Report No.
UCB/SEMM-2002/07

Structural Engineering
Mechanics and Materials

Numerical Simulation of Case II Diffusion in One Dimension

By

Prashanth K. Vijalapura

August 2002

Department of Civil and Environmental Engineering
University of California, Berkeley

Abstract

In this report, time stepping schemes for the numerical simulation of Case II diffusion of a solvent into a glassy polymer are investigated. Two examples of 1-D models that produce Case II behavior are considered. The Initial Boundary Value Problem (IBVP) obtained from the governing equations that incorporate these Case II diffusion models are highly nonlinear due to the following: a nonlinear flux law to produce sharp fronts, a concentration dependent relaxation time of the polymer to model glass–rubber transition and coupling between the diffusion and mechanical displacements. To handle these nonlinearities effectively, a collection of relevant mathematical results are provided. Most notably, the solutions to the solvent concentration under certain conditions satisfy a maximum and a minimum principle. Also, the Differential-Algebraic-Equations (DAEs) arising from the spatial discretization of the IBVP are stiff due to mesh size and the concentration dependent relaxation time for the polymer. With this mathematical background, a fully implicit second order Backward Differentiation Formula (BDF2) and a partitioned implicit-explicit Explicit-Runge-Kutta-BDF2 adaptive time stepping schemes are proposed. A Krylov Newton method which provides an alternative to standard Newton's method or its variants for solving nonlinear equations is also examined. Through a careful parametric study, these numerical techniques for simulation of Case II behavior for the two models are evaluated. In addition to these techniques for time integration, a projection method to handle undershoots due to oscillations in the concentration solution field on coarse meshes is also presented.

The main inferences from the study are: Fully implicit schemes perform better than the implicit-explicit schemes. This is because the stiffness effects on the stability of the partitioned schemes were more significant than the reduction in the radius of convergence for Newton (or Krylov Newton) methods while solving nonlinear equations in fully implicit schemes. Adaptive time stepping is found to be very essential for robust and accurate time integration. The projection method for handling oscillations is also found to be very necessary for one of the models.

Contents

List of Figures

List of Tables

1 INTRODUCTION

The problem of interest here is the modeling and numerical simulation of diffusion of a liquid solvent into a glassy polymer. In particular, Case II diffusion where the solvent diffusion into the glassy polymer is accompanied by a glass to rubber phase transition of the polymer is considered here. In experiments, Case II diffusion is recognized by the appearance of sharp fronts of the solvent that separate a swollen rubbery gel and the solid glassy core occurring from glass–rubber transition. The fronts advance at a constant speed resulting in a linear solvent-mass uptake in time. The presence of the solvent changes the material properties such as the time scales of the relaxation mechanisms during the glass–rubber transition. These main characteristics of Case II diffusion are described in some of the experimental works such as ARGON ET AL. [1999] and THOMAS AND WINDLE [1978].

The motivation for numerical simulation of Case II diffusion comes from the lithography process in the manufacture of semiconductor chips during which topographical features are etched on the chip (see CROFFIE [1999]). In a typical process, a polymer slab used for making the chip is irradiated on its surface through a mask to selectively expose the slab to radiation. This causes the polymer chains in the exposed regions to cross-link. These cross-linked regions in turn have very low solvent diffusivity (the solvent here is called the *silylating agent*) compared to the uncross-linked regions. After the radiation step, the partially cross-linked polymer is brought into contact with the silylating agent. The silylating agent diffuses into the uncross-linked regions up to a certain depth in the slab forming a solvent-polymer mixture while the cross-linked regions due to low solvent diffusivity remain intact. These regions where the solvent-polymer mixture is formed have very different chemical properties compared to the cross-linked regions and are in turn washed away using an etching procedure. This leaves behind the cross-linked regions which in turn form the topographical features on the polymer slab. The solvent diffusion step in these sequence of steps is modeled by Case II diffusion.

From experimental results, it is clear that models for Case II diffusion should be able to capture the formation and propagation of sharp solvent fronts, swelling of the polymer due to solvent presence, and the change in material properties of the polymer. Case II models handle these requirements by coupling the diffusion and the deformation phenomena. The first fully coupled three dimensional (3-D) model for Case II diffusion was presented in GOVINDJEE [1991] where independent concentration and displacement fields were considered. Several other models for Case II diffusion also exist in the literature (see, WU AND PEPPAS [1993b], CARBONELL AND SARTI [1990], ARGON ET AL. [1999], VRENTAS AND VRENTAS [2001]). In these models, simplifications on the mechanical behavior are made to arrive at a formulation effectively in terms of concentration.

Numerical simulation of Case II phenomenon involves solving an Initial Boundary Value Problem (IBVP) numerically. The governing equations that make up the IBVP are typically a set of highly nonlinear partial differential equations in space and time. The nonlinearities in the IBVP arise from the nonlinearities in the flux law to model sharp fronts, concentra-

tion dependent relaxation times for the polymer to model glass–rubber transition and the coupling between the diffusion and deformation. These nonlinearities make the numerical implementation of these models as challenging as the development of the models itself. In many of the references mentioned earlier, not much attention was paid to the understanding of the qualitative behavior of the solutions to the IBVP in a mathematical setting. In fact, numerical methods developed without this understanding were usually too specific to the example simulations that were considered (see, for example, WU AND PEPPAS [1993b]). Furthermore, from our own experience with one of the Case II diffusion models, traditional techniques such as, a fixed stepsize Backward Euler method for time stepping and the finite element method for the spatial discretization, were found to be grossly inadequate in terms of cost efficiency and robustness. Thus, there is a strong need to understand the mathematical behavior of the solutions to the IBVP better and to develop appropriate numerical techniques.

The main objectives for the present work are the following.

- To understand the important features a model should have in order to produce Case II behavior.
- To understand the qualitative behavior of the solutions in a mathematical setting.
- To develop efficient and robust numerical schemes for solving the governing equations that incorporate these models.

To this end, two Case II diffusion models in a simple 1-D setting are considered in this report. The governing equations which include the global balance laws and constitutive relations for the two models are presented in Section 2. These equations are normalized into a dimensionless form to arrive at the Initial Boundary Value Problem (IBVP) for numerical solution. In Section 3, spatial discretization using the finite element method and time integration using two adaptive time stepping schemes is described. A collection of relevant mathematical concepts such as the maximum and minimum principles and stiffness are provided to aid the understanding of choice of numerical schemes. Various numerical issues such as stability, accuracy and robustness of time stepping schemes are highlighted. In Section 4, numerical examples for simulating Case II diffusion using the two models are provided. A careful parametric study to address the various issues during time integration of the governing equations is also presented. Finally, the work in this report is summarized and concluding remarks are made in Section 5.

2 Coupled Stress-Diffusion Models for Case II Diffusion

In this section, two examples of 1-D models for Case II diffusion are considered. The first example is a 1-D coupled field model with independent displacement and concentration fields. For this example, many of the ideas in Govindjee [1991] are used to arrive at the full set of balance laws and constitutive relations that make up the governing equations. For the second example, an existing Case II diffusion model and the accompanying equations for the balance laws, from the work of Wu and Peppas [1993a, 1993b], is considered. In contrast to the first model, the Wu and Peppas model is essentially a single field (concentration) model with assumptions on the mechanical behavior. These 1-D models are chosen to understand the various features Case II diffusion models should have and also their numerical implementation, in a simple setting.

