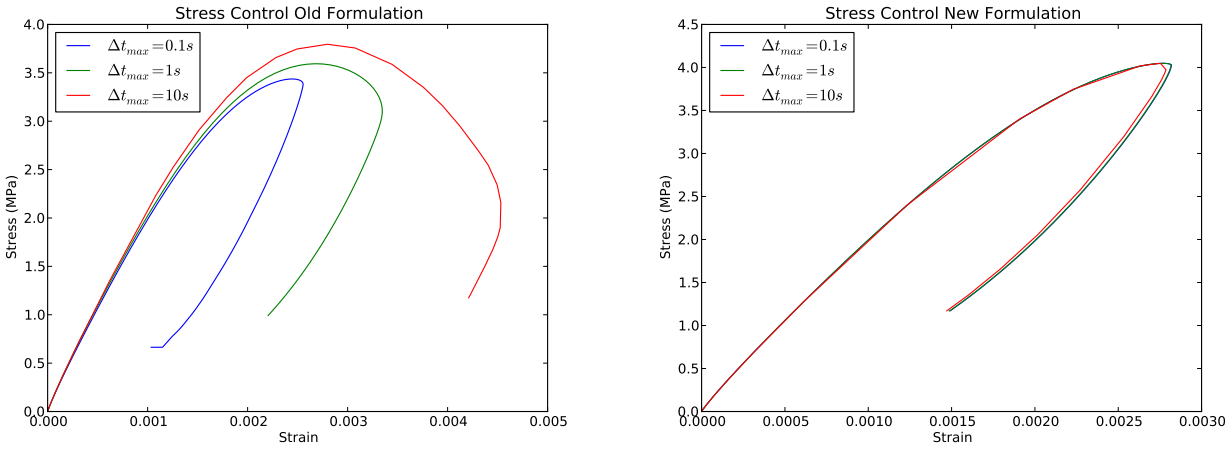
(a) Old formulation

(b) New formulation

Figure 6: Compression simulations at $\dot{\epsilon} = 0.1s^{-1}$

Next, a series of stress controlled compression simulations were performed. The stress and time were output from the constant strain rate simulations and applied to the sample. Then stress was scaled by 0.95 to keep

the response from hitting the instability at the peak. The results of these simulations are shown in Figure 9. The response of the old formulation exhibits great dependance on time step size while the new formulation exhibits very little dependance on time step size. The old formula simulations unload in a different way than the new formulation. This is likely because the model exhibits such large dependance on time step size. A smaller scale factor may be required to assure that the old formula simulations do not pass the snap-through point in the stress-strain behavior.



(a) Old formulation

(b) New formulation

Figure 7: Stress controled compression simulations

A final series of compression simulations was performed using the new implementation to show good performance with respect to mesh size variation. Three meshes resolutions were chosen. The coarse mesh corresponds with the mesh used in the previous simulations. The medium and fine meshes are the result of consecutive mesh doubling. The mesh size dependance exhibited is insignificant which is expected of a well performing model because the stress in a uniaxial simulation should exhibit no spatial dependance.

To demonstrate the new formulation in a simulation where a spatially varying stress is present a cantilever beam with a tip load was modeled. Figure 9a and 9b show the change in load-deflection behavior with time step and mesh size respectively. There is no dependance on time step size though there is a dependance mesh size once damage starts to accumulate. This is expected as the model does not yet try to assure consistent

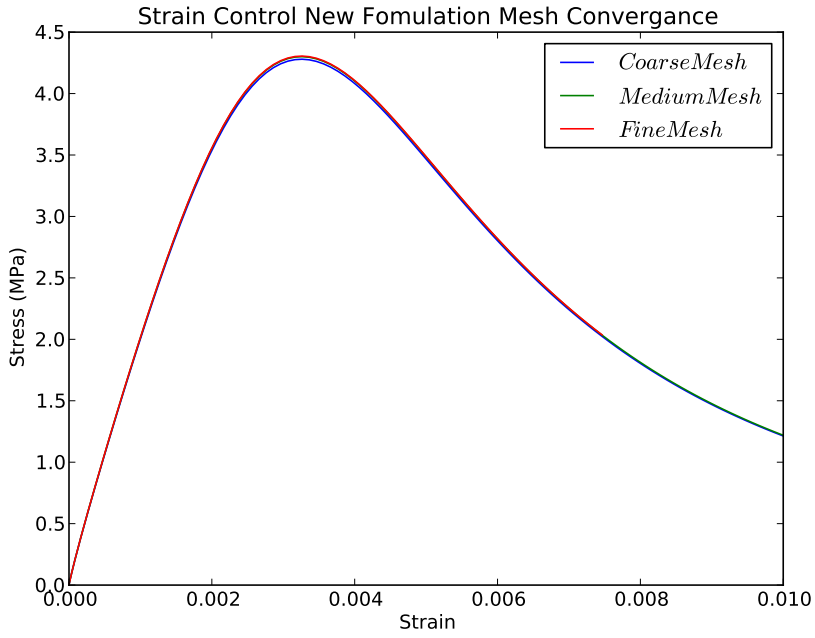
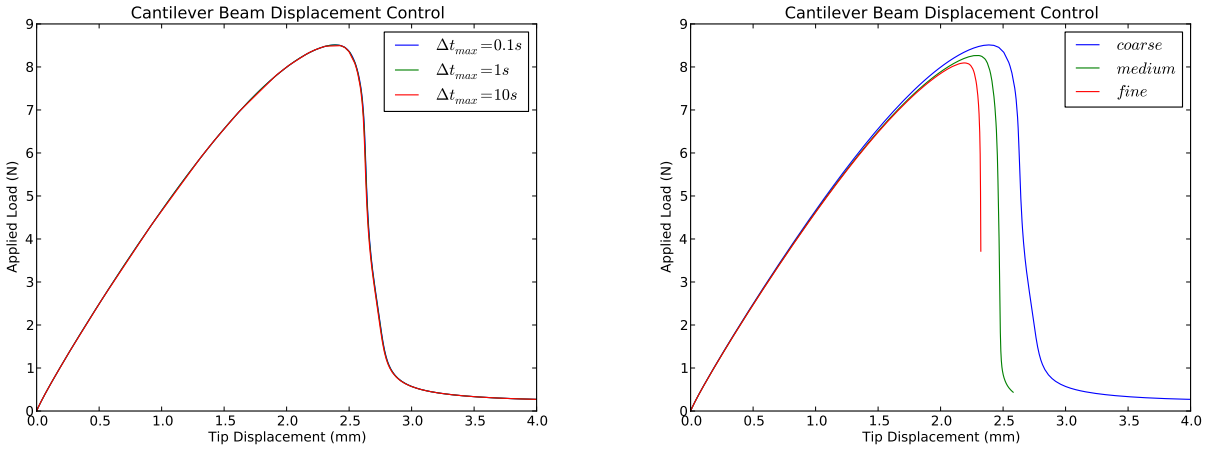


Figure 8: Mesh convergence of new formulation

specific energy dissipation during failure.



