INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2014; **99**:54–78
Published online 14 April 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme.4672

A semi-implicit integration scheme for a combined viscoelastic-damage model of plastic bonded explosives

Miles A. Buechler and Darby J. Luscher*,†

Los Alamos National Laboratory, Los Alamos, NM 87545, USA

SUMMARY

This paper presents a new implementation of a constitutive model commonly used to represent plastic bonded explosives in finite element simulations of thermomechanical response. The constitutive model, viscoSCRAM, combines linear viscoelasticity with isotropic damage evolution. The original implementation was focused on short duration transient events; thus, an explicit update scheme was used. For longer duration simulations that employ significantly 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 stresses are in agreement with implicit update equations at the end of each increment. The crack growth is advanced in time using a sub-incremental explicit scheme; thus, the entire implementation is semi-implicit. The theory is briefly discussed along with previous explicit integration schemes. The new integration algorithm and its implementation into the finite element code, Abaqus, are detailed. Finally, the new and old algorithms are compared via simulations of uniaxial compression and beam bending. The semi-implicit scheme has been demonstrated to provide higher accuracy for a given allocated computational time for the quasistatic cases considered here. Published 2014. This article is a US Government work and is in the public domain in the USA.

Received 15 April 2013; Revised 2 December 2013; Accepted 18 March 2014

KEY WORDS: viscoSCRAM; damage; PBX-9501; plastic bonded explosive; semi-implicit

1. INTRODUCTION

This paper details a new implementation of a constitutive model commonly 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 wt.% high-melting explosive (HMX) crystals bonded together by 2.5 wt.% elastomer binder (estane) and 2.5 wt.% 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 and its early implementation. Indeed, viscoSCRAM, or variations

*Correspondence to: Darby J. Luscher, Los Alamos National Laboratory, Los Alamos, NM 87545, USA.
†E-mail: djl@lanl.gov

of the theory, is used to model plastic bonded explosives over a range of thermomechanical conditions [1–9].

Originally, viscoSCRAM was developed to model stress–strain response over strain rates ranging from approximately 100 to $10,000\mathrm{s}^{-1}$ for short duration events resulting from mechanical impact [1, 6, 10, 11]. Lagrange finite element codes (e.g., Abaqus Explicit, EPIC) employing explicit time integration of the equations of motion are typically used to solve this class of problems. 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 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, presents 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-order explicit method over several sub-increments of the full step.

From a more general perspective, coupled viscoelastic and damage models (in some cases including viscoplasticity) have broad application ranging from filled rubber materials [12, 13], biological tissues [14], polymeric composites [15], granular materials such as asphalts [16, 17], and energetic composites such as explosives [1, 2] and propellants [18] to name a few. It is beyond the scope of this paper to review the broad area of coupled damage-viscoelasticity; instead, we provide here a brief synopsis of the numerical implementation of various theories that have relevance to viscoSCRAM. For example, Peña *et al.* [14] develop a finite-strain viscoelastic-damage model for application to biological tissues within the framework of internal state variable theory. They implement the theory using a 'recursive' update procedure that benefits from the use of a damage evolution threshold surface within strain space, thus obviating an iterative procedure to solve nonlinear coupled equations at the constitutive level. Yang *et al.* [15] develop a micromechanics-based viscoelastic-damage model that combines Boltzmann superposition in the Laplace domain for modeling viscoelasticity with a damage model whose evolution is based on an assumed Weibull distribution of the intrinsic resistance to damage. Their model focuses on particle-reinforced composites comprising a relatively low volume fraction of spherical elastic particles embedded within a vislocelastic matrix. Darabi *et al.* [16] cast a viscoelastic-viscoplastic-damage model within an extended continuum damage mechanics framework and take advantage of the so-called healing configuration to decouple the constitutive update of viscoelastoplasticity from that of damage. The stress in the healing configuration is unaffected by damage; thus, the constitutive integration can be conducted using recursive-iterative Newton Raphson scheme with recourse to closest point projection or radial return for the viscous trajectory back to a viscoplastic yield surface. On the other hand, the strain increment in the healing configuration is not known at the beginning of the time step leading to an iterative nonlinear solution scheme to couple the two otherwise independently updated parts of the model. Thus, the implementation of their model manifests as a nesting of two nonlinear solvers, the innermost solver determining the updated stress in the healing configuration given an increment in healing strain and the outer solver determining the increment of strain in the healing configuration. Zhu and Sun [17] combine generalized Maxwell viscoelastic model with Drucker–Prager non-associated viscoplasticity and continuum damage mechanics within the ISO framework. Their implementation exploits the backward difference solvers for stiff ODEs available in the numerical software, Matlab (TM). Consequently, their implementation is limited to investigation of specific cases that do not necessitate integration within a finite element code to solve general thermomechanical boundary value problems. Xu *et al.* [18] develop an elastic viscoplastic damage model through application of micromechanical homogenization to representative volume elements of polymer bonded particulate composites. Their implementation combines an explicit hypoelastic update relating stress to the accumulation of elastic deformation and a backstress associated with viscoplasticity, with an explicit forward Euler scheme for integration of the damage evolution equations and an implicit backward Euler scheme for integrating the flow rule and the evolution of the partition of strain into binder, particle, and damage parts. Because of the implicit update of the flow rule and 'matrix' strain, their implementation utilizes a nesting of two nonlinear Newton solvers.

The paper is organized as follows. Section 2 gives a brief history of viscoSCRAM theory and lays out the constitutive equations. Section 3 discusses one common constitutive update algorithm and its limitations. A new semi-implicit constitutive update scheme developed in this work is presented in Section 4. Results from example simulations used to compare performance of the new implementation with the previous implementation are presented in Section 6.

Indicial notation is routinely applied throughout for clarity. However, 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, iteration, and element of concern. In direct notation a tensor is represented in bold, for example, $\mathbf{A}$, while in indicial notation, the coefficients comprising the tensor (resolved against a fixed Cartesian basis) are indicated, for example, 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; that is, summation over three Cartesian indices is implied by repeated indices. For example, $\mathtt{Trace}\,[\mathbf{A}] = A_{kk} = A_{11} + A_{22} + A_{33}$. The fourth-order identity tensor is represented in direct notation by $\mathbb{I}$ and indicial notation as $\frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})$. The second-order identity tensor is represented in direct notation as $\mathbf{I}$ and in indicial notation by coefficients of the Kronecker delta, $\delta_{ij} = 1$ for $i = j$ and 0 otherwise. The scalar product between two second-order tensors is denoted by the operator :, that is, $\mathbf{A} : \mathbf{B} = A_{ij}B_{ij}$.

2. VISCOSCRAM: BRIEF HISTORY AND THEORY

A brief history of this model is discussed here in order to describe the original intentions for the theory and 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. The rate process of microcracking damage manifests as a degradation of the stress carrying ability of the composite. Viscoelasticity of the binder is represented by a generalized Maxwell model, and the breakage of bonds within the binder, between binder and particulates, and within the 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 [19] 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 elasticity, deformation due to opening of existing cracks, shearing of closed cracks, and subsequent growth of cracks. The 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, see Dienes *et al.* [11].

Addessio and Johnson [10] simplified the original SCRAM theory into an isotropic theory (ISOSCRAM) 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 whose 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 [10] 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 material's Raleigh wave speed.

Based on elements of the original SCRAM theory and ISO-SCRAM, Bennett *et al.* [1] and Hackett and Bennett [2] 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 [11] than the damage surface-based kinetics of Addessio and Johnson [10]. It should be noted that viscoSCRAM does not include mechanisms for shear dilatancy, which 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 [11].

Kinematics

The total strain tensor, $\epsilon_{ij}^{\text{tot}}$, reflects contributions from 'mechanical' strain and thermal expansion following the traditional linear thermoelastic relationship

$$\epsilon_{ij}^{\text{tot}} = \epsilon_{ij} + \alpha(T - T_0)\delta_{ij} \tag{1}$$

where ϵ_{ij} is the mechanical strain tensor, α is an isotropic linear coefficient of thermal expansion, and T and T_0 are the current and reference temperatures, respectively. Small mechanical strains, ϵ_{ij}, are assumed, and the mechanical strain tensor is decomposed into volumetric and deviatoric parts according to

$$\epsilon_{ij} = e_{ij} + \frac{1}{3}\epsilon_{kk}\delta_{ij} \tag{2}$$

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

$$e_{ij} = e_{ij}^{ve} + e_{ij}^{D} \tag{3}$$

where e^{ve} is the viscoelastic component of the deviatoric strain and e^{D} is the portion of the deviatoric strain because of 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, that is,

$$\sigma_{ij} = S_{ij} + \delta_{ij}\sigma^{m} \tag{4}$$

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 bulk response, that is,

$$\sigma^{m} = K\epsilon_{kk} \tag{5}$$

where K is the bulk modulus. Note that (1) some implementations favor using a separate viscoelastic model for the bulk response, and (2) it is conceivable that one may replace this relationship with a more general pressure-volume-temperature equation-of-state [6]. 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

$$\dot{S}_{ij} = 2G^{\infty}\dot{e}^{ve}_{ij} + \sum_{\kappa=1}^{N}\left(2G^{(\kappa)}\dot{e}^{ve}_{ij} - \frac{S^{(\kappa)}_{ij}}{\tau^{(\kappa)}}\right) \tag{6}$$

where G^{∞} is the steady-state shear modulus, and $G^{(\kappa)}$, $\tau^{(\kappa)}$, $S^{(\kappa)}_{ij}$ are the shear modulus, relaxation time, and deviatoric stresses associated with the κ^{th} (of N) Maxwell element [20]. The damage strain is related to the deviatoric stress through the damage compliance relationship

$$e^{D}_{ij} = \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 S_{ij} \tag{7}$$

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 [10].