2.1 Case II Diffusion: Model-1

The governing equations for a coupled Case II diffusion model with mechanical displacements and solvent concentrations as the independent fields are presented here. The relevant balance laws are stated, and, constitutive relations are provided to arrive at a complete set of equations that govern the model. Finally, these governing equations are normalized into a dimensionless form suitable for numerical computation. The various steps in this section clearly illustrate the general framework for modeling Case II diffusion.

2.1.1 Conservation of Mass

The equations of mass balance are recalled here from Govindjee [1991] and particularized for the 1-D case. The mass balance equations are obtained by considering a moving control volume which is closed with respect to the solid polymer skeleton while being open with respect to the liquid solvent. Let Ω be the 1-D reference configuration for the solid-liquid mixture and $\varphi : \Omega \to \mathbb{R}$ denote the deformation mapping of the solid skeleton, mapping points $X \in \Omega$ to $x \in \mathbb{R}$. It is assumed that both liquid and solid constituents coexist at points X.

The statement of Mass Balance for the polymer is given by

$$\frac{d}{dt}\int_{\varphi(\mathcal{R})} \rho \, dx = 0 \tag{1}$$

where $\mathcal{R} \subset \Omega$ and $\frac{d}{dt}(\cdot)$ is the material time derivative with respect to the velocity of the solid. Further, $\rho : \varphi(\mathcal{R}) \to \mathbb{R}^+$, is the spatial density of the solid. These equations are transformed to the reference configuration $\mathcal{R}$ by using the stretch $\lambda = \frac{\partial \varphi}{\partial X}$, and the localizing assumption is made to arrive at the following strong form of the polymer mass balance.

$$(\rho \circ \varphi)\lambda = \rho_0 \tag{2}$$

In equation (2), ρ_0 refers to the reference placement density of the solid.

Similarly, the mass balance of the liquid is given by

$$\frac{d}{dt}\int_{\varphi(\mathcal{R})} m\,dx = -\int_{\varphi(\partial\mathcal{R})} f\,n d\gamma \tag{3}$$

where $\phi(\partial\mathcal{R})$ denotes the boundary of the region of interest, m denotes the spatial concentration of the liquid, f denotes the 1-D flux of the solvent at the boundaries (with normal $n = \pm 1$) with respect to the moving control volume. Transforming the integrals to the reference configuration and using the localizing assumption, the following strong form of the solvent mass balance is obtained.

$$\frac{dM}{dt} = -\frac{df}{dX} \tag{4}$$

Here, $M = (m \circ \phi)\lambda$ denotes the reference solvent concentration, *i.e.*, the solvent mass per unit reference length.

2.1.2 Balance of Linear Momentum

The balance of linear momentum is written for the mixture as a whole. Assuming the absence of body forces and inertia effects, the balance of linear momentum in 1-D takes the form

$$\frac{d\sigma}{dx} = 0 \tag{5}$$

where the scalar σ is the 1-D stress. Equation (5) for linear momentum balance can be equivalently written in reference coordinates X as

$$\frac{d\sigma}{dX} = 0 \tag{6}$$

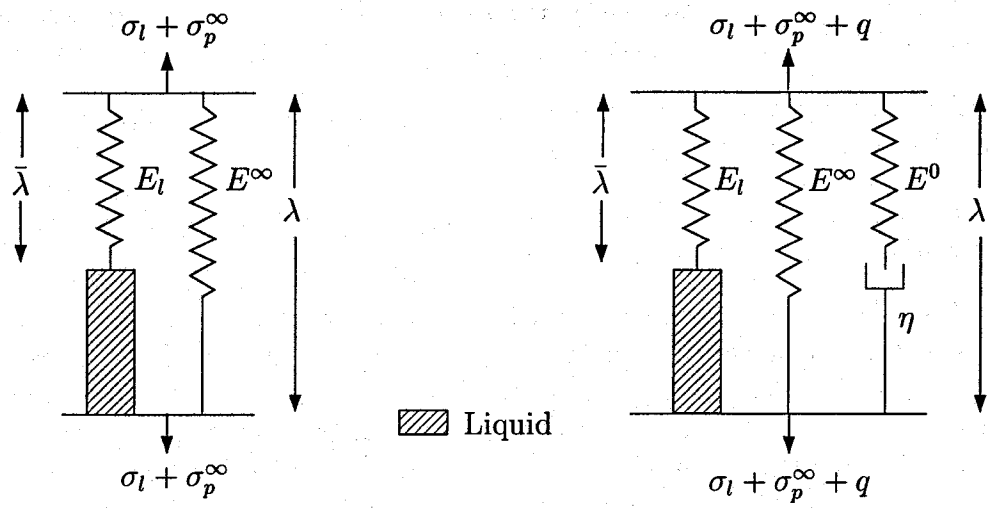
The balance of angular momentum is vacuously true in the 1-D case.

2.1.3 Constitutive Assumptions

Constitutive relations for the diffusive flux f and the stress σ complement the balance laws to form a complete set of governing equations. While the balance laws are applicable to any solid-liquid mixture, the constitutive relations define the specific Case II diffusion model of interest.

Stress-Strain relations for the Solvent-Polymer mixture

The total stress in the mixture is assumed to have contributions from the solvent and the polymer skeleton. These contributions are modeled by a simple rheological model shown in Fig. (1). For better understanding, the elastic case in Fig. (1) is considered first. For a given stretch λ, the elastic stress contribution σ_p^∞ from the polymer skeleton is assumed to



(a) Elastic Case (b) Viscoelastic Case

Figure 1: Rheological Model for Solvent-Polymer Mixture

be given by the stress in the elastic spring with spring constant E^∞. Similarly, the solvent stress contribution σ_l is assumed to be given by the stress in the spring with spring constant E_l. For calculating this stress, the spring with spring constant E_l is thought to experience an effective stretch $\bar{\lambda} = h(M)\lambda$, for a given stretch λ and a concentration M. The solvent stress is in turn calculated using the effective stretch $\bar{\lambda}$. The concentration dependent factor $h : M \to [0, 1]$ is assumed to monotonically decrease with increasing concentration and have the limiting values $h(0) = 1$ for $M = 0$, and $h \searrow 0$ as $M \to \infty$.

The actual functional forms of the dependence of σ_p^∞ on λ, and σ_l on $\bar{\lambda}$, are motivated by some of the nonlinear elastic models in OGDEN [1997]. In the present case, the following stretch dependences for the stresses are chosen.

$$\sigma_p^\infty = E^\infty(\lambda - \frac{1}{\lambda}) \tag{7a}$$

$$\begin{aligned} \sigma_l &= E_l(\bar{\lambda} - \frac{1}{\bar{\lambda}}) \\ &= E_l(\lambda h(M) - \frac{1}{\lambda h(M)}) \end{aligned} \tag{7b}$$

From (7a), σ_p^∞ is compressive(or negative) when $\lambda < 1$ and tensile when $\lambda > 1$. On the other hand, the sign of σ_l depends both on M and λ. For a fixed stretch λ, when M is large enough to make $h(M)$ small and consequently $\bar{\lambda} < 1$, the solvent stress would be negative. This is true even when $\lambda > 1$ corresponding to a swollen mixture.

The total stress σ for the elastic case is given by

$$\sigma(\lambda, M) = \sigma_l(\bar{\lambda}(\lambda, M)) + \sigma_p^\infty(\lambda) \tag{8}$$

These constitutive relations for the total stress correspond to a fully compressible solvent-polymer mixture. The total stress also satisfies the following physically motivated limits.

- For a fixed concentration M, as the stretch λ goes to zero corresponding to zero mixture length, the mixture stresses go to $-\infty$.
- For a fixed stretch λ, as $M \nearrow \infty$ corresponding to massive amounts of the solvent, the total stress goes to $-\infty$ as well.