(a) Temporal convergence

(b) Mesh convergence

Figure 9: Temporal and spatial convergence in cantilever beam bending simulations

7 Comparrison to experimental data

A preliminary fitting method involving brute force adjustment of parameters was used to fit uniaxial compression and tension data collected by Darla Thompson. This section documents how well the model can fit the experimental data. First we compared a series of simulations to demonstrate the model and estimated parameters performance with respect to uniaxial tension experiments performed by Darla Thompson in 2009 at room temperature (23°C). The tests were performed at nominally constant strain rates varying from about 2e-6 s^{-1} to 2e-2s^{-2} This data was heavily weighted during the brute force calibrations so these comparisons are expected to be very good. Simulation results along with their corresponding experimental data is shown in Figure 10. The simulations are plotted with a dotted line and their corresponding experimental data is plotted with a solid line of the same color. As expected the data fits the loading phase very well but doesn't fit the failure very well. The failure is brittle in nature and this model is unlikely to capture the failure behavior very well and as demonstrated previously would be completely unable to consistently predict this behavior over different meshes so this region was ignored during the fitting process.

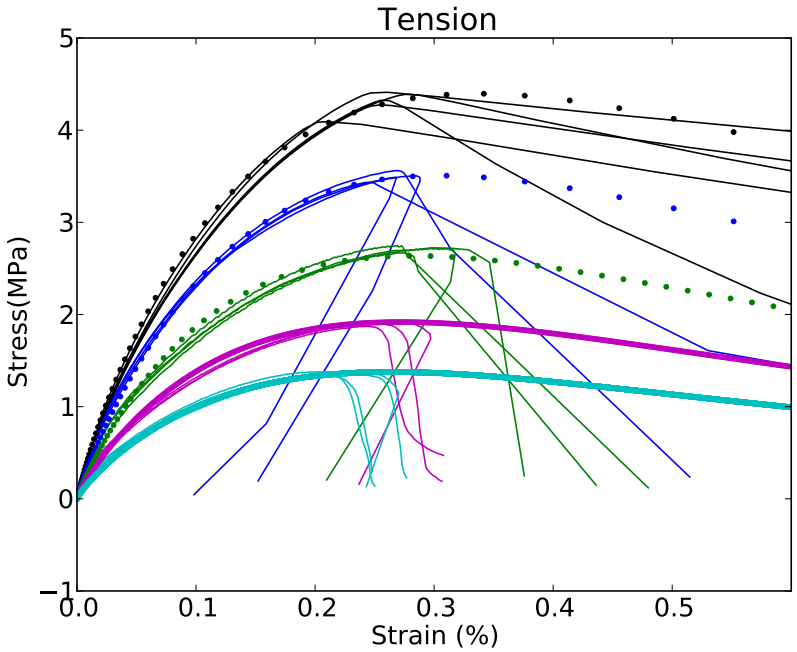


Figure 10: Comparison of uniaxial tension simulations to experimental data

Comparisons of compression simulations to experiments are shown in Figure 11. Again simulations are shown dotted and experiments are shown using solid lines. The rates vary from 1e-6 s^{-1} to 1e-2s^{-2}. The compression comparisons are less than perfect. The only good fits are at about 1e-4s^{-1} and 1e-3s^{-1}.

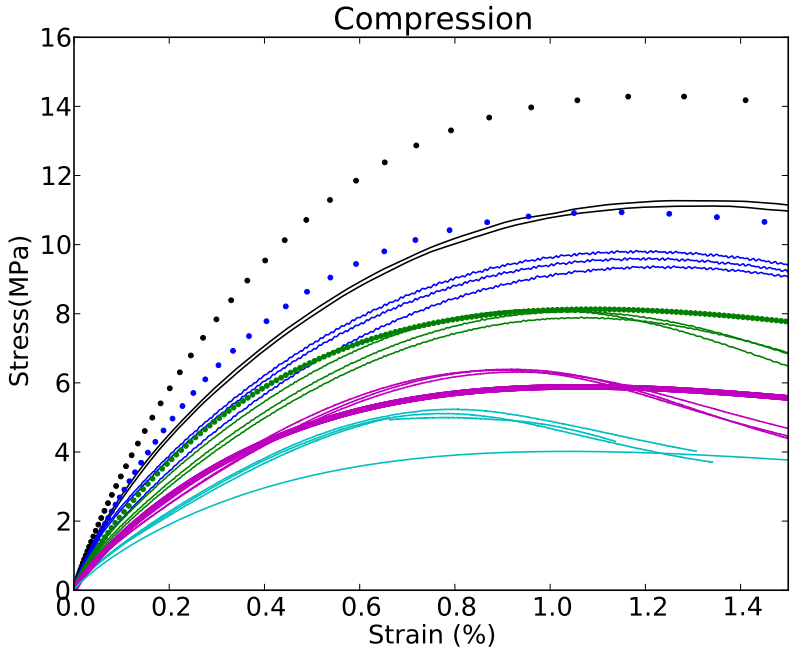


Figure 11: Comparison of uniaxial compression simulations to experimental data

Finally, we compared the model performance in creep simulations. Uniaxial compression creep tests were modeled at 30°C for three load levels including 0.5 MPa 1.0 MPa, and 1.5 MPa. They are plotted in Figure 12. The 1.0 MPa simulations is a very good approximation of the experiment but the 0.5 MPa simulation over predicts the creep of the experiment and the 1.5 MPa simulation under predicts the creep. The simulated behavior is consistent with linear viscoelasticity. It is hypothesized that the material has a plastic behavior which the model is unable to capture and thus fitting more than one of these curves is currently not possible.

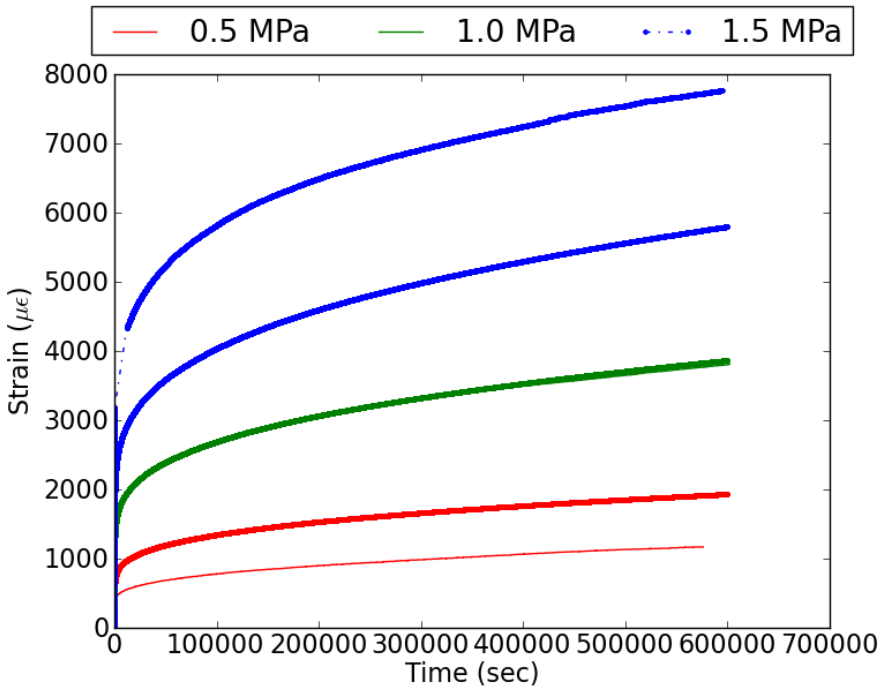


Figure 12: Comparison of creep simulations to experimental data

8 Summary and concluding remarks

The goal of this work was to improve implementation of an existing set of constitutive equations. The prominent inadequacy of the previous implementation is the behavior with large time steps. This was demonstrated in Section 6. The explicit crack update algorithm used in the previous formulation is at fault for the poor behavior with large time steps. The old formulation also did not include any facility for determining the time step size, which required the user to determine the appropriate size. This is not practical in large multi-component simulations. The combination of the semi-implicit crack update and the automatic time step selection algorithm greatly improved the performance and usability of the material model. Comparisons of simulation results using both models were shown in Section 6, and even in relatively high rate situations the new formulation outperforms the old when comparing simulations of similar accuracy.