Kinetics of damage growth

Crack growth is based on the phenomenology of brittle fracture mechanics. There are two distinct regions of crack growth rate, that is, stable and unstable growth (cf. Dienes *et al.* [11]). Accordingly, the crack growth rate depends upon an effective stress intensity [1, 2, 11, 21],

$$K_I = \bar{\sigma}\sqrt{\pi c} \tag{8}$$

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

$$\bar{\sigma} = \begin{cases} \left(\frac{3}{2}S_{ij}S_{ij}\right)^{\frac{1}{2}}, & \text{for } \sigma_m < 0 \quad \text{(9a)} \\ \left(\frac{3}{2}\sigma_{ij}\sigma_{ij}\right)^{\frac{1}{2}}, & \text{otherwise} \quad \text{(9b)} \end{cases}$$

The crack growth rate is

$$\dot{c} = \begin{cases} v_{res}\left(\frac{K_I}{K_1}\right)^{m}, & \text{for } K_I < K' \quad \text{(10a)} \\ v_{res}\left[1 - \left(\frac{K_{0\mu}}{K_I}\right)^2\right], & \text{otherwise} \quad \text{(10b)} \end{cases}$$

where v_{res} is obtained from an empirical equation $\log_{10}(v_{res}) = v_a \log_{10}\left(\dot{\bar{\epsilon}}\right) + v_b$. Typically v_{res} is limited by the theoretical terminal crack speed, v_{max}, (in principle, the Raleigh wave speed). The exponent m is simply a model parameter. The two distinct regimes of crack growth correspond to stable (Equation (10a)) and unstable (Equation (10b)) crack growth, respectively. The stress intensity at which the behavior transitions from Equation (10a) to Equation (10b) is defined as

$$K' = K_{0\mu}\left(1 + \frac{2}{m}\right)^{1/2} \tag{11}$$

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. In the original model [2], the frictional threshold stress intensity is computed by

$$K_{0\mu} = K_0\left[1 - \frac{\pi\mu'\sigma_m\sqrt{c}}{K_0}\left(1 - \frac{\mu'\sigma_m\sqrt{c}}{K_0}\right)\right]^{\frac{1}{2}} \tag{12}$$

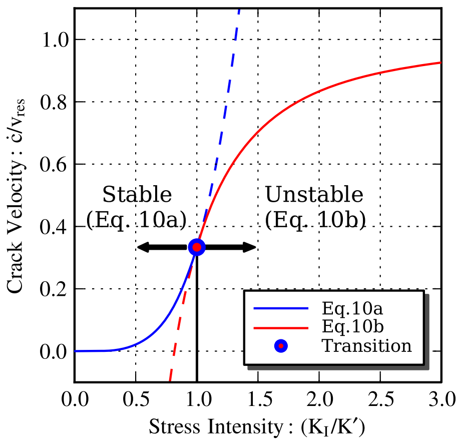


Figure 1. Relationship between normalized crack velocity $\dot{c}/v_{res}$ and normalized stress intensity K_I/K' illustrating transition between stable and unstable crack growth.

where μ' is a friction coefficient and K_0 is a model parameter representing the frictionless threshold stress intensity of the material. This 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 (12)). Such behavior is physically consistent with the concept of an internal frictional resistance to damage evolution under compression [1, 2] and is consistent with experimental data. However, the frictional dependence is symmetric about zero mean stress, that is, the same under tension and compression. Consequently, frictional interaction under tensile loading is inconsistent with the phenomenology leading to Equation (12). 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, $\sigma_m > 0$. Instead, here, we address this issue through a simple modification of Equation (12) made by replacing σ_m with $- < p >$ where the thermodynamic pressure $p = -\sigma_m$ and $< \bullet >$ are Macaulay brackets that return the argument for values greater than zero, otherwise zero. Thus,

$$K_{0\mu} = K_0 \left[1 + \frac{\pi \mu' < p > \sqrt{c}}{K_0} \left(1 + \frac{\mu' < p > \sqrt{c}}{K_0} \right) \right]^{\frac{1}{2}} \tag{13}$$

The normalizing stress intensity in the denominator of Equation (10a) is defined as

$$K_1 = K' \left(1 + \frac{m}{2} \right)^{1/m} \tag{14}$$

Figure 1 illustrates the transition from stable to unstable crack growth kinetics based on Equations (10a) and (10b). Dependence of the transition threshold stress intensity, K', on the current mean crack size and mean stress presents numerical difficulties for implementing a fully implicit update scheme. As discussed in Section 4, these difficulties motivate the semi-implicit scheme developed in this work.

Note that, according to Hackett and Bennett [2], the effective frictional coefficient, μ', is related to the internal static coefficient of friction, μ_s, by

$$\mu' = \left[\frac{45}{2\left(3 - 2\mu_s^2\right)} \right]^{\frac{1}{2}} \mu_s \tag{15}$$

However, given the level of empiricism involved with determining μ_s, it is justifiable to instead specify μ' directly. The model parameters are summarized in Table I.

Finally, we note a few typographical errors appearing in Ref.[2] with the intention of clarifying differences.

Table I. 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	$\mathrm{FL}^{-3/2}$
μ'	Material frictional parameter	–
m	Crack growth kinetics shape parameter	–
v_{max}	Maximum crack speed	Lt^{-1}
α	Isotropic thermal expansion coefficient	T^{-1}

- K' erroneously appears in place of K_1 in Equation (10a).
- K_1 erroneously appears in place of K_I in Equation (10b).
- The power $\frac{1}{2}$ was erroneously left off in Equation (15).

3. HISTORICAL IMPLEMENTATION

Preceding the discussion of the new semi-implicit integration scheme in the next section, this section outlines the implementation developed in [2]. Differentiating Equation (7) with respect to time results in

$$\dot{\boldsymbol{e}}^D = \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \dot{\boldsymbol{S}} + \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \dot{c}\boldsymbol{S} \tag{16}$$

Substituting the time derivative of Equation (3) along with Equation (16) into Equation (6) gives

$$\dot{\boldsymbol{S}} = 2G_0\dot{\boldsymbol{e}} - 3\left(\frac{c}{a}\right)^2 \frac{\dot{c}}{a}\boldsymbol{S} - \left(\frac{c}{a}\right)^3 \dot{\boldsymbol{S}} - \sum_{\kappa=1}^{N}\left(\frac{\boldsymbol{S}^{(\kappa)}}{\tau^{(\kappa)}}\right) \tag{17}$$

which is rearranged as

$$\dot{\boldsymbol{S}} = \psi(c)\dot{\boldsymbol{e}} - \theta(c,\dot{c})\left[\boldsymbol{S} - \boldsymbol{\lambda}\left(c, \boldsymbol{S}^{(\kappa)}\right)\right] \tag{18}$$

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

$$\psi(c) = \frac{2G_0}{1+\left(\frac{c}{a}\right)^3} \tag{19}$$

and

$$\theta(c,\dot{c}) = \frac{3\left(\frac{c}{a}\right)^2 \frac{\dot{c}}{a}}{1+\left(\frac{c}{a}\right)^3} \tag{20}$$

respectively. Additionally, the *state-dependent* tensor-valued function, $\boldsymbol{\lambda}$, is defined as

$$\boldsymbol{\lambda}\left(c, \boldsymbol{S}^{(\kappa)}\right) = \frac{\sum_{\kappa=1}^{N}\left(\frac{\boldsymbol{S}^{(\kappa)}}{\tau^{(\kappa)}}\right)}{3\left(\frac{c}{a}\right)^2 \frac{\dot{c}}{a}} \tag{21}$$

It is noteworthy that the state dependence of the functions appearing in Equation (18) cause it to be a nonlinear first-order differential equation, because the non-constant coefficients depend upon the solution. Hackett and Bennett propose a central differencing scheme to integrate the differential equation [2]. However, the state dependence of the coefficient functions in Equation (18) 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

$$\dot{A}_{n+\frac{1}{2}} = \frac{A_{n+1} - A_n}{\Delta t} \tag{22}$$

Making use of Equation (22) enables us to write a discrete form of Equation (18) evaluated at the middle of the current increment, that is,

$$\theta_{n+\frac{1}{2}} \mathbf{S}_{n+\frac{1}{2}} + \frac{\mathbf{S}_{n+1} - \mathbf{S}_n}{\Delta t} = \psi_{n+\frac{1}{2}} \frac{\mathbf{e}_{n+1} - \mathbf{e}_n}{\Delta t} - \theta_{n+\frac{1}{2}} \boldsymbol{\lambda}_{n+\frac{1}{2}} \tag{23}$$

Hackett and Bennett make the reasonable approximation [2],

$$\mathbf{S}_{n+\frac{1}{2}} \approx \frac{1}{2}(\mathbf{S}_{n+1} + \mathbf{S}_n) \tag{24}$$

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 (23) to be solved for the updated stresses, that is,

$$\mathbf{S}_{n+1} = \mathbf{S}_n + \frac{\psi_\star}{\left(1 + \frac{\Delta t}{2}\theta_\star\right)} \Delta \mathbf{e} - \frac{\theta_\star \Delta t}{\left(1 + \frac{\Delta t}{2}\theta_\star\right)} (\boldsymbol{\lambda}_\star + \mathbf{S}_n) \tag{25}$$

where $\Delta \mathbf{e} = \mathbf{e}_{n+1} - \mathbf{e}_n$. Equation (25) 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 $\mathbf{S}_{n+1}$. However, Hackett and Bennett [2] do not perform an implicit solution of Equation (25). Instead, their implementation makes the approximations

$$\boldsymbol{\lambda}_\star = \boldsymbol{\lambda}_n \qquad \psi_\star = \psi_n \qquad \theta_\star = \theta_n \tag{26}$$

so that Equation (25) becomes the purely *explicit* update

$$\mathbf{S}_{n+1} = \mathbf{S}_n + \frac{\psi_n}{\left(1 + \frac{\Delta t}{2}\theta_n\right)} \Delta \mathbf{e} - \frac{\theta_n \Delta t}{\left(1 + \frac{\Delta t}{2}\theta_n\right)} (\boldsymbol{\lambda}_n + \mathbf{S}_n) \tag{27}$$

Updates to the mean crack size are provided by *explicit* time integration of Equation (10a) (or (10b)), that is,

$$c_{n+1} = c_n + \dot{c}_n \, \Delta t \tag{28}$$

The time rate of change of stress in each Maxwell element is given by combining the time derivative of Equation (3) with Equation (16), and the argument of the summation in Equation (6), resulting in

$$\dot{\boldsymbol{S}}^{(\kappa)} = 2G^{(\kappa)}\dot{\boldsymbol{e}} - \frac{\boldsymbol{S}^{(\kappa)}}{\tau^{(\kappa)}} - \frac{G^{(\kappa)}}{G^0}\left[\frac{3}{a}\left(\frac{c}{a}\right)^2 \dot{c}\boldsymbol{S} + \left(\frac{c}{a}\right)^3 \dot{\boldsymbol{S}}\right] \tag{29}$$

Hackett and Bennett [2] mention that they integrated Equation (29) using a variety of methods, although the legacy implementation uses

$$\Delta\mathbf{S}^{(\kappa)} = 2G^{(\kappa)}\Delta\mathbf{e} - \frac{\Delta t\mathbf{S}_n^{(\kappa)}}{\tau^{(\kappa)}} - \frac{G^{(\kappa)}}{G_0}\left[\frac{3}{a}\left(\frac{c_{n+1}}{a}\right)^2 \Delta c\,\mathbf{S}_{n+1} + \left(\frac{c_{n+1}}{a}\right)^3 (\mathbf{S}_{n+1} - \mathbf{S}_n)\right] \tag{30}$$

4. IMPLEMENTATION OF A SEMI-IMPLICIT INTEGRATION SCHEME

Our semi-implicit constitutive update scheme treats the decomposition of strain (Equation (3)), viscoelastic stress update (Equation (6)), and the final damaged state in an implicit manner (Equation (7)). The resulting nonlinear system of equations is solved using a Newton iterative scheme. 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 method to integrate the crack growth kinetics over potentially several sub-increments of a full time step. Accordingly, the total update scheme is semi-implicit, that is, 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 (3); the relationship between stress and viscoelastic strain, Equation (6); the relationship between stress, average crack length, and damage strain, Equation (7); and the kinetics governing crack size evolution, Equations (10a) and (10b). Attention is focused first on the discrete form of Equations (3), (6), and (7). 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, $t = t_{n+1}$. Again, the full 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, $A_k \rightarrow A_{n+1}$ 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 (6)) is summarized in Algorithm 1 (cf. [20, 22]), which is exact under the condition that the viscoelastic strain rate is constant over the step, $\dot{e}_{ij}^{ve} = \Delta e_{ij}^{ve}/\Delta t \;\forall\, t \in [t_n, t_{n+1}]$.