To understand how the rheological model captures the swelling behavior, the case of (stress) free swelling is considered. For this case, the equilibrium combination of λ and M is a combination of their values which makes the total stress in the elastic case zero. As expected, $\lambda = 1$ and $M = 0$ is an equilibrium combination. Holding the stretch λ fixed, while increasing M from the equilibrium value corresponding to λ, results in a negative total stress. Therefore, the mixture expands or swells further to relieve the negative total stress and thereby producing the right swelling behavior.

The relaxation mechanisms of the polymer chains is modeled by adding viscoelasticity for calculating polymer stresses. The elastic model for the polymer is extended to a Standard Linear Solid (SLS) model by adding a spring-dashpot combination in parallel as shown in Fig. (1). The total stress is given by

$$\sigma = \sigma_l + \underbrace{\sigma_p^{\infty} + q}_{\sigma_p} \tag{9}$$

where the viscoelastic contribution q satisfies

$$\frac{dq}{dt} + \frac{q}{\tau(M)} = \beta \frac{d}{dt}[E^0(\lambda - \frac{1}{\lambda})] \tag{10}$$

In equation (10), τ denotes a concentration dependent relaxation time that models the change in the relaxation times during the glass–rubber transition, and β is a constant material parameter. The relaxation time is the ratio of the viscocity η and the spring constant E^0 in Fig. (1). The specific functional forms for the concentration dependence of relaxation time $\tau(M)$ and the stretch factor $h(M)$ will be provided in a later section.

Flux law

Recognizing that the driving force for the solvent diffusion is the chemical potential μ (CUSSLER [1984]), the flux law is assumed to be of the form

$$f = -Bm\frac{d\mu}{dx} \tag{11}$$

where B is a concentration dependent mobility coefficient and m is the spatial concentration of the solvent. Using the relation, $(m \circ \varphi)\lambda = M$, and transforming the derivatives to the

reference X coordinates, the flux law takes the form,

$$f = -B\frac{M}{\lambda^2}\frac{d\mu}{dX} \tag{12}$$

Having specified the flux law, the chemical potential alone remains to be specified to complete the model. We assume μ to be of a simple form

$$\frac{\mu}{RT} = \log(\frac{M}{M+\rho_{0l}}) + \frac{V}{g(M)\lambda} \tag{13}$$

where the coupling is introduced between the diffusion and the mechanical displacement through the stretch λ and the dimensionless coupling coefficient V. Here, ρ_{0l} is the density of the pure solvent at standard temperature and pressure, R is the universal gas constant, and T is the absolute temperature. The argument of the logarithmic term in (13) is the normalized concentration introduced in the next section. For small values of M, the normalized concentration is linear in M and the first term on the right hand side of (13) generates the standard Fick's law. The coupling is introduced through the second term. The function $g(M)$ is chosen so that $g(0) = 1$ and $g\prime(M) < 0$ in order to satisfy the following conditions

- $\frac{1}{RT}\frac{d\mu}{dM} = \frac{\rho_{0l}}{M(M+\rho_{0l})} - \frac{Vg\prime}{g^2\lambda} > 0$
- $\frac{1}{RT}\frac{d\mu}{d\lambda} = -\frac{V}{g\lambda^2} < 0$

These conditions ensure that $\mu \to \infty$ monotonically, when either of the following two situations arise: (1) $\lambda \searrow 0$ for fixed M or (2) $M \nearrow \infty$ for fixed λ. These were exactly the same two limits considered for the stress constitutive relations. Further, the $\log(\cdot)$ term in the chemical potential ensures that $\mu = -\infty$ when $M = 0$, as required (see CUSSLER [1984]).

By providing the constitutive relations for the stress response, flux, and the chemical potential, the model is complete. The specific functional forms of concentration dependent mobility coefficient and the coupling function in the chemical potential, crucial for the formation and propagation of sharp fronts, are provided later. In the meanwhile, a few remarks are in order.

Remark 1

In GOVINDJEE [1991], constitutive relations for stresses and the chemical potential are arrived at by using energy and entropy balance laws. This required the calculation of the total energy of the mixture which has two parts–energy from the elasticity of the mixture and the energy of mixing. The energy of mixing was calculated from micromechanics using Hildebrand's theory of mixing (see HILDEBRAND [1947]). This energy approach resulted in expressions for stresses and chemical potential with too many parameters. Instead, only the key features of the above model are incorporated here to obtain simpler but in a way *adhoc* constitutive relations which still produce the right behavior.

Remark 2
The functional form of the chemical potential provided here is different from GOVINDJEE [1991]. However, a micromechanical motivation for the coupling of the diffusion and swelling phenomena through the stretch (or determinant of the deformation gradient in the 3-D case) provided in the above reference can be used here. Similarly, the stress constitutive relations can also be motivated from the corresponding relations provided in the reference.

Remark 3
In some models, coupling between diffusion and mechanical deformation is introduced through the stress (or pressure in the 3-D case) dependence of the chemical potential, instead of the stretch dependence (see WEITSMAN [1987]). Model-2 presented in this report assumes the dependence of the chemical potential on a stress-like quantity.

2.1.4 NORMALIZATION OF THE GOVERNING EQUATIONS

The governing equations are normalized by first considering a dimensionless concentration $\omega \in [0, 1]$ such that

$$\omega = \frac{M}{M + \rho_{0l}} \tag{14}$$

where ρ_{0l} is the density of the pure solvent at (say) standard temperature and pressure. This normalization is chosen so that under assumptions of *ideal mixing*, *i.e.*, the total mixture volume is the sum of the volumes of the dry polymer and the solvent, ω corresponds to exactly the volume fraction of the solvent. The solvent volume fraction is the primitive variable for Model-2. The functional forms of the various coefficients and the free parameters are tabulated in table 1. These functional forms are shown to produce the Case II behavior through some numerical examples provided at the end.

	Functional Form	Parameters
Diffusion Coefficient B	$B(M(\omega)) = B_0 \exp(a_d\omega)/RT$	B_0, a_d
Coupling Function in μ	$g(M(\omega)) = \frac{1}{2}(1 - \tanh(\frac{\omega - \gamma}{\delta}))$	γ, δ
Relaxation Time τ	$\tau(M(\omega)) = \tau_0 \exp(-a_\eta\omega)$	τ_0, a_η
Stretch factor h	$h(M(\omega)) = \exp(-\frac{\zeta\omega}{1-\omega})$	ζ

Table 1: Choice of functions for diffusion coefficient, the coupling function in the chemical potential, the relaxation time, and the stretch factor for Model-1

Assuming the reference domain length of the polymer to be L and, ω_{max}, a constant, as the maximum normalized concentration in the domain, the various quantities are non-dimensionalized as follows.

$$\text{Non-dimensional Length: } \bar{X} = \frac{X}{L} \tag{15a}$$

$$\text{Non-dimensional Time: } \theta = \frac{B_0 \exp(a_d \omega_{max}) t}{L^2} \tag{15b}$$

Similarly, the non-dimensional coefficient of relaxation is given by

$$\bar{\tau_0} = \frac{B_0 \exp(a_d \omega_{max}) \tau_0}{L^2} \tag{16}$$

The coefficient $\bar{\tau_0}$ relates the time scales of the diffusion and the relaxation phenomena and is often called the *Diffusion Deborah Number.*

The dimensionless chemical potential is given by

$$\bar{\mu} = \frac{\mu}{RT} \tag{17}$$

For use in the normalized balance laws presented next, a dimensionless mobility coefficient $\bar{B}$ is defined as

$$\bar{B} = \frac{\exp(a_d(\omega - \omega_{max}))\omega}{\lambda^2(1-\omega)} \tag{18}$$

Further, the stresses and the spring moduli are scaled by a constant E which has dimensions of stress to obtain their dimensionless counterparts, *i.e.*, $\bar{(\cdot)} = (\cdot)/E$, where $(\cdot)$ is one of σ_p, σ_l, q, E^∞, E_l or E^0. Typically, a value of $E = 1$ is chosen and the values of the dimensionless spring moduli are chosen accordingly.