A brute force fitting process was performed and is discussed in Section 7. Uniaxial tensile data was weighted heavily in the fitting process, so the comparison to tensile experimental data is quite good. It should

be noted that the parameters selected in the fitting process resulted in a saturation of the stress intensity contribution to the crack growth equations in tension. This is to say that in the tensile simulations strain rate is the only important factor in the response. We do not state that this is correct, just that it fits the data very well. Conversely, the only parameter left to change after fitting the tensile data is the friction coefficient μ'. This provided a decent fit to strain rates in the range 1e-4 s^{-1} to 1e-3s^{-2}, but relatively bad fits outside this range. Other techniques for fitting are currently being explored which take advantage of deconvolving the viscoelastic response from the viscodamage response. These techniques could potentially improve the fits data, but ultimately this set of constitutive equations might not be appropriate for this model.

Our two concerns are the lack of nonlinearity in the volumetric response, and the lack of a permanent deformation mechanism. The mean stress is attributed to a linear elastic bulk modulus and there is not volumetric strain associated with the damage. Uniaxial data can be fit with the interesting mechanics isolated to the deviatoric response, but the fits are not unique so appropriate behavior under other stress states in unlikely. Experimental evidence also suggests that permanent deformation mechanisms are important Buechler (2012). This model must attribute all the nonviscoelastic response to damage. Because some of the response is attributable to plasticity the damage likely evolves too fast in order to fit the monotonic experimental data.

The next generation of constitutive models are currently being developed. The developers may take advantage of lessons learned during development of this new implementation.

Acknowledgements

We would like to thank the Enhanced Surveillance Campaign (ESC) for supporting the constitutive model development documented in this report as well as Darla Thompson for the experimental data she collected under ESC.

References

Abaqus. *Abaqus theory manual*. Simulia, Providence, RI, version 6.10 edition, 2010.

F. L. Addessio and J. N. Johnson. A constitutive model for the dynamic-response of brittle materials. *J. Appl. Phys.*, 67(7):3275–3286, 1990.

J. G. Bennett, K. S. Haberman, J. N. Johnson, B. W. Asay, and B. F. Henson. A constitutive model for the non-shock ignition and mechanical response of high explosives. *J. Mechanics Phys. Solids*, 46(12):2303–2322, 1998.

Miles A. Buechler. Observed physical processes in mechanical tests of pbx9501 and recommendations for experiments to explore a possible plasticity/damage threshold. Technical Report LA-UR-12-21072, Los Alamos National Laboratory, 2012.

J. K. Dienes. A statistical-theory of fragmentation processes. *Mechanics Materials*, 4(3-4):325–335, 1985.

J. K. Dienes. Strain-softening via scram. Technical Report LA-UR-98-3620, Los Alamos National Laboratory, 1998.

J. K. Dienes, Q. H. Zuo, and J. D. Kershner. Impact initiation of explosives and propellants via statistical crack mechanics. *J. Mechanics Phys. Solids*, 54(6):1237–1275, 2006.

Robert M. Hackett and Joel G. Bennett. An implicit finite element material model for energetic particulate composite materials. *International Journal for Numerical Methods in Engineering*, 2000.

Lawrence E. Malvern. *Introduction to the Mechanics of a Continuous Medium.* Prentice-Hall, Englewood Cliffs, New Jersey, 1969.

L.F. SHAMPINE P. BOGACKI. A 3(2) pair of runge - kutta formulas. *Appl. Math. Lett.*, 2(4):321–325, 1989.

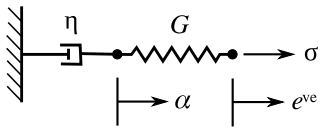


Figure 13: Rheological representation of a single Maxwell element consisting of a damper with viscosity η and a spring of stiffness G in series.

Appendices

A Development of stress equation for a single Maxwell element

Derivations of the relationship between viscoelastic strain and stress for a single Maxwell element, shown in Figure 13, are presented here. These derivations essentially follow those presented in, for example, Malvern (1969) and Abaqus (2010). A single Maxwell element consists of a spring and viscous damper in series as shown in Figure 13. The strain in the damper is α and the total viscoelastic strain across the Maxwell element is e^{ve}. Thus the strain across the elastic spring is $(e^{ve} - \alpha)$. Through Hooke's law, the stress acting on the spring is

$$\sigma = G\,(e^{ve} - \alpha) \tag{45}$$

where σ is the stress and G is the shear modulus (spring stiffness). Equilibrium dictates that the same stress, i.e. σ, is applied (transmitted through the spring) to the damper resulting in the viscous strain rate

$$\dot{\alpha} = \frac{\sigma}{\eta} \tag{46}$$

where η reflects the viscosity of the damper. Substitution of Equation 46 into the time derivative of Equation 45 enables one to solve for the stress rate, i.e.,

$$\dot{\sigma} = G\dot{e}^{ev} - \frac{\sigma}{\tau} \tag{47}$$

where a relaxation time constant has been introduced and is defined as $\tau = \eta/G$. If we apply a Laplace transform to both sides of Equation 47 we get

$$s\sigma(s) = sGe^{ve}(s) - \frac{\sigma(s)}{\tau} \tag{48}$$

or

$$\sigma(s) = \frac{sG}{s + \frac{1}{\tau}} e^{ve}(s) \tag{49}$$

which transformed back using the the inverse Laplace transform results in the convolution integral

$$\sigma(t) = \int_0^t Ge^{\frac{-(t-\zeta)}{\tau}} \dot{e}^{ev} d\zeta \tag{50}$$

B Development of stress equation for a generalized Maxwell element

The single Maxwell element can be generalized to include N Maxwell elements in parallel with an elastic spring as shown in Figure 14. These derivations essentially follow those presented in, for example, Malvern (1969) and Abaqus (2010). Compatibility of the deformation of each Maxwell element requires that the total

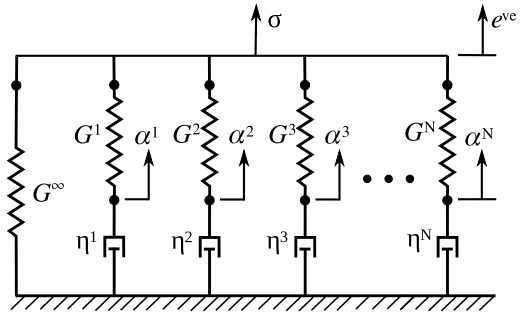


Figure 14: Rheological representation of a generalized Maxwell element comprising N Maxwell elements in parallel with an elastic spring.