Algorithm 1 Maxwell

```
Input: VARS : $\Delta\mathbf{e}^{ve}, \Delta t, \mathbf{e}_n^{ve}, \boldsymbol{\alpha}_n^{(\kappa)}$    PARAMS : $G_0, G^{(\kappa)}, \tau^{(\kappa)} \quad \forall\, \kappa \in [1, N]$
Output: $\mathbf{S}_{n+1}$ , $\boldsymbol{\alpha}_{n+1}^{(\kappa)}$
  $\Delta\boldsymbol{\alpha}^{(\kappa)} = \left(1 - \exp\left[\frac{-\Delta t}{\tau^{\kappa}}\right]\right)\left(\mathbf{e}_n^{ve} - \boldsymbol{\alpha}_n^{(\kappa)}\right) + \frac{\tau^{(\kappa)}}{\Delta t}\left(\frac{\Delta t}{\tau^{(\kappa)}} + \mathtt{exp}\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right] - 1\right)\Delta\mathbf{e}^{ve}$
  $\boldsymbol{\alpha}_{n+1}^{(\kappa)} = \boldsymbol{\alpha}_n^{(\kappa)} + \Delta\boldsymbol{\alpha}^{(\kappa)}$
  $\mathbf{S}_{n+1} = G_0\left(\mathbf{e}_{n+1} - \sum_{\kappa=1}^{N}\frac{G^{(\kappa)}}{G_0}\boldsymbol{\alpha}_{n+1}^{(\kappa)}\right)$
```

Next, we write Equations (3), (6), and (7) in a discrete form at time $t = t_{n+1}$,

$$\mathbf{e}_{n+1}^{D} = \frac{1}{2G_0}\left(\frac{c_{n+1}}{a}\right)^3 \mathbf{S}_{n+1} \tag{31}$$

$$\mathbf{S}_{n+1} = \mathtt{Maxwell}\left(\mathbf{e}_{n+1}^{ve}, \mathcal{S}_n\right) \tag{32}$$

$$\mathbf{e}_{n+1}^{ve} = \mathbf{e}_{n+1} - \mathbf{e}_{n+1}^{D} \tag{33}$$

where $\mathcal{S}_n = \left(\boldsymbol{\sigma}_n, \mathbf{e}_n^{ve}, \mathbf{e}_n^D, \boldsymbol{\alpha}_n^{(\kappa)}, c_n\right)$ is the state of the material at the beginning of the increment. Substitution of Equations (31) and (33) into Equation (32) produces an expression for the implicit update of stress, that is,

$$\mathbf{S}_{n+1} = \texttt{Maxwell}\left(\left[\mathbf{e}_{n+1} - \frac{1}{2G_0}\left(\frac{c_{n+1}}{a}\right)^3 \mathbf{S}_{n+1}\right], \mathcal{S}_n\right) \tag{34}$$

While it is the algorithmic objective to solve this nonlinear system of equations, Equation (34) 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 (31), accordingly

$$\mathbf{e}_k^D \approx \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \mathbf{S}_k \tag{35}$$

which enables the definition of a residual error term, that is,

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

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 $\Delta\mathbf{e}$, respectively, (2) the state of the material at the beginning of the increment, $\mathcal{S}_n = \left(\boldsymbol{\sigma}_n, \mathbf{e}_n^{ve}, \mathbf{e}_n^D, \boldsymbol{\alpha}_n^{(\kappa)}, c_n\right)$, and (3) the material model parameters listed in Table IV. 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, $\mathcal{S}_{n+1} = \left(\mathbf{e}_{n+1}^{ve}, \mathbf{e}_{n+1}^D, \boldsymbol{\alpha}_{n+1}^{(\kappa)}, c_{n+1}\right)$. Algorithm 2 begins at Step 1 by initializing the iterate of the solution variable, that is, $\Delta\mathbf{e}^D = \mathbf{0}$ and setting the iteration counter $k = 0$. Steps 2–9 comprise a loop, which is repeated until a norm of the residual is below an acceptable error tolerance or the number of iterations exceeds a maximum allowable limit. In the latter case, the current global finite element solution increment is aborted and repeated with a smaller time step size. Within the iterative 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 Algorithm 3 (UpdateCrack, whose details are addressed below) to update the average crack size consistent with Equations (10a) and (10b). 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, $\mathbb{J}$, used for

Algorithm 2 Semi-Implicit Update Scheme

```
Input: VARS : $\Delta\mathbf{e}, \Delta t, \boldsymbol{\sigma}_n, \mathbf{e}_n^{ve}, \mathbf{e}_n^D, \boldsymbol{\alpha}_n^{(\kappa)}, c_n$     PARAMS : see Table IV
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:     $\Delta\mathbf{e}^{ve} = \Delta\mathbf{e} - \Delta\mathbf{e}^D$,   $\mathbf{e}_k^D = \mathbf{e}_n^D + \Delta\mathbf{e}^D$
 4:     call Maxwell $\longrightarrow \mathbf{S}_k$ , $\boldsymbol{\alpha}_k^{(\kappa)}$
 5:     call CrackUpdate $\longrightarrow c_k$
 6:     compute residual   $\mathbf{R}_k = \frac{1}{2G_0}\left(\frac{c_k}{a}\right)^3 \mathbf{S}_k - \mathbf{e}_k^D$
 7:     compute Jacobian   $\mathbb{J}_k$, Equation (47)
 8:     update $\Delta\mathbf{e}^D \leftarrow \Delta\mathbf{e}^D - \mathbb{J}_k^{-1} : \mathbf{R}_k$
 9: until $(\mathbf{R}_k : \mathbf{R}_k) <$ TOL   $(k \leftarrow k + 1)$
10: update:  $\mathbf{S}_{n+1} = \mathbf{S}_k, \mathbf{e}_{n+1}^D = \mathbf{e}_k^D, \mathbf{e}_{n+1}^{ve} = \mathbf{e}_n^{ve} + \Delta\mathbf{e}^{ve}, \boldsymbol{\alpha}_{n+1}^{(\kappa)} = \boldsymbol{\alpha}_k^{(\kappa)}, c_{n+1} = c_k$
```

a full Newton method to solve Equation (36) is derived subsequently. 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.

Explicit sub-incremental integration of crack growth kinetics

The integration of crack growth rate Equations (10a) and (10b) is performed using the sub-increment explicit integration scheme detailed 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 (10a) and (10b) is 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 tolerance 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 third-order embedded Runge–Kutta algorithm detailed in Algorithm 4. The arguments provided as input to Algorithm 3 are (1) the stresses at the end of the current (previous) time increment, $\boldsymbol{\sigma}_{k+1}$ ($\boldsymbol{\sigma}_n$), (2) the mean crack size at the end of the previous increment, c_n, (3) the full time step size, Δt, and (4) the crack rate model parameters. The algorithm returns the crack size at the end of the current increment, c_{k+1}, as well as the algorithmically consistent tangents, $\partial\mathcal{C}_{k+1} = \left\{\frac{\partial c}{\partial \bar{\sigma}}, \frac{\partial c}{\partial p}, \frac{\partial c}{\partial v_{res}}\right\}_{k+1}$.

The various times and time steps involved in the sub-incrementation scheme are illustrated in Figure 2. The main idea is that the full time step, Δt, is split into sub-increments, Δt^*, and the stress tensor is assumed to vary linearly in time within the full time step. The net cumulative fraction of the total time step that has been completed at the end of the previously accepted sub-increment is $\Delta t_{\text{net}}/\Delta t$, and the corresponding time at the end of the previously accepted sub-increment is $t^- = t_n + \Delta t_{\text{net}}$. The time at the end of the trial sub-increment, Δt^*, is $t^+ = t^- + \Delta t^*$.

Algorithm 3 CrackUpdate, third-order sub-incrementation embedded Runge–Kutta crack update

```
Input: Vars : $\boldsymbol{\sigma}_{k+1}, \boldsymbol{\sigma}_n, c_n, \Delta t$,    Params : $K_0, \mu', m, v_{res}$
Output: $c_{k+1}, \partial\mathcal{C}_{k+1}$
compute:   $\dot{\boldsymbol{\sigma}} = (\boldsymbol{\sigma}_{k+1} - \boldsymbol{\sigma}_n)/\Delta t$
initialize:   $\Delta t^* = \Delta t, \quad \Delta t_{\text{net}} = 0, \quad \partial\mathcal{C} = 0, \quad c^- = c_n, \quad \boldsymbol{\sigma}^- = \boldsymbol{\sigma}_n$
repeat
    $\boldsymbol{\sigma}^+ = \boldsymbol{\sigma}^- + \dot{\boldsymbol{\sigma}}\,\Delta t^*$
    call RK3($\Delta t^*, \boldsymbol{\sigma}^+, \boldsymbol{\sigma}-, c^-$) $\longrightarrow c^+, \Delta\partial\mathcal{C}, \Delta t_{\text{rec}}$
    if $\Delta t^* \leq \Delta t_{\text{rec}}$ then
        update:  $\Delta t_{\text{net}} \leftarrow \Delta t_{\text{net}} + \Delta t^*, \quad \partial\mathcal{C} \leftarrow \partial\mathcal{C} + \Delta\partial\mathcal{C}, \quad c^- \leftarrow c^+, \quad \boldsymbol{\sigma}^- \leftarrow \boldsymbol{\sigma}^+$
        update: $\Delta t^* \leftarrow \Delta t^* + 0.9\,(\Delta t_{\text{rec}} - \Delta t^*)$
    else
        $\Delta t^* = 0.9\Delta t_{\text{rec}}$
    end if
    update $\Delta t^* \leftarrow$ Min($\Delta t^*, \Delta t - \Delta t_{\text{net}}$)
until $\Delta t_{\text{net}} = \Delta t$
$c_{k+1} = c^+, \quad \partial\mathcal{C}_{k+1} = \partial\mathcal{C}$
```

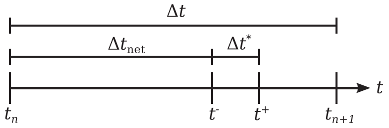


Figure 2. Diagram illustrating various times and time steps associated with sub-incrementation scheme Algorithm 3.