Finally, the dimensionless form of the governing equations in normalized reference coordinates and normalized time are summarized below.

- Conservation of the Solvent Mass

$$\frac{1}{(1-\omega)^2}\frac{d\omega}{d\theta} = \frac{d}{d\bar{X}}\left(\bar{B}(\omega,\lambda)\left[\frac{d\bar{\mu}}{d\omega}\frac{d\omega}{d\bar{X}} + \frac{d\bar{\mu}}{d\lambda}\frac{d\lambda}{d\bar{X}}\right]\right) \tag{19}$$

- Conservation of Linear Momentum

$$\frac{d\bar{\sigma}}{d\bar{X}} = 0 \tag{20}$$

- Total Stress

$$\bar{\sigma} = \bar{\sigma}_p^{\infty} + \bar{q} + \bar{\sigma}_l \tag{21}$$

- Elastic Stresses

$$\bar{\sigma}_p^{\infty} = \bar{E}^{\infty}(\lambda - \frac{1}{\lambda}) \tag{22a}$$

$$\bar{\sigma}_l = \bar{E}_l(\lambda h(M) - \frac{1}{\lambda h(M)}) \tag{22b}$$

- Viscoelastic Stress

$$\frac{d\bar{q}}{d\theta} + \frac{\bar{q}}{\bar{\tau}_0 \exp(-a_\eta \omega)} = \beta \frac{d}{d\theta}[\bar{E}^0(\lambda - \frac{1}{\lambda})] \tag{23}$$

These normalized equations are used to construct numerical approximations in later sections.

2.2 Case II Diffusion: Model-2

The model provided in the previous section considers the mechanical displacement and the solvent concentration as independent fields. However, as mentioned in the introduction, many models for Case II diffusion make assumptions about the displacements and formulate the governing equations essentially in terms of solvent concentration. The existing model of Wu and Peppas [1993a, 1993b] is chosen as an example for these single field models. Unlike in the previous section, only the important assumptions and equations are summarized while the reader is referred to the original works listed here.

2.2.1 Balance Laws and Constitutive Assumptions

The fundamental assumption in this model is that of ideal mixing. This assumption implies that the spatial solvent concentration m is proportional to the solvent volume fraction ν with the solvent density in its pure state ρ_{0l} as the proportionality constant. With this understanding, the volume fraction ν is sometimes referred to as solvent concentration in this report. The assumption of ideal mixing also implies that the solvent concentration and the displacements are no longer independent.

The spatial form of mass balance for the solvent is given by

$$\frac{\partial m}{\partial t} + \frac{d}{dx}(mV_1) = 0 \tag{24}$$

where V_1 is the spatial velocity of the solvent in 1-D, x denotes the spatial coordinate and $\partial(\cdot)/\partial t$ denotes the time derivative (not the material derivative) for a fixed x. Defining X

as the referential coordinates associated with the dry polymer, the ideal mixing assumption implies that the stretch λ satisfies

$$\lambda = \frac{dx}{dX} = \frac{1}{\nu_p} \tag{25}$$

Here, ν_p is the volume fraction of the polymer and it satisfies the relation $\nu_p + \nu = 1$. Using the relation for λ in (25), and taking the material derivative of ν_p one obtains

$$\frac{d\nu_p}{dt} = -\nu_p^2 \frac{d\lambda}{dt} = -\nu_p^2 \frac{dV_2}{dX} \tag{26}$$

where V_2 is the spatial velocity of the polymer component. Using (25), equation (26) takes the form

$$\frac{d\nu_p}{dt} = -\nu_p \frac{dV_2}{dx} \tag{27}$$

Finally, expanding the material derivative of ν_p as $d(\nu_p)/dt = \partial\nu_p/\partial t + \nu_p dV_2/dx$, the following relation is obtained.

$$\frac{\partial \nu_p}{\partial t} + \frac{d}{dx}(\nu_p V_2) = 0 \tag{28}$$

Equation (28) is exactly the mass balance of the polymer in spatial coordinates x. The important thing to note here is that the polymer mass balance is obtained as a consequence of the ideal mixing assumption and need not be specifically accounted for.

Assuming the volume average velocity $V_0 = \nu V_1 + \nu_p V_2$ to be zero, the solvent flux is given by

$$f = m(V_1 - V_0) = mV_1 \tag{29}$$

Skipping the details of the derivation (see Wu and Peppas [1993b]), the flux law is simply stated to be of the form

$$f = -\frac{D_0 \exp(a_d \nu)\nu\rho_{0l}}{(1-\nu)(1-2\chi\nu)} \frac{d\mu}{dx} \tag{30}$$

where D_0 and a_d are constants. The constant χ is the so-called Flory interaction parameter and μ is the chemical potential. Using the relation $m = \rho_{0l}\nu$ and substituting the flux law into (24), the following mass balance equation in ν is obtained.

$$\frac{\partial \nu}{\partial t} = \frac{d}{dx}\left(\frac{D_0 \exp(a_d\nu)\nu}{(1-\nu)(1-2\chi\nu)} \frac{d\mu}{dx} \right) \tag{31}$$

The chemical potential is given by

$$\mu = RT(\log(\nu) + (1-\nu) + \chi(1-\nu)^2) + \hat{V}P \tag{32}$$

where $\hat{V}$ is the coupling coefficient and P is what is termed as the swelling pressure. The statement of mass conservation for the solvent and the flux law are similar to those in model-1. However, the chemical potential depends on P, a stress quantity here whereas it depended on the stretch λ in model-1.

In the original work cited before, the swelling pressure is introduced by considering the following relation for the 3-D total stress tensor $\boldsymbol{\sigma}_{Tot}$

$$\boldsymbol{\sigma}_{Tot} = -P\mathbf{I} + \boldsymbol{\sigma} \tag{33}$$

Here, $\mathbf{I}$ is the identity tensor and P appears to be the indeterminate pressure associated with the volume constraint introduced by the ideal mixing assumption. Constitutive relations are provided for the part $\boldsymbol{\sigma}$. In the absence of body forces and inertia effects, the balance of linear momentum takes the form

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}_{Tot} = -\boldsymbol{\nabla} P + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{0} \tag{34}$$

Equation (34) is particularized to 1-D to arrive at

$$\frac{d\sigma_{Tot}}{dx} = -\frac{dP}{dx} + \frac{d\sigma}{dx} = 0 \tag{35}$$

where σ_{Tot} and σ are the 1-D counterparts of the 3-D case.