strain of the general Maxwell model be equal to the viscoelastic strain for all individual elements. Equilibrium requires that the total stress acting on the general Maxwell model, σ, is the sum of the internal stresses acting

on each of the individual elements $\sigma^{(j)}$. Accordingly, summing Equation 49 over the N elements and including a contribution from the elastic spring gives

$$\sigma(s) = \left(G^{\infty} + \sum_{\kappa=1}^{N} \frac{sG^{(\kappa)}}{s + \frac{1}{\tau^{(\kappa)}}} \right) \epsilon(s) \tag{51}$$

which can be transformed back to the time domain using an inverse Laplace transform resulting in

$$\sigma(t) = \int_0^t G^{\infty} + \sum_{\kappa=1}^{N} \left(G^{(\kappa)} \exp\left[\frac{-(t-\zeta)}{\tau^{(\kappa)}} \right] \right) \dot{\epsilon}^{ev} d\zeta \tag{52}$$

Typically, Equation 52 is written as

$$\sigma(t) = \int_0^t G(t-\zeta) \dot{\epsilon}^{ev} d\zeta \tag{53}$$

where the relaxation modulus, $G(t)$ is expressed as a Prony-series, i.e.,

$$G(t) = G^{\infty} + \sum_{\kappa=1}^{N} G^{(\kappa)} \exp\left[\frac{-t}{\tau^{(\kappa)}} \right] \tag{54}$$

Thus far, the developed rheological models are expressed in terms of scalar variables representing stress, strain, and internal dissipative strain. It is striaghtforward to generalize the relationships to tensoral form; here, we accomplish this by decoupling the deviatoric and volumetric responses, each being modeled with a seperate generalized Maxwell description, i.e.,

$$\sigma_{ij}(t) = \int_0^t 2G(t-\zeta) \dot{e}_{ij}^{ev} d\zeta + \int_0^t \delta_{ij} \; K(t-\zeta) \dot{\epsilon}_{kk} d\zeta \tag{55}$$

where $\dot{e}^{ev}$ is the deviatoric strain rate, $\dot{\epsilon}_{kk}$ is the volumetric strain rate and the bulk relaxation modulus $K(t)$ is expressed by the Prony series

$$K(t) = K^{\infty} + \sum_{\kappa=1}^{N} K^{(\kappa)} \exp\left[\frac{-t}{\tau^{(\kappa)}} \right] \tag{56}$$

The challenge is determining the coefficients G^{∞}, K^{∞}, $G^{(\kappa)}$, $K^{(\kappa)}$, and $\tau^{(\kappa)}$ from experiments. Also, note that the time constants need not be the same for both the shear modulus and the bulk modulus, though for convenience they commonly are.

C Integration of viscoelastic stress equations

Two methods for integrating the viscoelastic stress equations in time are presented in this Section. The first method is introduced to provide some intuitive sense of the objective of the viscoelastic update algorithm and because it is relevant to legacy implementations of viscSCRAM. The second method is that implemented within the commercial finite element code, Abaqus, for its own viscoelastic material model (Abaqus, 2010). Derivations for this second method essentially follow that presented in Abaqus (2010).

The first method discussed is to sum Equation 47 over all the elements in the general Maxwell model and then to integrate the stress rate.

$$\dot{\sigma} = G^{\infty}\dot{\epsilon}^{ve} + \sum_{\kappa=1}^{N}\left(G^{(\kappa)}\dot{\epsilon}^{ve} - \frac{\sigma^{(\kappa)}}{\tau^{(\kappa)}}\right) \tag{57}$$

where $\sigma^{(\kappa)}$ is the stress carried by each individual Maxwell element. In similar fashion to the previous section, Equation 57 is generalized to a tensoral form in separate terms of deviatoric and volumetric response, i.e.,

$$\dot{\sigma}_{ij} = \dot{S}_{ij} + \delta_{ij}\dot{\sigma}^{m} \tag{58}$$

where

$$\dot{S}_{ij} = 2G^{\infty}\dot{e}^{ve}_{ij} + \sum_{\kappa=1}^{N}\left(2G^{(\kappa)}\dot{e}^{ve}_{ij} - \frac{S^{(\kappa)}_{ij}}{\tau^{(\kappa)}}\right) \tag{59}$$

$$\dot{\sigma}^{m} = K^{\infty}\dot{\epsilon}^{ve}_{kk} + \sum_{\kappa=1}^{N}\left(K^{(\kappa)}\dot{\epsilon}^{ve}_{kk} - \frac{\sigma^{(\kappa)}_{kk}}{3\tau^{(\kappa)}}\right) \tag{60}$$

and $S^{(\kappa)}_{ij}$ ($\sigma^{(\kappa)}_{kk}/3$) is the portion of the deviatoric (mean) stress carried by each Maxwell element (respective). A viscoelastic update algorithm can be built by applying an integration scheme (e.g. forward or backward Euler) to integrate Equations 59, and 60. However, note that the individual Maxwell stresses must also be integrated forward in time in a manner consistent with that for Equations 59 and 60. Consequently, the internal stresses would take on the role of state variable to reflect the history of the material.

An alternative method, exploited in Algorithm 1, for integrating the viscoelastic equations achieves an analytically exact (under certain conditions) update equation. The method can be applied to mean and deviatoric stresses can be integrated separately. Thus, only the deviatoric part is addressed here. The same

approach can be applied to update the mean stress if using a generalized Maxwell bulk viscoelastic model. First, we generalize Equation 50 to relate the deviatoric stress in the κ^{th} Maxwell element, i.e.,

$$S_{ij}^{(\kappa)} = 2G^{(\kappa)} \left(e_{ij}^{ve} - \alpha_{ij}^{(\kappa)} \right) \tag{61}$$

Likewise, a simple generalization of the single Maxwell element stress computed in Equation 50 gives the deviatoric stress in the κ^{th} Maxwell element, $S_{ij}^{(\kappa)}$, is

$$S_{ij}^{(\kappa)}(t) = \int_0^t 2G^{(\kappa)} \exp\left[\frac{-(t-\zeta)}{\tau^{(\kappa)}} \right] \dot{e}_{ij}^{ve} d\zeta \tag{62}$$

Each of the previous two equations provides the deviatoric stress acting within a single Maxwell element. The former, Equation 61, is based on the elasticity of the element in terms of the total and internal strains, while the latter, Equation 62, is the convolution integral for stress dependent solely on the total viscoelastic strain history. Substitution of Equation 62 into Equation 61 yields an expression for the internal (dissipative) strain associated with the κ^{th} Maxwell element, i.e.,

$$\alpha_{ij}^{(\kappa)}(t) = \int_0^t \left(1 - \exp\left[\frac{-(t-\zeta)}{\tau^{(\kappa)}} \right] \right) \dot{e}_{ij}^{ve} d\zeta \tag{63}$$

The internal strain of Equation 63 is evaluated at time $t = t_{n+1}$ and separated into two terms, i.e.,

$$\alpha_{ij}^{(\kappa)}(t_{n+1}) = \int_0^{t_n} \left(1 - \exp\left[\frac{-(t_{n+1}-\zeta)}{\tau^{(\kappa)}} \right] \right) \dot{e}_{ij}^{ve} d\zeta + \int_{t_n}^{t_{n+1}} \left(1 - \exp\left[\frac{-(t_{n+1}-\zeta)}{\tau^{(\kappa)}} \right] \right) \dot{e}_{ij}^{ve} d\zeta \tag{64}$$