In Step 1, the average stress rates over the increment are calculated. In Step 2, the algorithm initializes several variables. Note that the first trial sub-increment time step is the full time step size, $\Delta t^* = \Delta t$. Steps 3–13 reflect a loop, which is performed until the net sub-incremental time step completed is equal to the full time step, $\Delta t_{\text{net}} = \Delta t$. Within this loop, Step 4 computes a trial value for the stress, $\boldsymbol{\sigma}^+$, corresponding to the trial sub-incremental time t^+. Step 5 uses a third-order Runge–Kutta scheme (cf. Algorithm 4) to integrate the crack kinetics of Equations (10a) and (10b) and compute a recommended time sub-increment time step size, Δt_{rec}, that ensures integration error in c is below an allowable error tolerance. If the sub-increment size is smaller than the recommended size, $\Delta t^* < \Delta t_{\text{rec}}$, the trial sub-increment is accepted, and all values are updated accordingly. Otherwise, the sub-increment size is reduced in Step 10 and reattempted. Step 12 ensures that the sum of the sub-increments does not exceed the total increment.

Integration of the differential Equations (10a) and (10b) over each sub-increment (Algorithm 3, Step 5) is performed using a third-order accurate Runge–Kutta technique attributed to Bogacki and Shampine [23]. This method balances accuracy, robustness, and computational efficiency using four function evaluations to provide an updated crack size, an estimate of the numerical error, and a recommended time increment to ensure the estimated error is less than a specified tolerance. Considering here the integration of Equations (10a) and (10b) over a sub-increment Δt^*, the updated crack size is computed by

$$c^+ = c^- + b_1^* A_1 + b_2^* A_2 + b_3^* A_3 + b_4^* A_4 \tag{37}$$

where the components, A_i, are computed as

$$A_i = \Delta t \dot{c} \left(\boldsymbol{\sigma}^- + d_i \Delta \boldsymbol{\sigma}, c^- + \sum_{j=1}^{i-1} a_{ij} A_j \right) \tag{38}$$

and b_i, d_i, and a_{ij} are integration coefficients listed in Table II. An estimate of numerical error introduced over the sub-increment is

$$E_{est} = \sum_{i=1}^{4} \left(b_i - b_i^* \right) A_i \tag{39}$$

If this error estimate is greater than some allowable error tolerance, E_{tol}, then the recommended sub-increment size Δt_{rec} is computed as

$$\Delta t_{\text{rec}} = \Delta t \left| \frac{E_{tol}}{E_{est}} \right|^{\frac{1}{3}} \tag{40}$$

A detailed outline of the implemented integration scheme is presented in Algorithm 4 for completeness.

Table II. Bogacki–Shampine Parameters for embedded Runge–Kutta Method.

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

Algorithm 4 RK3, third-order embedded Runge–Kutta crack update

```
Input: Vars : $\boldsymbol{\sigma}^+$, $\tilde{\boldsymbol{\sigma}}^-$, $c^-$, $\Delta t^*$,   Params : $K_0, \mu', m, v_{res}$,   Const : $b_i, b_i^*, d_i, a_{i,j}$ (cf. Table II)
Output: $c^+$, $\Delta\partial\mathcal{C}$, $\Delta t_{\mathrm{rec}}$
initialize: $\Delta\boldsymbol{\sigma} = \boldsymbol{\sigma}^+ - \boldsymbol{\sigma}^-$
for $i = 1$ to 4 do
    compute: $\boldsymbol{\sigma}_{(i)} = \boldsymbol{\sigma}^- + d_i \Delta\boldsymbol{\sigma}$,   $c_{(i)} = c^- + \sum_j (a_{ij} A_j)$
    evaluate Equation (10a), (10b) at $(\boldsymbol{\sigma}_{(i)}, c_{(i)}) \longrightarrow \dot{c}, \frac{\partial \dot{c}}{\partial \bar{\sigma}}, \frac{\partial \dot{c}}{\partial c}$
    compute: $A_i = \Delta t^* \dot{c}$
end for
$c^+ = c^- + \sum_i b_i A_i$,   $E_{est} = \sum_i \left(b_i - b_i^*\right) A_i$,   $\Delta t_{rec} = \Delta t \left|\frac{E_{tol}}{E_{est}}\right|^{\frac{1}{3}}$
```

Damage Strain Error Jacobian

The Newton iterative scheme used to solve Equation (36) within Algorithm 2 requires computation of the system Jacobian defined as

$$\mathbb{J} = \frac{\partial \mathbf{R}_k}{\partial \mathbf{e}_k^D} \tag{41}$$

The residual at the end of the k^{th} increment can be expressed as

$$\mathbf{R}_k = \frac{1}{2G_0} \left(\frac{c_k}{a}\right)^3 \mathbf{S}_k - \mathbf{e}_k + \mathbf{e}_k^{ve} \tag{42}$$

Thus, an infinitesimally small incremental change in the residual at the end of the k^{th} iteration can be expressed as

$$\Delta\mathbf{R}_k = \frac{1}{2G_0} \left(\frac{c_k}{a}\right)^3 \Delta\mathbf{S}_k + \frac{3}{2G_0 a} \left(\frac{c_k}{a}\right)^2 \mathbf{S}_k \Delta c_k - \Delta\mathbf{e}_k + \Delta\mathbf{e}_k^{ve} \tag{43}$$

An incremental change in deviatoric stress attributed to an incremental change in viscoelastic strain is expressed using the deviatoric viscoelastic tangent stiffness,

$$\Delta\mathbf{S}_k = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{e}_k^{ve} \tag{44}$$

Likewise, the incremental change in damage is

$$\Delta c_k = \frac{\partial c_k}{\partial \mathbf{S}} : \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{e}_k^{ve} \tag{45}$$

where $\frac{\partial c_k}{\partial \mathbf{S}}$ is computed in Algorithm 3. Combining Equations (43), (44), and (45) results in

$$\Delta\mathbf{R}_k = \frac{1}{2G_0} \left(\frac{c_k}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{e}_k^{ve} + \frac{3}{2G_0 a} \left(\frac{c_k}{a}\right)^2 \mathbf{S}_k \frac{\partial c}{\partial \mathbf{S}} : \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta\mathbf{e}_k^{ve} - \Delta\mathbf{e}_k + \Delta\mathbf{e}_k^{ve} \tag{46}$$

The system Jacobian for solving the semi-implicit residual equations is computed by

$$\mathbb{J}_k = \frac{\partial \Delta \mathbf{R}_k}{\partial \Delta \boldsymbol{\epsilon}_{ve}} = \left(1 + \frac{G_t}{G_0}\left(\frac{c_k}{a}\right)^3\right)\mathbb{I} + \frac{9G_t}{2G_0 a\bar{\sigma}}\left(\frac{c_k}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}}\mathbf{S}_k \otimes \mathbf{S}_k \tag{47}$$

where use has been made of $\frac{\partial \mathbf{S}}{\partial \mathbf{e}_{ve}} = 2G_t \mathbb{I}_{\text{sym}}$

The partial derivative of the updated crack size with respect to effective stress, $\frac{\partial c}{\partial \bar{\sigma}}$, is required for Equation (47) and is approximated by the derivative over a full time increment (without sub-incrementation) as

$$\begin{aligned}\frac{\partial c}{\partial \bar{\sigma}} \approx &\left(\sum_{i=2}^{4} b^*_{(i)} d_{(i)} \left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_{(i)}\right)\Delta t + \left(\sum_{i=3}^{4}\sum_{j=2}^{i-1} b^*_{(i)} a_{(ij)} d_{(j)} \left.\frac{\partial \dot{c}}{\partial c}\right|_{(i)} \left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_{(j)}\right)\Delta t^2 \\ &+ \left(b^*_{(4)} a_{(32)} d_{(2)} \left.\frac{\partial \dot{c}}{\partial c}\right|_{(4)} \left.\frac{\partial \dot{c}}{\partial c}\right|_{(3)} \left.\frac{\partial \dot{c}}{\partial \bar{\sigma}}\right|_{(2)}\right)\Delta t^3\end{aligned} \tag{48}$$

5. CONSTITUTIVE JACOBIAN

In order to implicitly solve the nonlinear global equations of motion, the constitutive update algorithm must provide a constitutive Jacobian tangent stiffness of the form $\frac{\partial \Delta \boldsymbol{\sigma}}{\partial \Delta \boldsymbol{\epsilon}}$. Using the chain rule of differentiation and Equations (2) and (4), the constitutive Jacobian can be expressed in a manner reflecting the volumetric and deviatoric contributions, that is,

$$\frac{\partial \Delta \boldsymbol{\sigma}}{\partial \Delta \boldsymbol{\epsilon}} = \frac{\partial \Delta \mathbf{S}}{\partial \Delta \mathbf{e}} : \mathbb{P} - \frac{\partial \Delta p}{\partial \Delta \mathbf{e}} : \mathbb{P} \otimes \mathbf{I} + \left[\frac{\partial \Delta \mathbf{S}}{\partial \Delta \epsilon^{vol}} - \frac{\partial \Delta p}{\partial \Delta \epsilon^{vol}}\mathbf{I}\right] \otimes \mathbf{I} \tag{49}$$

where the deviatoric projection operator has been introduced as

$$\mathbb{P} = \mathbb{I} - \frac{1}{3}\mathbf{I} \otimes \mathbf{I} \tag{50}$$

Noting that the deviatoric stress directly depends only on the viscoelastic part of deviatoric strain, and making use of Equation (3),

$$\Delta \mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \left(\Delta \mathbf{e} - \Delta \mathbf{e}^{D}\right) \tag{51}$$