Finally, a constitutive relation in terms of a Maxwell fluid is provided for the evolution of σ.

$$\frac{\partial \sigma}{\partial t} + \frac{\sigma}{\tau} = \frac{\partial}{\partial t}(E\lambda) \tag{36}$$

where E is an elastic constant, and $\tau = \tau_0 \exp(-a_\eta \nu)$ is the volume fraction dependent relaxation time with τ_0 and a_η being constants. From the ideal mixing assumption, the relation $\lambda = 1/(1-\nu)$ is used to write the evolution of σ in terms of ν alone to obtain

$$\frac{\partial \sigma}{\partial t} + \frac{\sigma}{\tau} = \frac{\partial}{\partial t}\left(E\frac{1}{1-\nu}\right) \tag{37}$$

In an earlier work, (see COHEN AND WHITE [1991]) an evolution equation similar to (37) was introduced in an adhoc fashion without reference to momentum balance or stress constitutive relations, and the flux was made to depend on the counterpart of σ in that model. This resulted in sharp fronts. Even in the present case, the flux dependence on the gradient of P (or σ) and the evolution equation (37) for σ have a similar effect in producing sharp fronts.

The ideal mixing assumption is too restrictive in the sense that it violates the independence of concentration and displacement fields. For example, the simplest case of solvent diffusion in a polymer which is constrained to swell(or expand) by fixing the polymer at the ends cannot be handled if ideal mixing is assumed. This is because the displacement or the deformation field is predetermined from the concentration field and is not independent of it. Consequently, the only situation where ideal mixing assumption holds is that of free swelling.

Using the definition $\lambda := dx/dX$ and the assumption of ideal mixing relating λ with ν, the solvent mass balance can be transformed to X coordinates corresponding to the dry polymer. Further, using (35) to replace the derivative of P by derivative of σ one obtains,

$$\frac{1}{1-\nu}\frac{\partial \nu}{\partial t} = \frac{d}{dX}\left(\frac{D_0 \exp(a_d \nu)\nu}{(1-2\chi\nu)}\left[\frac{d\mu}{d\nu}\frac{d\nu}{dX} + \frac{d\mu}{dP}\frac{d\sigma}{dX}\right]\right) \tag{38}$$

Equations (38) and (37) are the governing equations for model-2 in terms of the single unknown field ν.

2.2.2 Normalization of the Governing Equations

As in the case of model-1, the governing equations are normalized into a dimensionless form through the following relations. Assuming the initial length of the polymer to be L and, ν_{max} (a constant) as the maximum normalized concentration in the domain, the following normalization is done.

$$\text{Non-dimensional Length: } \bar{X} = \frac{X}{L} \tag{39a}$$

$$\text{Non-dimensional Time: } \theta = \frac{D_0 \exp(a_d \nu_{max})t}{RTL^2} \tag{39b}$$

The dimensionless coefficient of the relaxation time $\bar{\tau_0}$ or the Diffusion Deborah number is given by

$$\bar{\tau_0} = \frac{D_0 \exp(a_d \nu_{max})\tau_0}{L^2} \tag{40}$$

The chemical potential is normalized as before

$$\bar{\mu} = \frac{\mu}{RT} \tag{41}$$

Having defined the dimensionless forms of length, time and the chemical potential, a dimensionless diffusion coefficient $\bar{D}$ is defined as

$$\bar{D} = \frac{\exp(a_d(\nu - \nu_{max}))\nu}{(1-2\chi\nu)(1-\nu)} \tag{42}$$

The stress and the elastic constant are rendered dimensionless using

$$\bar{\sigma} = \frac{\hat{V}\sigma}{RT} \tag{43a}$$

$$\bar{E} = \frac{\hat{V}E}{RT} \tag{43b}$$

Finally, the dimensionless forms of the governing equations used for numerical computations are summarized below

- Stress evolution equation

$$\frac{\partial \bar{\sigma}}{\partial \theta} + \frac{\bar{\sigma}}{\bar{\tau}_0 \exp(-a_\eta \nu)} = \frac{\partial}{\partial \theta}\left(\bar{E}\frac{1}{1-\nu}\right) \tag{44}$$

- Mass balance of the Solvent

$$\frac{1}{1-\nu}\frac{\partial \nu}{\partial \theta} = \frac{d}{d\bar{X}}\left(\bar{D}(\nu)(1-\nu)\left[\frac{d\bar{\mu}}{d\nu}\frac{d\nu}{d\bar{X}} + \frac{d\bar{\mu}}{dP}\frac{d\bar{\sigma}}{d\bar{X}}\right]\right) \tag{45}$$

3 Numerical Implementation

3.1 Introduction

In this section, the numerical implementation of the two Case II diffusion models will be discussed. The starting point for the numerical implementation is the normalized governing equations. Given the balance laws in strong form, the corresponding weak forms are constructed. The weak forms for the mass and momentum balance are discretized in space using the finite element method. This spatial discretization results in a system of differential-algebraic equations(DAE) in time.

The DAEs are discretized in time using suitable time stepping schemes. Two time stepping schemes, namely, a fully implicit second order Back Differentiation Formula (BDF2) and a Runge-Kutta-BDF2 implicit-explicit scheme are presented. Adaptive time stepping is incorporated into both the time stepping schemes. The various numerical issues arising from the space and time discretization of the IBVP are addressed.

3.2 Contruction of the Galerkin weak form

3.2.1 Weak form Balance Laws: Model-1

The normalized balance laws provided in section 2.1.4 are used to construct the weak form over the domain of interest [0, 1].

Define the solution space for the concentration field as

$$\mathcal{B} = \{\omega(\theta) : (0,1) \to \mathbb{R} \mid \omega(\theta) \mid_{\Gamma_\omega} = \omega_0(\theta)\} \tag{46}$$

where Γ_ω denotes the concentration Dirichlet boundary where ω is specified. Functions in $\mathcal{B}$ are assumed to have the required regularity for the integrals in the weak form to make sense (see Hughes [2000]). The space of test functions (or variations in the tangent space at a point $\omega \in \mathcal{B}$) is defined as

$$T_\omega\mathcal{B} = \{\delta\omega : (0,1) \to \mathbb{R} \mid \delta\omega \mid_{\Gamma_\omega} = 0\} \tag{47}$$

From their definition, the space of test functions satisfy the homogeneous counterpart of the Dirichlet boundary conditions.

For building the weak from, equation (19) is multiplied by an element of $T_\omega\mathcal{B}$ and integrated over the domain $[0, 1]$. The integration by parts formula is applied to the resulting integrals to obtain the weak form given below.

$$\begin{aligned} G_\omega(\omega, u; \delta\omega) = &\int_0^1 \frac{1}{(1-\omega)^2}\frac{d\omega}{d\theta}\,\delta\omega\, d\bar{X} \\ &+ \int_0^1 \bar{B}(\omega,\lambda)\left(\frac{d\bar{\mu}}{d\omega}\omega_{,\bar{X}} + \frac{d\bar{\mu}}{d\lambda}\lambda_{,\bar{X}}\right)\delta\omega_{,\bar{X}}\, d\bar{X} - \check{f}\delta\omega \mid_{\Gamma_f} = 0 \end{aligned} \tag{48}$$

where u is the displacement field, and Γ_f is the portion of the boundary where the flux is specified with $\check{f}$ as the specified flux. The flux and the concentration boundaries satisfy $\Gamma_f \bigcap \Gamma_\omega = \emptyset$ and $\Gamma_f \bigcup \Gamma_\omega = \Gamma$, where Γ is the boundary of the domain. Finally, the Galerkin weak form of the solvent mass balance reads: At each time θ, given ω_0, $\check{f}$, equation (48) is satisfied for all $\delta\omega \in T_\omega \mathcal{B}$.

In equation (48), derivatives of stretch which correspond to second derivatives of displacements need to be computed. Comments on handling these higher derivatives are made under **Remark** 4 on page 21.