Substituting the identity

$$1 - \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}} \right] + \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}} \right] \left(1 - \exp\left[\frac{\zeta - t_n}{\tau^{(\kappa)}} \right] \right) = 1 - \exp\left[\frac{\zeta - t_n}{\tau^{(\kappa)}} \right] \tag{65}$$

into Equation 64 and evaluating the integral for the first term of the latter produces

$$\alpha_{ij}^{(\kappa)}(t_{n+1}) = \left(1 - \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}} \right] \right) e_{ij,n}^{ve} + \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}} \right] \alpha_{ij}^{(\kappa)}(t_n) + \int_{t_n}^{t_{n+1}} \left(1 - \exp\left[\frac{\zeta - t_{n+1})}{\tau^{(\kappa)}} \right] \right) \dot{e}_{ij}^{ve} d\zeta \tag{66}$$

Approximating the strain rate as constant over a time increment, i.e., $\dot{e}_{ij}^{ve} \approx \Delta e_{ij}^{ve} / \Delta t$ allows the integral in Equation 66 to be evaluated resulting in

$$\alpha_{ij}^{(\kappa)}(t_{n+1}) = \left(1 - \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}} \right] \right) e_{ij,n}^{ve} + \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}} \right] \alpha_{ij}^{(\kappa)}(t_n) + \frac{\Delta e_{ij}^{ve}}{\Delta t} \left(\Delta t + \tau^{(\kappa)} \left(\exp\left[-\frac{\Delta t}{tau^{(\kappa)}} \right] - 1 \right) \right) \tag{67}$$

Finally, subtracting the value of internal strain at the end of the previous increment from both sides of the previous equation gives

$$\boxed{\Delta\alpha_{ij}^{(\kappa)} = \left(1 - \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right]\right)\left(e_{ij,n}^{ve} - \alpha_{ij,n}^{(\kappa)}\right) + \frac{\tau^{(\kappa)}}{\Delta t}\left(\frac{\Delta t}{\tau^{(\kappa)}} + \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right] - 1\right)\Delta e_{ij}^{ve}} \tag{68}$$

Equation 68 provides an analytical update equation for the internal (dissipative) strains for each Maxwell element, which is exact under the conditions that $\dot{e}_{ij}^{ve} = \Delta e_{ij}^{ve}/\Delta t \quad \forall\ t \in [t_n, t_{n+1}]$. Such conditions are unlikely in a strict sense; however, in may cases this is a reasonable approximation and the departure from this assumption provides the basis for the error metric used in the automatic time stepping algorithm developed in Section **??**. Using Equation 68 to update the internal strains at each increment, it is straightforward to compute the total deviatoric stress by summing Equation 61 over each Maxwell element and including the contribution from the elastic spring as

$$S_{ij} = 2G_0 e_{ij}^{ve} - \sum_{\kappa=1}^{N} 2G^{(\kappa)}\alpha_{ij}^{(\kappa)} \tag{69}$$

where the instantaneous shear modulus, G_0, has been introduced and is defined as

$$G_0 = G^{\infty} + \sum_{\kappa=1}^{N} G^{(\kappa)} \tag{70}$$

The same procedure can be repeated for the volumetric/hydrostatic terms; these details are omitted here.

D Maxwell Constitutive Tangent

The deviatoric tangent stiffness associated with the generalized Maxwell viscoelastic model is derived in this appendix using the relationships from Appendix C. The tensoral form of the viscoelastic deviatoric constitutive tangent is

$$\mathbf{C}' = \frac{\partial \Delta\mathbf{S}}{\partial \Delta\mathbf{e}^{ve}} \tag{71}$$

Using Equation 69 allows us to write a linearized form,

$$\Delta\mathbf{S} = 2G_0\Delta\mathbf{e}^{ve} - \sum_{\kappa=1}^{N} 2G^{(\kappa)}\Delta\boldsymbol{\alpha}^{(\kappa)} \tag{72}$$

substituting Equation 68 into Equation 72 gives

$$\Delta\mathbf{S} = 2G_0\Delta\mathbf{e}^{ve} - \sum_{\kappa=1}^{N} 2G^{(\kappa)}\left(\left(1-\exp\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right]\right)\left(\mathbf{e}_n^{ve}-\boldsymbol{\alpha}_n^{(\kappa)}\right)+\frac{\tau^{(\kappa)}}{\Delta t}\left(\frac{\Delta t}{\tau^{(\kappa)}}+\exp\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right]-1\right)\Delta\mathbf{e}^{ve}\right) \quad (73)$$

The deviatoric part of the viscoelastic constitutive tangent is obtained by differentiating Equation 73 with respect to $\Delta\mathbf{e}^{ve}$, i.e.,

$$\mathbf{C}' = \mathbf{I}\left[2G_0 - \sum_{\kappa=1}^{N} 2G^{(\kappa)}\frac{\tau^{(\kappa)}}{\Delta t}\left(\frac{\Delta t}{\tau^{(\kappa)}}+\exp\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right]-1\right)\right] \quad (74)$$

If we define the scalar

$$G_t = G_0 - \sum_{\kappa=1}^{N} G^{(\kappa)}\frac{\tau^{(\kappa)}}{\Delta t}\left(\frac{\Delta t}{\tau^{(\kappa)}}+\exp\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right]-1\right), \quad (75)$$

then the deviatoric viscoelastic tangent stiffness becomes

$$\frac{\partial\mathbf{S}}{\partial\mathbf{e}^{ve}} = 2G_t\mathbf{I}^{sym} \quad (76)$$

E Damage Strain Error Jacobian

The update scheme of Algorithm 2 requires a jacobian for the residual term of Equation 37 which is repeated here for convinience.

$$\mathbf{R}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3\mathbf{S}_k - \mathbf{e}_k^D \quad (77)$$

The error function implemented in the new viscoSCRAM subroutine is in terms of the viscoelastic strain instead of the damage strain, so we get

$$\mathbf{R}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3\mathbf{S}_k - \mathbf{e}_k + \mathbf{e}_k^{ve} \quad (78)$$

The incremental form of which is

$$\Delta\mathbf{R}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3\Delta\mathbf{S}_k + \frac{3}{2G_0 a}\left(\frac{c_k}{a}\right)^2\mathbf{S}_k\Delta c_k - \Delta\mathbf{e}_k + \Delta\mathbf{e}_k^{ve} \quad (79)$$

The increment in deviatoric stress is a function of the deviatoric viscoelastic tangent stiffness and the viscoelastic strain increment as shown in Equation 80

$$\Delta\mathbf{S}_k = \frac{\partial\mathbf{S}}{\partial\mathbf{e}^{ve}} : \Delta\mathbf{e}_k^{ve} \quad (80)$$