Differentiating Equation (7), substituting the result into Equation (51), and collecting terms gives

$$\left[\mathbb{I} + \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}}\right] : \Delta \mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \Delta \mathbf{e} - \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S}\Delta c \tag{52}$$

The dependence of crack growth rate on stress and strain permits the infinitesimal change in crack size to be expressed using the chain rule as

$$\Delta c = \frac{\partial c}{\partial \bar{\sigma}}\frac{\partial \bar{\sigma}}{\partial \mathbf{S}} : \Delta \mathbf{S} + \frac{\partial c}{\partial p}\Delta p + \frac{\partial c}{\partial v_{res}}\frac{\partial v_{res}}{\partial \dot{\bar{e}}}\frac{\partial \dot{\bar{e}}}{\partial \dot{\mathbf{e}}} : \frac{\partial \dot{\mathbf{e}}}{\partial \Delta \mathbf{e}} : \Delta \mathbf{e} \tag{53}$$

Noting that, $\frac{\partial \bar{\sigma}}{\partial \mathbf{S}} = \frac{3\mathbf{S}}{2\bar{\sigma}}$, considering Equation

$$\frac{\partial v_{res}}{\partial \dot{\bar{e}}}\frac{\partial \dot{\bar{e}}}{\partial \dot{\mathbf{e}}} : \frac{\partial \dot{\mathbf{e}}}{\partial \Delta \mathbf{e}} = \frac{2}{3}\frac{v_a \; v_{res}}{\Delta t \; \dot{\bar{e}}^2}\dot{\mathbf{e}} \tag{54}$$

and substituting these expressions into Equation (53) and, in turn, into Equation (52) yields

$$\begin{aligned}
&\left[\mathbb{I} + \frac{1}{2G_0}\left(\frac{c}{a}\right)^3 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} + \frac{9}{4G_0 a\bar{\sigma}}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}} \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S} \otimes \mathbf{S}\right] : \Delta \mathbf{S} \\
&= \left[\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} - \frac{v_{res} v_a}{G_0 a \Delta t \dot{\bar{e}}^2}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial v_{res}} \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S} \otimes \dot{\mathbf{e}}\right] : \Delta \mathbf{e} \\
&\quad - \frac{3}{2G_0 a}\left(\frac{c}{a}\right)^2 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S}\left(\frac{\partial c}{\partial p} \frac{\partial p}{\partial \epsilon_{vol}} \Delta \epsilon_{vol}\right)
\end{aligned} \tag{55}$$

Differentiating Equation (32) (cf. Algorithm 1) provides an expression for

$$\frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} = 2G_t \mathbb{I} \tag{56}$$

where

$$G_t = G_0 - \sum_{\kappa=1}^{N} G^{(\kappa)} \frac{\tau^{(\kappa)}}{\Delta t}\left(\frac{\Delta t}{\tau^{(\kappa)}} + \exp\left[\frac{-\Delta t}{\tau^{(\kappa)}}\right] - 1\right) \tag{57}$$

In this paper, the volumetric response is defined using a linearly elastic (i.e. constant) bulk modulus, thus, $\frac{\partial p}{\partial \epsilon_{vol}} = -K$. To include bulk viscoelastic effects, one would replace K with a volumetric analog to the right of Equation (56). Solving Equation (55) for $\Delta \mathbf{S}$ and substituting Equation (56) and the bulk modulus into the result provides

$$\Delta \mathbf{S} = \mathbb{A}^{-1} : \mathbb{B} : \Delta \mathbf{e} + \mathbb{A}^{-1} : \mathbf{C}\, \Delta \epsilon_{vol} \tag{58}$$

where

$$\begin{aligned}
\mathbb{A} &= \left(1 + \frac{G_t}{G_0}\left(\frac{c}{a}\right)^3\right)\mathbb{I} + \frac{9G_t}{2G_0 a\bar{\sigma}}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}} \mathbf{S} \otimes \mathbf{S} \\
\mathbb{B} &= 2G_t \mathbb{I} - \frac{2G_t v_{res} v_a}{G_0 a \Delta t \dot{\bar{e}}^2}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial v_{res}} \mathbf{S} \otimes \dot{\mathbf{e}} \\
\mathbf{C} &= \frac{3}{2}\frac{K}{G_0 a}\left(\frac{c}{a}\right)^2 \frac{\partial \mathbf{S}}{\partial \mathbf{e}^{ve}} : \mathbf{S}\left(\frac{\partial c}{\partial p}\right)
\end{aligned} \tag{59}$$

Equation (58) is differentiated to obtain

$$\frac{\partial \Delta \mathbf{S}}{\partial \Delta \mathbf{e}} = \mathbb{A}^{-1} : \mathbb{B} \tag{60}$$

and

$$\frac{\partial \Delta \mathbf{S}}{\partial \Delta \epsilon_{vol}} = \mathbb{A}^{-1} : \mathbf{C} \tag{61}$$

Substituting Equations (60) and (61) into (49) and noting that the pressure has no direct dependence on the deviatoric strain in viscoScram, that is, $\frac{\partial p}{\partial \mathbf{e}} = \mathbf{0}$, gives

$$\frac{\partial \Delta \boldsymbol{\sigma}}{\partial \Delta \boldsymbol{\epsilon}} = \mathbb{A}^{-1} : \mathbb{B} : \left[\mathbb{I} - \frac{1}{3}\mathbf{I} \otimes \mathbf{I}\right] + [\mathbb{A}^{-1} : \mathbf{C} + K\mathbf{I}] \otimes \mathbf{I} \tag{62}$$

Letting

$$\begin{aligned}
A_1 &= 1 + \frac{G_t}{G_0}\left(\frac{c}{a}\right)^3 \\
A_2 &= \frac{9G_t}{2G_0 a\bar{\sigma}}\left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial \bar{\sigma}}
\end{aligned} \tag{63}$$

an expression for $\mathbb{A}^{-1}$ is obtained using a fourth-order corollary to the Sherman–Morrison formula (where the inverse of a fourth-order tensor is defined such that $\mathbb{A}^{-1} : \mathbb{A} = \mathbb{I}$) [24]

$$\mathbb{A}^{-1} = \frac{1}{A_1}\mathbb{I} - \frac{A_2 \mathbf{S} \otimes \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \tag{64}$$

Further simplifications to the notation are afforded by the substitutions

$$A_3 = \frac{2G_t v_{res} v_a}{G_0 a \Delta t\, \dot{e}^2} \left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial v_{res}} \tag{65}$$

$$A_4 = \frac{3KG_t}{G_0 a} \left(\frac{c}{a}\right)^2 \frac{\partial c}{\partial p} \tag{66}$$

Using these simplifications and Equation (64) enables

$$\begin{aligned} \frac{\partial \Delta \boldsymbol{\sigma}}{\partial \Delta \boldsymbol{\epsilon}} &= K\mathbf{I} \otimes \mathbf{I} + \frac{2G_t}{A_1}\left[\mathbb{I} - \frac{1}{3}\mathbf{I} \otimes \mathbf{I}\right] - \frac{2G_t A_2}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} \mathbf{S} \otimes \mathbf{S} \\ &+ \left[\frac{A_2 A_3 \mathbf{S} : \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}} - \frac{A_3}{A_1}\right] \mathbf{S} \otimes \dot{\mathbf{e}} + \left[\frac{A_4}{A_1} - \frac{A_2 A_4 \mathbf{S} : \mathbf{S}}{A_1^2 + A_1 A_2 \mathbf{S} : \mathbf{S}}\right] \mathbf{S} \otimes \mathbf{I} \end{aligned} \tag{67}$$

which is equivalent to Equation (62) but has the benefit of being more readily converted into a matrix reflecting Voigt notation.

6. COMPARISON WITH HISTORICAL IMPLEMENTATION

In this section, the performance of the new semi-implicit integration scheme is compared to that of the explicit integration scheme [2] for finite element simulations of uniaxial stress compression of a cylindrical specimen and bending of a cantilevered beam under tip load. The parameters used for these simulations do not necessarily reflect PBX-9501 (i.e., were not obtained by fitting to experimental data) and are listed in Tables III and IV.

Uniaxial compression of a cylindrical specimen

As illustrated in Figure 3, a 3D simulation of the compression of a cylindrical specimen of diameter 9.525 mm and length 19.05 mm under uniaxial stress conditions was simulated using a quarter-symmetry finite element mesh composed of 456 fully integrated linear hexahedral elements

Table III. Generalized Maxwell parameters used in simulations to compare integration algorithms.

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$	$i = 9$	$i = 10$
τ_i (s)	10^3	10^2	10^1	10^0	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
G_i (MPa)	109	108	139	170	213	267	341	434	581	726

Table IV. Damage kinetics parameters used in simulations to compare integration algorithms.

Parameter	Value	Unit	Parameter	Value	Unit
K	3460	MPa	v_{max}	3×10^5	$\frac{\text{mm}}{\text{s}}$
G_∞	404	MPa	v_a	0.892	–
K_0	0.03	MPa $\sqrt{\text{mm}}$	v_b	2.28	–
μ'	1.16	–	c_1	15.7	–
m	10.0	–	c_2	199.4	K
c_0	3.0	μm	T_0	303.1	K
a	1.0	mm			

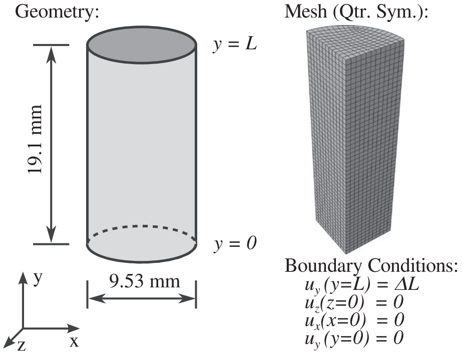
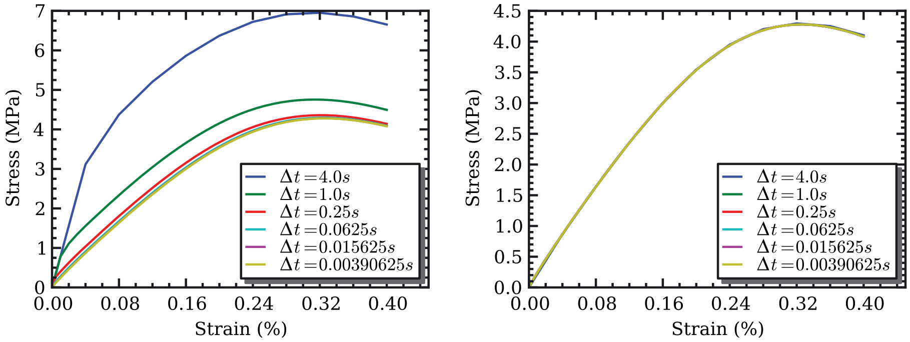


Figure 3. Illustration of model for simulations of uniaxial compression of cylindrical specimen.