The weak form for the linear momentum balance is similarly constructed by defining the solution space for the displacements as

$$\mathcal{S} = \{u(\theta) : (0,1) \to \mathbb{R} \mid u(\theta) \mid_{\Gamma_u} = u_0(\theta)\} \tag{49}$$

where Γ_u denotes the displacement Dirichlet boundary where the displacement u is specified. The space of test functions (or variations in the tangent space at a point $u \in \mathcal{S}$) is defined as

$$T_u\mathcal{S} = \{\delta u : (0,1) \to \mathbb{R} \mid \delta u \mid_{\Gamma_u} = 0\} \tag{50}$$

Multiplying (20) by an element in $T_u\mathcal{S}$, integrating over the domain $[0,1]$, and using the integration by parts formula yields the following

$$H_u(\omega, u; \delta u) = \int_0^1 \bar{\sigma}(\lambda, \omega)\, \delta u_{,\bar{X}}\, d\bar{X} - \check{q}\delta u \mid_{\Gamma_q} = 0 \tag{51}$$

Here, Γ_q is the boundary of the domain where forces are specified and $\check{q}$ is the specified force. Also, the force and the displacement boundaries satisfy $\Gamma_q \bigcap \Gamma_u = \emptyset$ and $\Gamma_q \bigcup \Gamma_u = \Gamma$. The weak form of the linear momentum balance reads: At each time θ, given u_0, $\check{q}$, equation (51) is satisfied for all $\delta u \in T_u\mathcal{S}$.

In the functionals H_u and G_ω, the dependency on the stretch λ is replaced by an equivalent dependence on u using the relation $\lambda = 1 + u_{,\bar{X}}$.

3.2.2 Weak form of Balance Laws: Model-2

The normalized solvent mass balance provided by equation (45) is used to construct the corresponding weak form. Using a similar procedure adopted in the case of Model-1, the following weak form is obtained.

$$\begin{aligned} G_\nu(\nu; \delta\nu) = &\int_0^1 \frac{1}{(1-\nu)} \frac{d\nu}{d\theta}\, \delta\nu\, d\bar{X} \\ &+ \int_0^1 \bar{D}(\nu)(1-\nu)\left(\frac{d\bar{\mu}}{d\nu}\nu_{,\bar{X}} + \frac{d\bar{\mu}}{dP}\bar{\sigma}_{,\bar{X}}\right)\delta\nu_{,\bar{X}}\, d\bar{X} - \check{f}\delta\nu \mid_{\Gamma_f} = 0 \end{aligned} \tag{52}$$

Here, the functions ν belong to the solution space $\mathcal{S}$ defined by

$$\mathcal{B} = \{\nu(\theta) : (0,1) \to \mathbb{R} \mid \nu(\theta) \mid_{\Gamma_\nu} = \nu_0(\theta)\} \tag{53}$$

where Γ_ν denotes the part of the boundary where ν is specified while Γ_f denotes the flux boundary with specified flux $\breve{f}$. As before, the flux and the concentration boundaries are disjoint. The variation $\delta\nu$ belongs to the space of test functions $T_\nu\mathcal{S}$ defined as

$$T_\nu\mathcal{B} = \{\delta\nu : (0,1) \to \mathbb{R} \mid \delta\nu \mid_{\Gamma_\nu} = 0\} \tag{54}$$

The statement of the weak form of the solvent mass balance reads: At each time θ, given ν_0, $\breve{f}$ equation (52) is satisfied for all $\delta\nu \in T_\nu\mathcal{S}$.

3.3 Spatial Discretization Using the Finite Element Method

Starting from the weak forms of the balance laws, suitable finite dimensional approximations of the solution space and the space of variations are made to spatially discretize the IBVP. This results in the Galerkin Finite Element equations. The various steps involved in the finite element discretization procedure is fairly standard and the reader are referred to, for example HUGHES [2000], for a comprehensive treatment. Only the important steps are highlighted here after introducing some notation.

3.3.1 FINITE ELEMENT DISCRETIZATION: MODEL-1

The solution space and the space of variations are approximated by standard conforming isoparametric interpolations for the displacement and the concentration fields. Starting with the space of variations $T_\omega\mathcal{B}$ for the concentration field, the finite dimensional approximation $T_\omega\mathcal{B}^h$ is defined as

$$T_\omega\mathcal{B}^h = \{\delta\omega^h \in T_\omega\mathcal{B} \mid \delta\omega_e^h = \sum_{A=1}^{n_{en}} N^A(\xi)\delta\omega^A,\ \delta\omega^A \in \mathbb{R}\} \tag{55}$$

where the superscript h's denote the quantities associated with the finite element approximation; the subscript e's denote quantities associated with the individual finite elements l_e (*i.e.*, $\delta\omega_e^h = \delta\omega^h \mid_{l_e}$); N^A are the shape functions defined on the isoparametric domain $[-1,1]$, and n_{en} is the number of nodes per element. The element coordinates $\bar{X}$ are given in terms of the isoparametric coordinates ξ by

$$\bar{X}_e^h(\xi) = \sum_{A=1}^{n_{en}} N^A(\xi)\bar{X}^A \tag{56}$$

where $\bar{X}^A$ are the coordinates of the nodes. Similarly, the solution space for the concentration field is approximated using

$$\mathcal{B}^h = \{\omega^h \mid \omega^h = \delta\omega^h + g^h,\ \delta\omega^h \in T_\omega\mathcal{B}^h \text{ and } g^h \mid_{\Gamma_\omega} = \omega_0\} \tag{57}$$

where g^h satisfies the interpolation

$$g^h = \sum_{A=1}^{n_{bn}} N^A \omega_0^A \tag{58}$$

with n_{bn} being the number of boundary nodes with Dirichlet boundary conditions and ω_0^A denoting the nodal values on the boundary nodes. Similar to the variations, the discrete solution field ω^h is interpolated in terms of the nodal values ω^A within each element, *i.e.*,

$$\omega_e^h = \sum_{A=1}^{n_{en}} N^A(\xi)\omega^A \tag{59}$$

The solution space $\mathcal{S}$ and the space of variations $T_u\mathcal{S}$ for the displacement field are approximated by their finite dimensional subspaces, $\mathcal{S}^h$ and $T_u\mathcal{S}^h$, respectively.

$$T_u\mathcal{S}^h = \{\delta u^h \in T_u\mathcal{S} \mid \delta u_e^h = \sum_{A=1}^{n_{en}} N^A(\xi)\delta u^A, \delta u^A \in \mathbb{R}\} \tag{60}$$

$$\mathcal{S}^h = \{u^h \mid u^h = \delta u^h + r^h,\ \delta u^h \in T_u\mathcal{B}^h \text{ and } r^h \mid_{\Gamma_u} = u_0\} \tag{61}$$

Here, r^h denotes the interpolation of the specified displacement on the Dirichlet boundary nodes, similar to g^h.

These approximations for the solution spaces and the space of variations for the mechanical displacements and the concentrations satisfy C^0 continuity across the inter-element boundaries.