The damage increment is

$$\Delta c_k = \frac{\partial c_k}{\partial \mathbf{S}} : \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta \mathbf{e}_k^{ve} \tag{81}$$

where $\frac{\partial c_k}{\partial \mathbf{S}}$ should be calculated consistently with the damage growth algorithm. Combining Equations 79, 80, and 81 results in

$$\Delta \mathbf{R}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta \mathbf{e}_k^{ve} + \frac{3}{2G_0 a}\left(\frac{c_k}{a}\right)^2 \mathbf{S}_k \frac{\partial c}{\partial \mathbf{S}} : \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta \mathbf{e}_k^{ve} - \Delta \mathbf{e}_k + \Delta \mathbf{e}_k^{ve} \tag{82}$$

Finally the residual jacobian can be estimated as

$$\frac{\Delta \mathbf{R}_k}{\Delta \mathbf{e}_k^{ve}} = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} + \frac{3}{2G_0 a}\left(\frac{c_k}{a}\right)^2 \mathbf{S}_k \otimes \frac{\partial c}{\partial \mathbf{S}} : \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} + \mathbb{I} \tag{83}$$

Substituting Equation 76 into Equation 83 results in

$$\frac{\Delta \mathbf{R}_k}{\Delta \mathbf{e}_k^{ve}} = \left(1 + \frac{G_t}{G_0}\left(\frac{c_k}{a}\right)^3\right)\mathbb{I} + \frac{9G_t}{2G_0 a\bar{\sigma}}\left(\frac{c_k}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}} \mathbf{S}_k \otimes \mathbf{S}_k \tag{84}$$

F Embedded Runge-Kutta Method

We have incorporated an embedded Runge-Kutta method for integrating the damage kinetics equation. The particular scheme chosen is known as the Bogacki-Shampine method and was introduced in P. BOGACKI (1989). This method provides a 3rd order accuracy requiring 4 evaluations of the kinetics equations. This method also provides an estimate of the numerical error as well as a recommendation of the appropriate time increment necessary to achieve an acceptable numerical error. Higher order methods such as Dormand-Prince were considered but they contained negative weighting terms of the Butcher Table (shown for Bogacki-Shampine in Table 3) which posed some numerical difficulties associated with negative damage growth.

The updated crack size is

$$c_{k+1} = c_k + b_1^* A_1 + b_2^* A_2 + b_3^* A_3 + b_4^* A_4 \tag{85}$$

where b_i are found in the Table 3. A_i are

$$A_i = \Delta t \dot{c}\left(\bar{\sigma}_k + d_i \Delta\bar{\sigma}, P_k + d_i \Delta P, c_k + \sum_{j=1}^{i-1} a_{ij} A_j\right) \tag{86}$$

Table 3: Bogacki-Shampine Parameters for Embedded Runge-Kutta Method

i	d_i	a_{ij}				b_i	b_i^*
1	0					$\frac{2}{9}$	$\frac{7}{24}$
2	$\frac{1}{2}$	$\frac{1}{2}$				$\frac{1}{3}$	$\frac{1}{4}$
3	$\frac{3}{4}$	0	$\frac{3}{4}$			$\frac{4}{9}$	$\frac{1}{3}$
4	1	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{4}{9}$	0	0	$\frac{1}{8}$
$j =$		1	2	3	4		

where d_i and a_{ij} are also found in Table 3

$$A_2 = \Delta t \dot{c} \left(\bar{\sigma}_k + d_2 \Delta \bar{\sigma}, P_k + d_2 \Delta P, c_k + a_{21} A_1 \right) \tag{87}$$

The numerical error estimate (E_{et}) is

$$E_{est} = \sum_{i=1}^{4} \left(b_i - b_i^* \right) A_i \tag{88}$$

and you want the magnitude of the error to be less than a specified tolerance. If it is not then a new timestep size is chosen as

$$\Delta t = \Delta t \left| \frac{tol}{E_{est}} \right|^{\frac{1}{3}} \tag{89}$$

We also need the quantity $\frac{\partial c}{\partial \bar{\sigma}}$ calculated in a way consistent with this algorithm.

$$\frac{\partial c}{\partial \bar{\sigma}} = \frac{\partial (\delta c)}{\partial (\delta \bar{\sigma})} \tag{90}$$

where δc is the virtual change in c due to a virtual change in $\bar{\sigma}$ given as $\delta \bar{\sigma}$. If we consider equation 85 then

$$\delta c = \delta c_k + b_1 \delta A_1 + b_2 \delta A_2 + b_3 \delta A_3 + b_4 \delta A_4 + b_5 \delta A_5 + b_6 \delta A_6 \tag{91}$$

c_k is fixed at the beginning of the increment and A_1 is evaluated based on information that is fixed at the

beginning of the increment so they can have no virtual change. This leaves

$$\delta c = b_2\delta A_2 + b_3\delta A_3 + b_4\delta A_4 + b_5\delta A_5 + b_6\delta A_6 \tag{92}$$

then

$$\delta A_2 = \Delta t \left(\left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_* \right) \delta\bar{\sigma} + \Delta t \left(\left.\frac{\partial \dot{c}}{\partial c}\frac{\partial c}{\partial A_1}\right) \right|_* \delta A_1 \tag{93}$$

where $*$ means that it is evaluated under the same conditions as A_2 was. In the context of evaluating δA_2 we should find that $\frac{\partial c}{\partial A_1}$ is a_{21}. However we already determined that δA_1 is zero so we are left with.

$$\frac{\delta A_2}{\delta\bar{\sigma}} = \Delta t \left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_* \tag{94}$$

Evaluating the virtual change in A_4

$$\delta A_3 = \Delta t \left(\left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_* \right) \delta\bar{\sigma} + \Delta t \left(\left.\frac{\partial \dot{c}}{\partial c}\right|_* \right) a_{31}\delta A_1 + \Delta t \left(\left.\frac{\partial \dot{c}}{\partial c}\right|_* \right) a_{32}\delta A_2 \tag{95}$$

Again, δA_1 is zero and if we replace δA_2 with $\frac{\delta A_2}{\delta\bar{\sigma}}\delta\bar{\sigma}$ then we get

$$\delta A_3 = \Delta t \left[\left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_* + \left(\left.\frac{\partial \dot{c}}{\partial c}\right|_* \right) a_{32}\frac{\delta A_2}{\delta\bar{\sigma}} \right] \delta\bar{\sigma} \tag{96}$$

if we continue this development we get

$$\frac{\delta A_i}{\delta\bar{\sigma}} = \Delta t \left[\left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_* + \sum_{j=1}^{i-1} \left(a_{ij} \left.\frac{\partial \dot{c}}{\partial c}\right|_* \frac{\delta A_j}{\delta\bar{\sigma}} \right) \right] \tag{97}$$

which leads to

$$\frac{\partial c}{\partial \bar{\sigma}} = \sum_{i=1}^{4} b_i^* \frac{\delta A_i}{\delta\bar{\sigma}} \tag{98}$$

Algorithm 4 RK3, 3rd order embedded Runge-Kutta crack update

Input: `Vars` : $\boldsymbol{\sigma}_{k+1}$, $\boldsymbol{\sigma}_k$, c_k, Δt^*, `Params` : $K_0, \mu', m, v_{res}, b[:], b^*[:], d[:], a[:,:]$