(a) Explicit update algorithm

(b) New Semi-implicit update algorithm

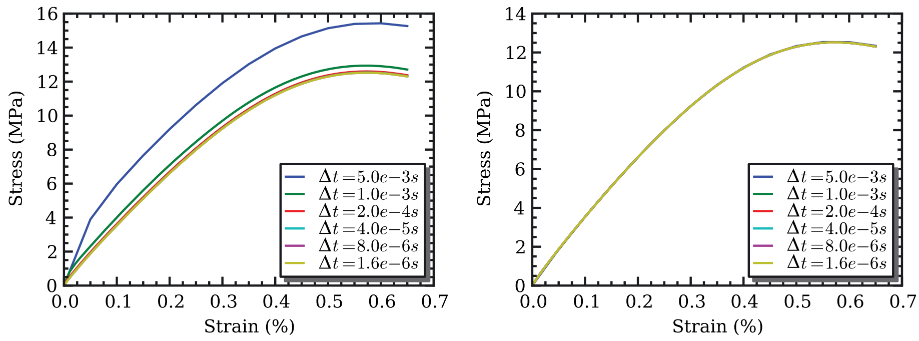
Figure 4. Comparison of stress versus strain results from simulations of uniaxial (stress) compression of cylinder at $\dot{\epsilon} = 10^{-4}\text{s}^{-1}$. (a) Explicit update algorithm, (b) New Semi-implicit update algorithm.

(C3D8) in Abaqus [20]. The uniaxial nominal stress (net axial force divided by initial area) versus nominal strain (computed as change in length divided by initial specimen length) for simulations of constant strain rate of 10^{-4} s^{-1} employing displacement boundary conditions (i.e., displacement controlled loading) is compared in Figure 4 for six time step sizes ranging from 4.0 to 4.0 s. The explicit implementation exhibits a greater dependence on time step size compared to the semi-implicit integration scheme as damage starts to accumulate. For both algorithms, the results converge as time step size is reduced, although the convergence occurs more rapidly (at larger time step size) for the semi-implicit scheme.

In order to investigate the performance of the new semi-implicit algorithm over the intended range of applicable strain rates, previously discussed simulations were repeated at a strain rate of 10^{-1} s^{-1}, and the corresponding stress–strain results are shown in Figure 5. Observations from Figure 5 are similar to those made from Figure 4.

To assess the temporal convergence rates exhibited in Figures 4 and 5, an error ansatz of $\sigma = \sigma_0 + A\Delta t^p$ was fit to the peak stress versus time step size. The estimated error $|\sigma - \sigma_0|$ is plotted versus time step size in Figure 6. The observed orders of convergence for the strain rate of 10^{-4}s^{-1} are approximately $p = 3.8$ and $p = 1.3$ for the semi-implicit and explicit algorithms, respectively. For a strain rate of 10^{-1}s^{-1}, the orders of convergence are $p = 1.9$ and $p = 1.1$ for the semi-implicit and explicit algorithms, respectively. As expected, the semi-implicit scheme converges at a higher rate than the explicit scheme, and, for any given time step size, the semi-implicit scheme is more accurate than the explicit scheme in these simulations.

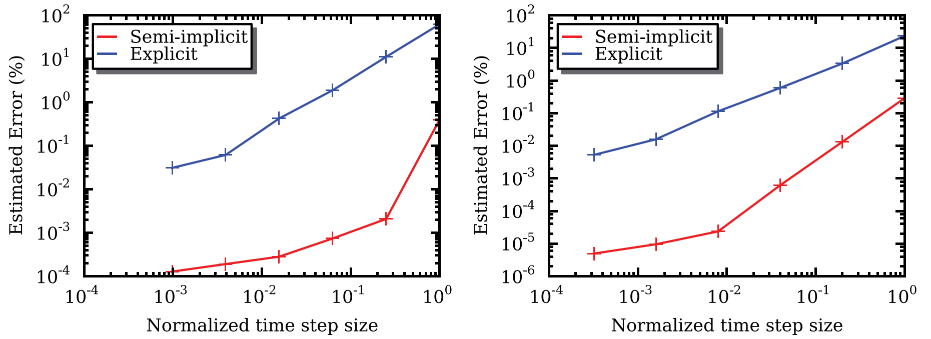
Another important aspect of algorithm performance is the computational cost. The computation time recorded as elapsed wallclock time for a given simulation is listed in Table V for the compression simulations under both strain rates. As expected, the semi-implicit scheme is always slower than the explicit scheme for a given time step size. The factor by which the explicit scheme outper-



(a) Explicit update algorithm

(b) New Semi-implicit update algorithm

Figure 5. Comparison of stress versus strain results from simulations of uniaxial (stress) compression of cylinder at $\dot{\epsilon} = 10^{-1}\text{s}^{-1}$. (a) Explicit update algorithm (b) New Semi-implicit update algorithm.



(a) $\dot{\epsilon} = 10^{-4}\text{s}^{-1}$

(b) $\dot{\epsilon} = 10^{-1}\text{s}^{-1}$

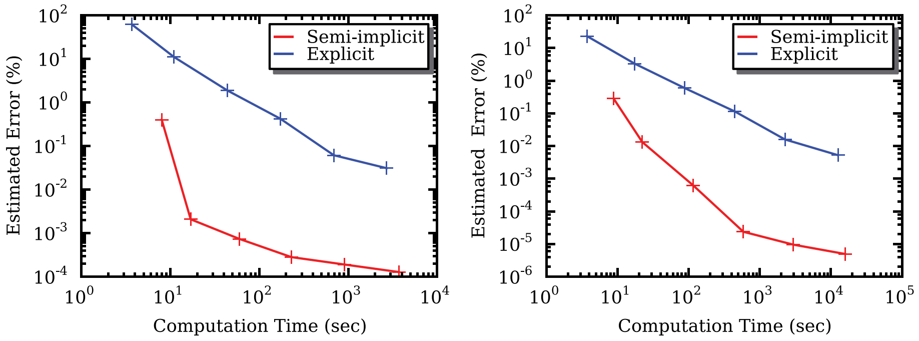
Figure 6. Comparison of temporal convergence illustrating estimated error in peak stress versus normalized time step size for simulations of uniaxial (stress) compression. *Blue* curve reflects the explicit update scheme and *red* reflects the new semi-implicit scheme. (a) $\dot{\epsilon} = 10^{-4}\text{s}^{-1}$ (b) $\dot{\epsilon} = 10^{-1}\text{s}^{-1}$.

Table V. Comparison of elapsed wallclock time (in seconds) for simulations using the explicit update [2] versus semi-implicit update (this work).

$\dot{\epsilon} = 10^{-4}\text{s}^{-1}$			$\dot{\epsilon} = 10^{-1}\text{s}^{-1}$		
Δt(s)	Explicit	Semi-implicit	Δt (s)	Explicit	Semi-implicit
4.0×10^{0}	3.66	7.98	5.0×10^{-3}	87.7	116
6.3×10^{-2}	172	230	2.0×10^{-4}	2260	2970
3.9×10^{-3}	2700	3700	1.6×10^{-6}	12600	15800

forms the semi-implicit scheme is reduced as the time step becomes smaller, because the number of iterations required by the semi-implicit algorithm is eventually reduced to one for a sufficiently small time step. There are more floating point operations in one iteration of the semi-implicit algorithm than in a single pass of the explicit scheme, so the semi-implicit scheme will always be slower than the explicit scheme for a fixed time step size.

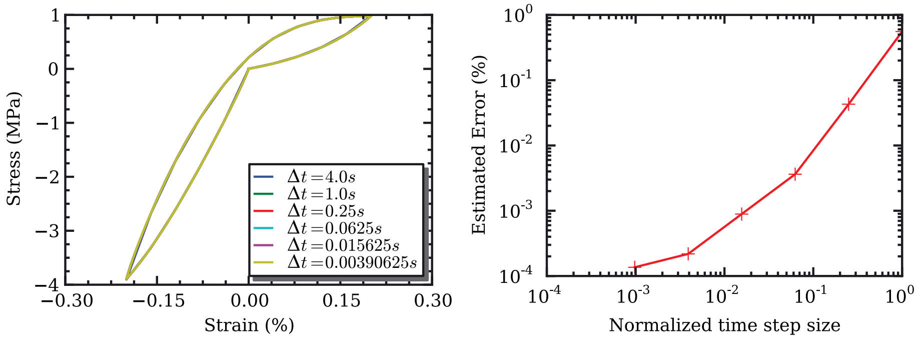
Recall the results from Figure 6, which indicated the semi-implicit scheme is more accurate for any given time step size than the explicit scheme. The increased accuracy comes at a cost as indicated by Table V. Therefore, a meaningful comparison of the performance of the integration scheme must balance the required computational time with the achieved accuracy. Estimated error is plotted against computational time for both strain rates in Figure 7. From these results, it is clear that, for a fixed allocation of computational time, the semi-implicit scheme is always more accurate. Equivalently, a required level of accuracy can be achieved with less computational time using the semi-implicit scheme compared to the explicit scheme.



(a) $\dot{\epsilon} = 10^{-4}\mathrm{s}^{-1}$

(b) $\dot{\epsilon} = 10^{-1}\mathrm{s}^{-1}$

Figure 7. Comparison of computational cost illustrating estimated error in peak stress versus computational time for simulations of uniaxial (stress) compression. *Blue* curve reflects the explicit update scheme and *red* reflects the new semi-implicit scheme. (a) $\dot{\epsilon} = 10^{-4}\mathrm{s}^{-1}$ (b) $\dot{\epsilon} = 10^{-1}\mathrm{s}^{-1}$.



(a)

(b)

Figure 8. Comparison of simulation results under cyclic compression-tension conditions for various time step sizes. (a) Stress versus strain and (b) error versus time step size indicating strong temporal convergence.