At this point, the finite dimensional counterparts of the statements of the weak forms for solvent mass balance and balance of linear momentum can be written as

$$\begin{aligned} G_{\omega^h}(\omega^h, u^h; \delta\omega^h) = &\int_0^1 \frac{1}{(1-\omega^h)^2}\frac{d\omega^h}{d\theta}\,\delta\omega^h\, d\bar{X} \\ &+ \int_0^1 \bar{B}(\omega^h, \lambda)\left(\frac{d\bar{\mu}}{d\omega^h}\omega^h{}_{,\bar{X}} + \frac{d\bar{\mu}}{d\lambda}\lambda_{,\bar{X}}\right)\delta\omega^h_{,\bar{X}}\, d\bar{X} - \breve{f}\delta\omega^h \mid_{\Gamma_f} = 0 \end{aligned} \tag{62}$$

and

$$H_{u^h}(\omega^h, u^h; \delta u^h) = \int_0^1 \bar{\sigma}(\lambda, \omega^h)\,\delta u^h_{,\bar{X}}\, d\bar{X} - \breve{q}\delta u^h \mid_{\Gamma_q} = 0 \tag{63}$$

where the above equations are satisfied for all $\delta\omega^h$ and δu^h, respectively. The dependence on the stretch λ in some of the functions in (62) and (63) is replaced by a dependence on u^h using the relation $\lambda = 1 + u^h_{,\bar{X}}$. Defining $\boldsymbol{\delta\omega}$ and $\boldsymbol{\delta u}$ as

$$\boldsymbol{\delta\omega} = \mathop{\mathbf{A}}_{A=1}^{n_{el}} [\delta\omega^1, \delta\omega^2, \ldots, \delta\omega^{n_{en}}]^T \tag{64}$$

$$\boldsymbol{\delta u} = \mathop{\mathbf{A}}_{A=1}^{n_{el}} [\delta u^1, \delta u^2, \ldots, \delta u^{n_{en}}]^T \tag{65}$$

where $\mathbf{A}$ is the standard assembly operator over the total n_{el} number of elements for the finite element method, the following matrix form of equations (62) and (63) can be obtained.

$$\text{Mass Balance:} \quad \boldsymbol{\delta\omega}^T\left[\mathbf{M}(\boldsymbol{\omega})\frac{d\boldsymbol{\omega}}{d\theta} + \mathbf{R}(\boldsymbol{\omega},\mathbf{u}) - \mathbf{f}\right] = \mathbf{0} \tag{66a}$$

$$\text{Momentum Balance:} \quad \boldsymbol{\delta u}^T[\mathbf{S}(\boldsymbol{\omega},\mathbf{u}) - \mathbf{s}] = \mathbf{0} \tag{66b}$$

The quantities $\boldsymbol{\omega}$ and $\mathbf{u}$ are defined in the same way as $\boldsymbol{\delta\omega}$ and $\boldsymbol{\delta u}$. Further, the various matrices and vectors in equations (66a) and (66b) are defined as

$$\mathbf{M}(\boldsymbol{\omega}) = \mathop{\mathbf{A}}_{A=1}^{n_{el}} \sum_{A=1}^{n_{en}}\sum_{B=1}^{n_{en}} \int_{L_e} N^A N^B \frac{d\bar{X}}{(1-\omega^h)^2} \tag{67}$$

$$\mathbf{R}(\boldsymbol{\omega},\mathbf{u}) = \mathop{\mathbf{A}}_{A=1}^{n_{el}} \sum_{A=1}^{n_{en}} \int_{L_e} N^A_{,\bar{X}} \bar{B}(\omega^h,\lambda)\left(\frac{d\bar{\mu}}{d\omega^h}\omega^h{}_{,\bar{X}} + \frac{d\bar{\mu}}{d\lambda}\lambda_{,\bar{X}}\right) d\bar{X} \tag{68}$$

$$\mathbf{Q}(\boldsymbol{\omega},\mathbf{u}) = \mathop{\mathbf{A}}_{A=1}^{n_{el}} \sum_{A=1}^{n_{en}} \int_{L_e} N^A_{,\bar{X}} \bar{\sigma}(\lambda,\omega^h)\, d\bar{X} \tag{69}$$

Here, $\mathbf{M}$ is the concentration dependent mass matrix and $\mathbf{f}$ and $\mathbf{q}$ denote the vector corresponding to the assembly of nodal fluxes and nodal forces.

Since equations (66a) and (66b) are valid for all variations $\boldsymbol{\delta\omega}$ and $\boldsymbol{\delta u}$, they are equivalent to the following Differential Algebraic Equations.

$$\text{Mass Balance:} \quad \mathbf{M}(\boldsymbol{\omega})\frac{d\boldsymbol{\omega}}{d\theta} + \mathbf{R}(\boldsymbol{\omega},\mathbf{u}) = \mathbf{f} \tag{70}$$

$$\text{Momentum Balance:} \quad \mathbf{S}(\boldsymbol{\omega},\mathbf{u}) = \mathbf{s} \tag{71}$$

3.3.2 Finite Element Discretization: Model-2

The steps outlining the procedure for the finite element approximation of Model-1 are repeated here to obtain the corresponding approximation in the case of Model-2. The finite dimensional approximation $T_\nu\mathcal{B}^h$ for the space of variations of the volume fraction field is given by

$$T_\nu\mathcal{B}^h = \{\delta\nu^h \in T_\nu\mathcal{B} \mid \delta\nu^h_e = \sum_{A=1}^{n_{en}} N^A(\xi)\delta\nu^A,\ \delta\nu^A \in \mathbb{R}\} \tag{72}$$

where $T_\nu\mathcal{B}^h \subset T_\nu\mathcal{B}$. As before, the solution space for the volume fraction field is approximated by

$$\mathcal{B}^h = \{\nu^h \mid \nu^h = \delta\nu^h + g^h,\ \delta\nu^h \in T_\nu\mathcal{B}^h \text{ and } g^h\,|_{\Gamma_\nu} = \nu_0\} \tag{73}$$

where where g^h satisfies the interpolation

$$g^h = \sum_{A=1}^{n_{bn}} N^A \nu_0^A \tag{74}$$

The finite dimensional counterpart of the weak form of the solvent mass balance can be written as

$$\begin{aligned} G_{\nu^h}(\nu^h; \delta\nu^h) = &\int_0^1 \frac{1}{(1-\nu^h)} \frac{d\nu^h}{d\theta} \delta\nu^h \, d\bar{X} \\ &+ \int_0^1 \bar{D}(\nu^h)(1-\nu^h)\left(\frac{d\bar{\mu}}{d\nu^h}\nu^h{}_{,\bar{X}} + \frac{d\bar{\mu}}{dP}\bar{\sigma}_{,\bar{X}}\right)\delta\nu^h_{,\bar{X}} \, d\bar{X} - \breve{f}\delta\nu^h \mid_{\Gamma_f} = 0 \end{aligned} \tag{75}$$

Defining

$$\boldsymbol{\delta\nu} = \mathop{\mathbf{A}}_{A=1}^{n_{el}} [\delta\nu^1, \delta\nu^2, \ \ldots, \ \delta\nu^{n_{en}}]^T \tag{76}$$

the matrix form of (75) is obtained

$$\begin{aligned} \text{Mass Balance:} \quad & \boldsymbol{\delta\nu}^T\left[\mathbf{M}(\boldsymbol{\nu})\frac{d\boldsymbol{\nu}}{d\theta} + \mathbf{R}(\boldsymbol{\nu}, \bar{\sigma}) - \mathbf{f}\right] = \mathbf{0} \\ \Rightarrow \quad & \mathbf{M}(\boldsymbol{\nu})\frac{d\boldsymbol{\nu}}{d\theta} + \mathbf{R}(\boldsymbol{\nu}, \bar{\sigma}) = \mathbf{f} \end{aligned} \tag{77}$$

In (77), the vector $\mathbf{R}$ explicitly depends on stress field $\bar{\sigma}$. Since, the various integrals over the elements are computed using a numerical quadrature rule, it turns out that the dependence of $\mathbf{R}$ on the entire stress field $\bar{\sigma}$ can be replaced by a dependence on $\bar{\sigma}^{(i)}$s which represent the stress only at the quadrature points. The mass matrix $\mathbf{M}$ and the vector $\mathbf{R}$ in (77) are given by

$$\mathbf{M}(\boldsymbol{\nu}) = \mathop{\mathbf{A}}_{A=1}^{n_{el}} \sum_{A=1}^{n_{en}}\sum_{B=1}^{n_{en}} \int_{L_e} N^A N^B \frac{d\bar{X}}{(1-\nu^h)} \tag{78}$$

$$\mathbf{R}(\boldsymbol{\nu}) = \mathop{\mathbf{A}}_{A=1}^{n_{el}} \sum_{A=1}^{n_{en}} \int_{L_e} N^A_{,\bar{X}} \bar{D}(\nu^h)(1-\nu)\left(\frac{d\bar{\mu}}{d\nu^h}\nu^h{}_{,\bar{X}} + \frac{d\bar{\mu}}{dP}\bar{\sigma}_{,\bar{X}}\right) d\bar{X} \tag{79}$$

The system of ordinary differential equations(ODEs) (which are a special case of DAEs) in (77) are solved using a suitable time stepping scheme.