Output: c_{k+1}, $\frac{\partial c}{\partial \bar{\sigma}}$, Δt_{rec}

initialize: $A[:] = 0$, $\frac{\delta A}{\delta \bar{\sigma}}[:] = 0$

$\Delta\bar{\sigma} = \bar{\sigma}_{k+1} - \bar{\sigma}_k$, $\Delta P = P_{k+1} - P_k$

for $i = 1$ **to** 6 **do**

  $\bar{\sigma}^* = \bar{\sigma}_k + d[i]\Delta\bar{\sigma}$, $P^* = P_k + d[i]\Delta P$, $c^* = c_k + \sum (a[i,:]A[:])$

  call `get_dot` $\longrightarrow \dot{c}$, $\frac{\partial \dot{c}}{\partial \bar{\sigma}}$, $\frac{\partial \dot{c}}{\partial c}$

  $A[i] = \Delta t \dot{c}$

  $\frac{\delta A}{\delta \bar{\sigma}}[i] = \Delta t \left[\frac{\partial \dot{c}}{\partial \bar{\sigma}} + \frac{\partial \dot{c}}{\partial c} \sum \left(a[i,:] \frac{\delta A}{\delta \bar{\sigma}}[:]\right)\right]$

end for

$c_{k+1} = c_k + \sum (b[:]A[:])$

$\frac{\partial c}{\partial \bar{\sigma}} = \sum \left(b[:] \frac{\delta A}{\delta \bar{\sigma}}[:]\right)$

$E_{est} = \sum (b[:]A[:] - b^*[:]A[:])$

$\Delta t_{rec} = \Delta t \left| \frac{tol}{E_{est}} \right|^{\frac{1}{5}}$

G Continuum Tangent Stiffness

We need a tangent stiffness of the form $\frac{\Delta\sigma}{\Delta\epsilon}$ and from appendix D we have $\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}}$, and $\frac{\partial p}{\partial \epsilon^{vol}} = -K$. The stress can be decomposed into deviatoric and spherical components as

$$\boldsymbol{\sigma} = \mathbf{S} - p\mathbf{i} \tag{99}$$

If we also decompose the strain into deviatoric e_{ij} and spherical ϵ^{vol} components then the tangent stiffness becomes

$$\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\epsilon}} = \left[\frac{\partial \mathbf{S}}{\partial \mathbf{e}} - \frac{\partial p}{\partial \mathbf{e}} \otimes \mathbf{i}\right] : \frac{\partial \mathbf{e}}{\partial \boldsymbol{\epsilon}} + \left[\frac{\partial \mathbf{S}}{\partial \epsilon^{vol}} - \frac{\partial p}{\partial \epsilon^{vol}} \mathbf{i}\right] \otimes \frac{\partial \epsilon^{vol}}{\partial \boldsymbol{\epsilon}} \tag{100}$$

given that

$$\mathbf{e} = \boldsymbol{\epsilon} - \frac{\epsilon_{vol}}{3}\mathbf{i} \tag{101}$$

$$\epsilon_{vol} = TR\left(\boldsymbol{\epsilon}\right) \tag{102}$$

we have

$$\frac{\partial \boldsymbol{e}}{\partial \boldsymbol{\epsilon}} = \mathbb{I} - \frac{1}{3}\mathbf{i} \otimes \mathbf{i} \tag{103}$$

and

$$\frac{\partial \epsilon^{vol}}{\partial \boldsymbol{\epsilon}} = \mathbf{i} \tag{104}$$

In viscoscram the deviatoric strain is further decomposed into viscoelastic and damage components.

$$\mathbf{e} = \mathbf{e}^{ve} + \mathbf{e}^{D} \tag{105}$$

but the deviatoric stress is only a function of the viscoelastic part so we can write an increment in stress as

$$\Delta\mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{e}^{ve} \tag{106}$$

Where $\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}}$ was defined in Appendix D

$$\Delta\mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \left(\Delta\mathbf{e} - \Delta\mathbf{e}^{D}\right) \tag{107}$$

$$\Delta\mathbf{e}^{D} = \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \Delta\mathbf{S} + \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \mathbf{S}\Delta c \tag{108}$$

$$\Delta\mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{e} - \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{S} - \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S}\Delta c \tag{109}$$

$$\left[\mathbb{I} + \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}}\right] : \Delta\mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{e} - \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S}\Delta c \tag{110}$$

$$\Delta c = \frac{\partial c}{\partial \bar{\sigma}}\frac{\partial \bar{\sigma}}{\partial \mathbf{S}} : \Delta\mathbf{S} + \frac{\partial c}{\partial p}\Delta p + \frac{\partial c}{\partial V_{res}}\frac{\partial V_{res}}{\partial \dot{\bar{e}}}\frac{\partial \dot{\bar{e}}}{\partial \dot{\mathbf{e}}} : \frac{\partial \dot{\mathbf{e}}}{\partial \Delta\mathbf{e}} : \Delta\mathbf{e} \tag{111}$$

$$\frac{\partial c}{\partial V_{res}}\frac{\partial V_{res}}{\partial \dot{\bar{e}}}\frac{\partial \dot{\bar{e}}}{\partial \dot{\mathbf{e}}} : \frac{\partial \dot{\mathbf{e}}}{\partial \Delta \mathbf{e}} = \frac{2V_{res}A}{3\Delta t \dot{\bar{e}}^2}\frac{\partial c}{\partial V_{res}}\dot{\mathbf{e}} \tag{112}$$

$$\left[\mathbb{I} + \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}}\right] : \Delta \mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta \mathbf{e} - \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S}\left(\frac{\partial c}{\partial \bar{\sigma}}\frac{\partial \bar{\sigma}}{\partial \mathbf{S}} : \Delta \mathbf{S} + \frac{\partial c}{\partial p}\Delta p + \frac{2V_{res}A}{3\Delta t \dot{\bar{e}}^2}\frac{\partial c}{\partial V_{res}}\dot{\mathbf{e}} : \Delta \mathbf{e}\right) \tag{113}$$

$$\frac{\partial \bar{\sigma}}{\partial \mathbf{S}} = \frac{3\mathbf{S}}{2\bar{\sigma}} \tag{114}$$

$$\left[\mathbb{I} + \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} + \frac{9}{4G_0 a\bar{\sigma}}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}}\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S} \otimes \mathbf{S}\right] : \Delta \mathbf{S} = \tag{115}$$

$$\left[\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} - \frac{V_{res}A}{G_0 a \Delta t \dot{\bar{e}}^2}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial V_{res}}\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S} \otimes \dot{\mathbf{e}}\right] : \Delta \mathbf{e} - \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S}\left(\frac{\partial c}{\partial p}\frac{\partial p}{\partial \epsilon_{vol}}\Delta \epsilon_{vol}\right) \tag{116}$$

If this is all right then we are left with

$$\frac{\Delta \mathbf{S}}{\Delta \mathbf{e}} = \left[\mathbb{I} + \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} + \frac{9}{4G_0 a\bar{\sigma}}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}}\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S} \otimes \mathbf{S}\right]^{-1} : \left[\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} - \frac{V_{res}A}{G_0 a \Delta t \dot{\bar{e}}^2}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial V_{res}}\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S} \otimes \dot{\mathbf{e}}\right] \tag{117}$$

From before we can write

$$\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} = 2G_t \mathbb{I} \tag{118}$$