In order to test the new implementation under reversals between tensile and compressive loading, simulations of a single compression-unload-tension-unload cycle at a fixed strain rate of $\dot{\epsilon} = 10^{-4}\mathrm{s}^{-1}$ were conducted at various time step sizes. Selected results from this suite of simulations are shown in Figure 8. It is clear from these figures that the strong temporal convergence is preserved even under reversals in stress that affect the crack growth through evaluation of Equations (9a), (9b), and (13). The stress–strain response depicted in Figure 8(a) reflects both damage growth and viscoelasticity. There is a clear tension-compression asymmetry in part due to the previous accumulation of damage upon tensile loading and in part due to the increased damage growth rate under tensile loading. It is a coincidence that, while there is a residual viscoelastic strain upon reaching zero stress after initial compressive loading and unloading, the subsequent tensile cycling unloads to nearly zero stress at a final state of zero strain.

Bending of cantilever beam

To evaluate the algorithms under a spatially non-uniform stress, a cantilever beam with a prescribed tip displacement was modeled. The model consisted of a cantilever beam with square (10 mm × 10 mm) cross-section and length $L = 100$ mm. The fixed end ($x = 0$) was modeled by specifying zero displacement in the x direction, nodes at the fixed end with coordinate $y = 0$ were constrained in the y direction, and nodes on the fixed end with coordinate $z = 0$ were constrained in the z direction (Figure 9). Figure 10(a) and (b) shows the change in load-deflection behavior with time step size for the explicit and semi-implicit schemes, respectively. For this case, the two algorithms are not expected to converge to the same result, because these simulations include regions of tensile stress, and, as discussed in Section 2, the semi-implicit

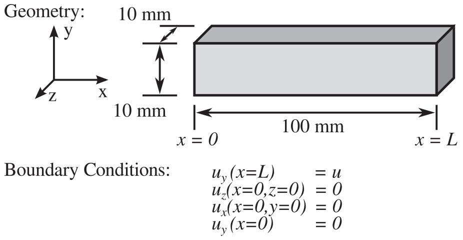
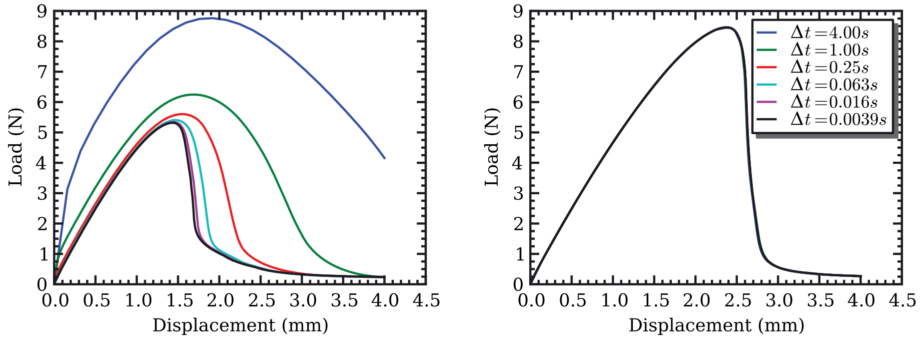


Figure 9. Illustration of model for simulations of cantilevered beam bending.



(a) Explicit

(b) Semi-implicit

Figure 10. Load versus displacement results from displacement controlled beam bending simulations. All curves in (b) lie on top of each other. Note that the legend in (b) also applies to (a).

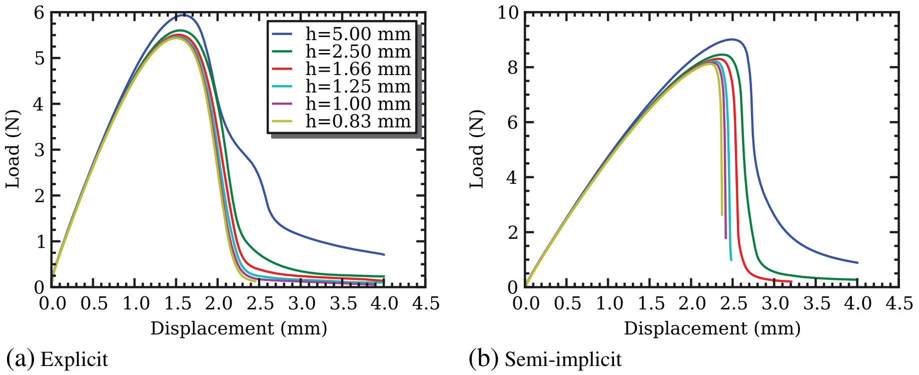
scheme employs the modified crack kinetics of Equation (13), whereas the explicit scheme employs Equation (12). Thus, these results reflect the solution to two *different* problems. Following previously discussed methods, the orders of temporal convergence in peak load were computed as $p = 1.7$ and $p = 0.84$ for semi-implicit and explicit schemes, respectively. The convergence behavior exhibited in Figure 10 is good for the semi-implicit scheme and relatively poor for the explicit scheme.

Spatial convergence for the cantilever beam bending problem was assessed in a similar manner as for the unaxial compression. The load versus displacement results for simulations employing six different element sizes ranging from 0.83 to 5.0 mm are shown in Figure 11. The observed orders of spatial convergence in peak load are $p = 1.1$ and $p = 1.3$ for semi-implicit and explicit schemes, respectively. The explicit scheme converges at a slightly higher rate, but we believe this is due to the differences in the crack kinetics model. These orders of spatial convergence are similar and, again, likely dominated by the truncation error in the element shape functions.

Restrained thermal contraction of hollow cylinder around mandrel

The targeted application of the newly implemented viscoSCRAM model is simulating the long-term thermomechanical response of weapon system components during storage and life-cycle events prior to transient impact events. In this section, we present results using the new implementation to demonstrate utility in this intended application. We consider the thermal contraction of an internally constrained hollow cylinder of PBX-9501 during evolving temperature gradients caused by differential cooling between the interior and exterior surfaces. The geometry, computational mesh, displacement boundary conditions, and prescribed temperature field used in this problem are depicted in Figure 12. The boundary conditions comprise zero displacement in radial and tangential directions on the interior surface of the cylinder reflecting a high-friction rigid internal restraint. The temperature on the inner surface of the hollow cylinder is spatially uniform and evolves at a constant rate of $T(r = R_0) = T_0 - \dot{T}_1 t$. The temperature distribution on the exterior surface is prescribed

DOI: 10.1002/nme



(a) Explicit (b) Semi-implicit

Figure 11. Load versus displacement results from cantilever beam bending simulations showing convergence with reduction of characteristic element size, h. Note that the legend in (a) also applies to (b). (a) Explicit (b) Semi-implicit.

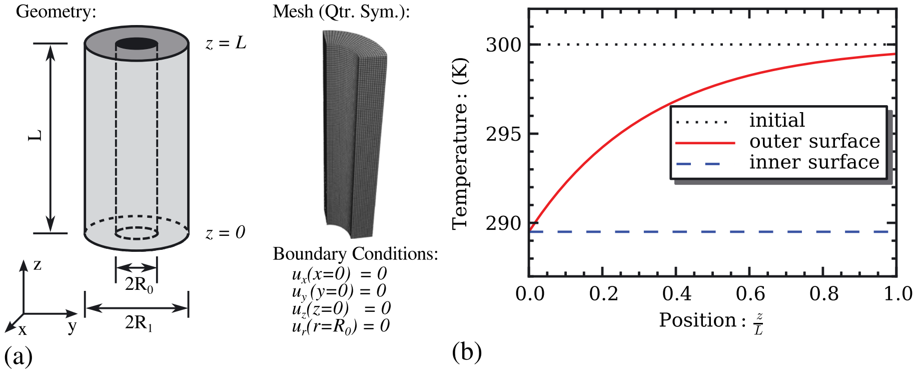


Figure 12. Illustration of model for simulations of restrained thermal contraction of hollow cylinder around rigid mandrel. (a) Geometry, mesh, and boundary conditions, (b) prescribed temperature profile at $t = 0$ s and $t = 300$ s.

according to

$$T(r = R_1, z, t) = T_0 - \dot{T}_1 t \exp\left(-3\frac{z}{L}\right) \tag{68}$$

and there is a linear variation through the thickness of the specimen. The Williams–Landel–Ferry time-temperature superposition model [25] is used for integration of the viscoelastic response (i.e., Algorithm 1) away from the reference temperature, $T_{ref} = 303.1$ K. Specifically, we integrate the viscoelastic response over a scaled time step, $\Delta t'$, using the relationship

$$\frac{\partial t'}{\partial t} = A(T) \tag{69}$$

where the shift function is defined as

$$-\log[A(T)] = \frac{c_1(T - T_{ref})}{c_2 + T - T_{ref}} \tag{70}$$

For the the rmomechanical problem at hand, characterized by slow rates of temperature change, it is sufficient to approximate the integral of Equation (69) numerically making the midpoint approximation

Table VI. Parameter values used for simulating the non-uniform thermal contraction of a hollow cylinder.

Parameter	Value	Unit	Parameter	Value	Unit
R_0	25	mm	T_{ref}	303.1	K
R_1	50	mm	T_0	300	K
L	250	mm	$\dot{T}_1$	0.035	K s^{-1}
c_1	15.7	-	α	55×10^{-6}	K^{-1}
c_2	199.4	K			

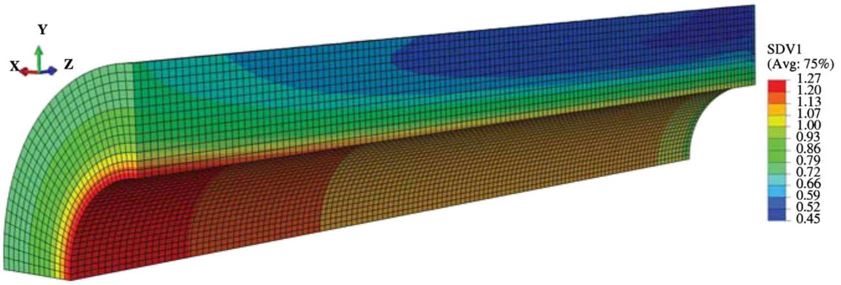
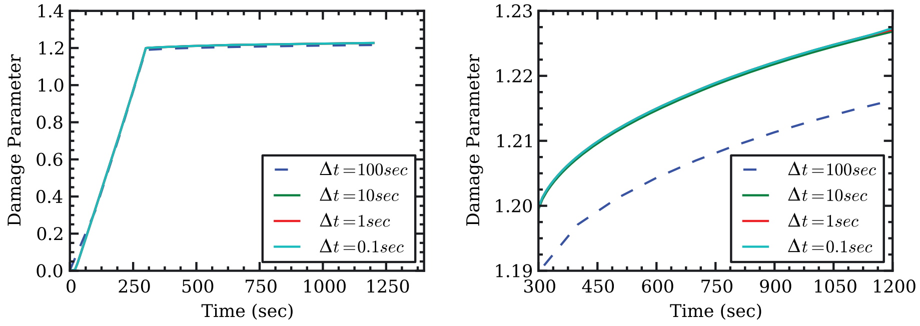


Figure 13. Contour plot of normalized damage variable $\frac{c}{a}$.