Then

$$\frac{\Delta \mathbf{S}}{\Delta \mathbf{e}} = \left[\left(1 + \frac{G_t}{G_0}\left(\frac{c}{a}\right)^3\right)\mathbb{I} + \frac{9G_t}{2G_0 a\bar{\sigma}}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}}\mathbf{S} \otimes \mathbf{S}\right]^{-1} : \left[2G_t\mathbb{I} - \frac{2G_t V_{res}A}{G_0 a \Delta t \dot{\bar{e}}^2}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial V_{res}}\mathbf{S} \otimes \dot{\mathbf{e}}\right] \tag{119}$$

Additionally, we will need

$$\frac{\Delta \mathbf{S}}{\Delta \epsilon_{vol}} = \frac{\Delta \mathbf{S}}{\Delta p}\frac{\Delta p}{\Delta \epsilon_{vol}} \tag{120}$$

which in this case the bulk behavior is linear elastic so

$$\frac{\Delta \mathbf{S}}{\Delta \epsilon_{vol}} = -K \frac{\Delta \mathbf{S}}{\Delta p} \tag{121}$$

From above we see that (the direct notation probably needs some work)

$$\frac{\Delta \mathbf{S}}{\Delta \epsilon_{vol}} = \left[\left(1 + \frac{G_t}{G_0} \left(\frac{c}{a} \right)^3 \right) \mathbb{I} + \frac{9 G_t}{2 G_0 a \bar{\sigma}} \left(\frac{c}{a} \right)^2 \frac{\partial c}{\partial \bar{\sigma}} \mathbf{S} \otimes \mathbf{S} \right]^{-1} : \left(\frac{3 K G_t}{G_0 a} \left(\frac{c}{a} \right)^2 \frac{\partial c}{\partial p} \mathbf{S} \right) \tag{122}$$

Finally, in viscoscram the pressure has no direct dependance on the deviatoric strain so $\frac{\partial p}{\partial e_{kl}} = 0$

Apply the 4th order corollary to the Sherman-Morrison formula (Cite Brannon or others)

$$A_1 = 1 + \frac{G_t}{G_0} \left(\frac{c}{a} \right)^3 \tag{123}$$

$$A_2 = \frac{9 G_t}{2 G_0 a \bar{\sigma}} \left(\frac{c}{a} \right)^2 \frac{\partial c}{\partial \bar{\sigma}} \tag{124}$$

$$[A_1 \mathbb{I} + A_2 \mathbf{S} \otimes \mathbf{S}]^{-1} = \frac{1}{A_1} \mathbb{I} - \frac{A_2 \mathbf{S} \otimes \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \tag{125}$$

$$A_3 = \frac{2 G_t V_{res} A}{G_0 a \Delta t \dot{\bar{e}}^2} \left(\frac{c}{a} \right)^2 \frac{\partial c}{\partial V_{res}} \tag{126}$$

$$A_4 = \frac{3 K G_t}{G_0 a} \left(\frac{c}{a} \right)^2 \frac{\partial c}{\partial p} \tag{127}$$

$$\frac{\Delta \mathbf{S}}{\Delta \mathbf{e}} = \left[\frac{1}{A_1} \mathbb{I} - \frac{A_2 \mathbf{S} \otimes \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \right] : [2 G_t \mathbb{I} - A_3 \mathbf{S} \otimes \dot{\mathbf{e}}] \tag{128}$$

$$\frac{\Delta \mathbf{S}}{\Delta \mathbf{e}} = \frac{2 G_t}{A_1} \mathbb{I} + \left[\frac{A_2 A_3 \mathbf{S} : \mathbf{S}}{A_1^2 + A1 A2 \mathbf{S} : \mathbf{S}} - \frac{A_3}{A_1} \right] \mathbf{S} \otimes \dot{\mathbf{e}} - \frac{2 G_t A_2}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \mathbf{S} \otimes \mathbf{S} \tag{129}$$

$$\frac{\Delta \mathbf{S}}{\Delta \epsilon_{vol}} = \left[\frac{A_4}{A_1} - \frac{A_2 A_4 \mathbf{S} : \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \right] \mathbf{S} \tag{130}$$

$$\frac{\Delta \sigma}{\Delta \epsilon} = \frac{2 G_t}{A_1} \mathbf{P}^{dev} + \left[\frac{A_2 A_3 \mathbf{S} : \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} - \frac{A_3}{A_1} \right] \mathbf{S} \otimes \dot{\mathbf{e}} - \frac{2 G_t A_2}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \mathbf{S} \otimes \mathbf{S} + \left[\frac{A_4}{A_1} - \frac{A_2 A_4 \mathbf{S} : \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \right] \mathbf{S} \otimes \mathbf{i} + K \mathbf{i} \otimes \mathbf{i} \tag{131}$$

H Parameters used in comparison of new and old formulation

The parameters are as follows

K	=	3460.0 MPa	G_1	=	109.0 MPa	τ_1	=	$1.0E+3$ s
G_∞	=	404.0 MPa	G_2	=	108.0 MPa	τ_2	=	$1.0E+2$ s
K_0	=	0.03 MPa$\sqrt{mm}$	G_3	=	139.0 MPa	τ_3	=	$1.0E+1$ s
μ'	=	1.159	G_4	=	170.0 MPa	τ_4	=	$1.0E+0$ s
m	=	10.0	G_5	=	213.0 MPa	τ_5	=	$1.0E-1$ s
a	=	1.0 mm	G_6	=	267.0 MPa	τ_6	=	$1.0E-2$ s
V_{max}	=	3.0E+5 $\frac{mm}{s}$	G_7	=	341.0 MPa	τ_7	=	$1.0E-3$ s
V_a	=	0.89164	G_8	=	434.0 MPa	τ_8	=	$1.0E-4$ s
V_b	=	2.28	G_9	=	581.0 MPa	τ_9	=	$1.0E-5$ s
c_1	=	15.679	G_{10}	=	726.0 MPa	τ_{10}	=	$1.0E-6$ s
c_2	=	199.375 kelvin						
T_{ref}	=	303.13 kelvin						

I Parameters used in comparison to experimental data

The parameters are as follows

K	=	5997 MPa	G_1	=	4000 MPa	τ_1	=	$1.0E-3$ s
G_∞	=	50.0 MPa	G_2	=	1000 MPa	τ_2	=	$1.0E-2$ s
K_0	=	$8.25E-5$ MPa$\sqrt{mm}$	G_3	=	500 MPa	τ_3	=	0.2152 s
μ'	=	2.55	G_4	=	400 MPa	τ_4	=	1.1 s
m	=	10.0	G_5	=	300 MPa	τ_5	=	6.3 s
a	=	$3.68E-4$ mm	G_6	=	225 MPa	τ_6	=	41.1 s
V_{max}	=	3.0E+5 $\frac{mm}{s}$	G_7	=	150 MPa	τ_7	=	272.3 s
V_a	=	.934	G_8	=	150 MPa	τ_8	=	1602 s
V_b	=	-1.1	G_9	=	50 MPa	τ_9	=	8627 s
c_1	=	15.679	G_{10}	=	50 MPa	τ_{10}	=	$4.83E+4$ s
c_2	=	199.375 kelvin	G_{11}	=	50 MPa	τ_{10}	=	$4.026E+5$ s
T_{ref}	=	303.13 kelvin						