(a) (b)

Figure 14. Time history of damage, $\frac{c}{a}$, evolution for thermal contraction simulation at location of maximum damage. (a) Over temperature 'ramp' and hold, (b) axes 'zoomed' to illustrate creep damage evolution during the isothermal hold.

$$\Delta t' = A\left(\frac{1}{2}(T_n + T_{n+1})\right)\Delta t \tag{71}$$

The parameter values used for this model problem are reported in Table VI.

Simulation results from this representative model problem are provided in Figures 13 and 14. Contours of the normalized damage magnitude, $\frac{c}{a}$, at the final state of the simulation (i.e., $t = 1200$ s) are plotted in Figure 13. The spatial variation of damage is consistent with the spatial variation in stresses caused by the non-uniform thermal contraction. The location of peak damage is on the inner surface $r = R_0$ at the end $z = 0$ coinciding with the largest decrease in temperature. Note that the temperature gradient is zero throughout the thickness at this location. Figure 14 compares damage evolution time histories at the location of maximum damage resulting from simulations employing differing time steps. From these results, it is qualitatively evident that the damage evolution exhibits good temporal convergence consistent with the quantitative analysis discussed for the previous test problems. In Figure 14(a), it is clear that the majority of damage evolution occurs during the initial cooling, as expected; however, Figure 14(b) illustrates continued damage evolution associated with creep response during the 'hold' at cold temperature.

7. SUMMARY AND CONCLUDING REMARKS

A semi-implicit integration scheme for the stress update of a viscoelastic rate-dependent damage model applicable to energetic particulate composite materials has been developed. Implicit integration is used for the stress update, the updated crack size, and the decomposition of strain into damage and viscoelastic parts. Within the resulting nonlinear solution scheme, the crack evolution equations are integrated using an explicit third-order Runge–Kutta scheme across several sub-increments within the full time step. These sub-incrementation step sizes are automatically determined by the algorithm in order to ensure acceptable accuracy in the updated crack size.

Contrary to the title of a previous publication [2], previous implementations of the material model use an explicit integration scheme and consequently are prone to accumulation of significant error at large time steps. While inconsequential for time step sizes typically employed for solving high-rate impact problems frequently addressed with this material model, the error prohibits using large time steps needed to tractably simulate the long-term response of components (e.g., creep) prior to an impact event. Often, the constitutive model is used in an implicit finite element code to simulate the long-term creep response and subsequently transferred to an explicit finite element code to simulate the transient response to an impact event. In these cases, the transient simulation could potentially initiate with large errors in the thermo-mechanical state of the material when using the explicit constitutive update scheme.

Comparisons of results from simulations performed using the new semi-implicit scheme and the previous explicit scheme enable the following conclusions:

- The semi-implicit integration scheme is more accurate than the explicit scheme for any given time step size.
- The semi-implicit scheme requires more computational time than the previous explicit scheme for any given time step size.
- The semi-implicit scheme outperforms the explicit scheme when considering computational time required for a given level of accuracy for the strain rates considered here.

Accuracy

The observed temporal convergence of the semi-implicit scheme was consistently greater than that of the explicit scheme by one order of magnitude or more for all considered cases. Demonstrated orders of convergence reflect a complex sequence of tolerances combined with a viscoelastic update scheme that is exact for constant viscoelastic strain rates.

Computational time

The computational time for the explicit integration scheme is approximately linear in the number of time steps, because each time step consists of a fixed number of floating point operations. On the other hand, the semi-implicit scheme requires a variable number of operations depending on the number of iterations for the implicit solver and the number of sub-incrementations required to update the crack size within each iteration. The semi-implicit scheme required up to approximately twice the computational time as the explicit scheme for the simulations considered here. As time step size is reduced, the amount that the computational time required by the semi-implicit scheme exceeded that required by the explicit scheme was reduced to 25%. The semi-implicit scheme always requires more computational time than the explicit scheme, but not unreasonably so, especially when considering the corresponding levels of accuracy.

Performance

The overall performance of these two integration schemes is assessed by comparing computational time (cost in some sense) versus accuracy (or benefit). In this manner, the results from the example simulations indicate that (for these simulations) the semi-implicit scheme provides a higher level of accuracy for a fixed allocation of computational time, or, equivalently, a required level of accuracy can be provided with a lower computational time using the semi-implicit integration scheme. It is our expectation that these results will not apply under transient impact simulations where the time step size required to adequately solve the global conservation of momentum equations is sufficiently small that accuracy in the constitutive update scheme is trivially adequate. In these cases, the lower computational time per step using the explicit scheme would likely render it superior, although it would be expected that the computational savings in using the explicit scheme would be no greater than 25%.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Enhanced Surveillance Campaign at Los Alamos National Laboratory (T.G. Zocco, Program Manager) for supporting the constitutive model development and implementation presented here, and several fruitful discussions with F. L. Addessio, B. E. Clements, and M. W. Lewis. Additionally, the authors are appreciative of suggestions made by reviewers to enhance both the robustness of the algorithm and the demonstration of its utility within this paper.

REFERENCES

1. Bennett JG, Haberman KS, Johnson JN, Asay BW, Henson BF. A constitutive model for the non-shock ignition and mechanical response of high explosives. *Journal of the Mechanics and Physics of Solids* 1998; **46**(12):2303–2322.
2. Hackett RM, Bennett JG. An implicit finite element material model for energetic particulate composite materials. *International Journal for Numerical Methods in Engineering* 2000; **49**:1191–1209.
3. Clements BE, Mas EM. A theory for plastic-bonded materials with a bimodal size distribution of filler particles. *Modelling and Simulation In Materials Science and Engineering* 2004; **12**(3):PII S0965–0393(04)74405–8.
4. Guilkey JE, Harman TB, Banerjee B. An Eulerian–Lagrangian approach for simulating explosions of energetic devices. *Computers & Structures* 2007; **85**(11–14):660–674.
5. Le VD, Gratton M, Caliez M, Frachon A, Picart D. Experimental mechanical characterization of plastic-bonded explosives. *Journal of Materials Science* 2010; **45**(21):5802–5813.
6. Clements BE, Thompson DG, Luscher DJ, DeLuca R, Brown GW. Taylor impact tests and simulations of plastic bonded explosives. *Proceedings of the 17th Biennial Conference of the American Physical Society Topical Group on Shock Compression of Condensed Matter*, Vol. 1426: Proceedings of Amer Phys Soc (APS), Top Grp, Chicago, IL, 2012; 661–664.
7. Peterson JR, Wight CA. An Eulerian–Lagrangian computational model for deflagration and detonation of high explosives. *Combustion and Flame* 2012; **159**(7):2491–2499.
8. Thompson DG, DeLuca R, Wright WJ. Taylor impact tests and simulations of plastic bonded explosives. *Proceedings of the 17th Biennial Conference of the American Physical Society Topical Group on Shock Compression of Condensed Matter*, Vol. 1426: Amer Phys Soc (APS), Top Grp; Los Alamos Natl Lab; Lawrence LivermoreEOLEOLNatl Lab; Sandia Natl Labs, Chicago, IL, 2012; 657–660.
9. Ma X, Zhang DZ, Giguere PT, Liu C. Axisymmetric computation of taylor cylinder impacts of ductile and brittle materials using original and dual domain material point methods. *International Journal of Impact Engineering* 2013; **54**:96–104.
10. Addessio FL, Johnson JN. A constitutive model for the dynamic-response of brittle materials. *Journal of Applied Physics* 1990; **67**(7):3275–3286.
11. Dienes JK, Zuo QH, Kershner JD. Impact initiation of explosives and propellants via statistical crack mechanics. *Journal of the Mechanics and Physics of Solids* 2006; **54**(6):1237–1275.
12. Simo JC. On a fully 3-dimensional finite-strain viscoelastic damage model – formulation and computational aspects. *Computer Methods In Applied Mechanics and Engineering* 1987; **60**(2):153–173.
13. Govindjee S, Simo JC. Mullins effect and the strain amplitude dependence of the storage modulus. *International Journal of Solids and Structures* 1992; **29**(14–15):1737–1751.
14. Peña E, Calvo B, Martinez MA, Doblare M. On finite-strain damage of viscoelastic-fibred materials. Application to soft biological tissues. *International Journal For Numerical Methods In Engineering* 2008; **74**(7):1198–1218.
15. Yang BJ, Kim BR, Lee HK. Micromechanics-based viscoelastic damage model for particle-reinforced polymeric composites. *Acta Mechanica* 2012; **223**(6):1307–1321.
16. Darabi MK, Abu Al-Rub RK, Masad EA, Little DN. A thermodynamic framework for constitutive modeling of time- and rate-dependent materials. Part II: numerical aspects and application to asphalt concrete. *International Journal of Plasticity* 2012; **35**:67–99.
17. Zhu HR, Sun L. A viscoelastic-viscoplastic damage constitutive model for asphalt mixtures based on thermodynamics. *International Journal of Plasticity* 2013; **40**:81–100.
18. Xu F, Aravas N, Sofronis P. Constitutive modeling of solid propellant materials with evolving microstructural damage. *Journal of the Mechanics and Physics of Solids* 2008; **56**(5):2050–2073.
19. Dienes JK. A statistical-theory of fragmentation processes. *Mechanics of Materials* 1985; **4**(3–4):325–335.
20. Abaqus. *Abaqus Theory Manual* (version 6.10). Simulia: Providence, RI, 2010.

21. Dienes JK. Strain-softening via SCRAM. *Technical Report LA-UR-98-3620*, Los Alamos National Laboratory, 1998.
22. Buechler MA, Luscher DJ. A semi-implicit integration scheme for stress update of viscoscram. *Technical Report LA-UR-12-24915*, Los Alamos National Laboratory, 2012.
23. Bogacki P, Shampine LF. A 3(2) pair of Runge–Kutta formulas. *Applied Mathematics Letters* 1989; **2**(4):321–325.
24. Brannon RM. *Functional and Structured Tensor Analysis for Engineers*. University of New Mexico: Albuquerque, NM, 2003.
25. Williams ML, Landel RF, Ferry JD. Mechanical properties of substances of high molecular weight .19. the temperature dependence of relaxation mechanisms in amorphous polymers and other glass-forming liquids. *Journal of the American Chemical Society* 1955; **77**(14):3701–3707.