By spatial discretization of the weak forms of the balance laws, semidiscrete DAEs in time are obtained for the two models. The ODEs representing the stress evolution equations at the quadrature points are discretized in time along with the spatial balance laws. In the meanwhile, some important remarks are made here.

Remark 4
In equation (62), derivatives of the stretches λ appear. These correspond to the second derivatives of the displacement interpolation. Since the displacements are interpolated to have only C^0 continuity, these second derivatives result in jump terms at the element boundaries. Incorporating these jump terms require additional data structures which provides each element with the information such as its neighboring elements and boundaries with the neighboring elements. This also involves computing boundary integrals (for the 2-D and 3-D cases) for the jump terms which require the knowledge of the solution field in the neighboring elements. Thus, the computational structure of the standard finite element method, where the various integrals computed over an element require only the knowledge of the solution field within that element, is lost. To handle this situation, the approach of CARRANZA ET AL. [1998], where a simple H^1 projection $\bar{\lambda}$ of the stretch λ is used for computing the derivatives of the stretch in (62), is adopted.

In the case of Model-2, derivatives of stress $\bar{\sigma}$ appear in (52). However, $\bar{\sigma}$ depends on only the volume fraction ν in equation (44), and C_0 continuity of the interpolation of ν implies the same for stress $\bar{\sigma}$. In this case, stress derivatives do not require computing jump terms along the element boundaries.

Remark 5
A fixed mesh is used to spatially discretize the balance laws using the finite element method. The solution to the concentration field is a travelling sharp front in time, and using adaptive or moving meshes to resolve the sharp front as it moves seems appropriate (see, for example, MILLER [1998a, 1998b] for an introduction to moving finite elements and CARRANZA ET AL. [1998] for adaptive remeshing). However, this cannot be done so easily in our case. The stress $\bar{\sigma}$ is calculated only at the quadrature points while numerically computing the integrals in (63) and (75). While these quadrature points are fixed in both number and spatial position in a fixed mesh, new quadrature points may be created (or old ones destroyed) due to mesh (un)refinement or the quadrature points may move when the elements move. In these situations, stresses along with quantities (called the internal variables) at the quadrature points on which the stresses depend have to be advected, interpolated or extrapolated with the changing mesh. The advection, interpolation or the extrapolation procedures on the stresses and the internal variables can affect the solution accuracy adversely. Therefore, adaptive or moving meshes is not considered here.

3.4 Time Discretization of the Spatially Discrete Equations

The time integration of the spatially discrete governing equations is discussed here. The full set of governing equations for time discretization include the DAEs that result from the finite element discretization of the balance laws, and the evolution equations for the stress at the quadrature points. The DAEs arising from the balance laws are referred to as *global equations* while the stress evolution equations are referred as *local equations.*

The goal of this section is to develop efficient and accurate time stepping schemes for the

time integration of the global and the local equations. Crucial to the choice of one time stepping scheme over another is the knowledge of the qualitative behavior of the solutions in time. Towards this end, a collection of mathematical results is provided first. Following this, a fully implicit second order BDF2 stepping scheme is introduced. The BDF2 scheme requires the solution of highly nonlinear algebraic equations iteratively at each time step. Strong nonlinearity results in a very small radius of convergence for iterative methods like Newton (or Quasi-Newton) method. Thus, robustness in terms of convergence of the iterations to the solution at each time step is a must. An adaptive time stepping strategy to efficiently handle some of these issues is described.

As an alternative to the fully implicit scheme, an adaptive explicit-implicit Runge-Kutta-BDF2 scheme is developed where it is hoped that the nonlinearities are handled effectively. Following this, remarks on the efficiency, accuracy and robustness for the two time stepping schemes are made. Finally, these time stepping schemes are applied to the spatially discrete governing equations for numerical integration in time.

An itemized list of the various subsections introduced in this section along with a brief description of each subsection is provided below. This is intended to serve as a road map for the material covered in this section.

Sec. 3.4.1 In this section, a collection of mathematical results for understanding the qualitative behavior of the solutions to the governing equations is provided.

Sec. 3.4.2 Some of the mathematical understanding gained in Sec. 3.4.1 is used to motivate the specific choices of the time stepping schemes made here.

Sec. 3.4.3 A complete description of the fully implicit second order BDF2 method with adaptive time stepping on an example system of DAEs is provided in this section.

Sec. 3.4.4 Analogous to Sec. 3.4.3, a complete description of a second order explicit Runge-Kutta method with adaptive time stepping on an example system of DAEs is provided in this section.

Sec. 3.4.5 The explicit Runge-Kutta method and the fully implicit BDF2 method are combined to obtain an implicit-explicit Runge-Kutta-BDF2 partitioned time stepping scheme with adaptive stepsize control in this section.

Sec. 3.4.6 The spatially discrete governing equations incorporating Model-1 are discretized using the fully implicit and the implicit-explicit time stepping schemes in this section.

Sec. 3.4.7 The various steps followed for Model-1 in Sec. 3.4.6 are repeated for Model-1 in this section.

3.4.1 Qualitative behavior of the solutions

Maximum and Minimum Principles

For purposes of understanding, a simple nonlinear diffusion equation on the domain $\Omega = (0,1)$, with boundary conditions $w(0,\theta) = w_{max}$ and $w(1,\theta) = w_{min}$, and initial conditions $w(\bar{X},0) = 0\ \forall \bar{X} \in \Omega$, is considered

$$\mathcal{L}(u) = \frac{dw}{d\theta} - \frac{d}{d\bar{X}}(\bar{B}(w)\frac{dw}{d\bar{X}}) = 0 \tag{80}$$

Here, w denotes concentration and $\bar{B}(w)$ is a concentration dependent diffusion coefficient. The variables θ and $\bar{X}$ denote time and spatial coordinate, respectively. For the case where the coefficient $B = B_0 > 0$ is a constant, the diffusion equation satisfies a maximum principle (see John [1982 pp. 216]) which states that: when w is continuous on $\bar{\Omega} = [0,1]$ for $\theta > 0$, and the partial derivatives of w exist and are continuous on Ω, and $\mathcal{L} \leq 0$ ($\mathcal{L} = 0$ being the special case), then

$$\max_{\bar{\Omega}} w = \max_{\partial\Omega} w = w_{max} \tag{81}$$

where $\partial\Omega$ is the boundary of Ω. Replacing w by $-w$ in the proof for the maximum principle, with the requirement that $\mathcal{L} \geq 0$, one can easily show a minimum principle as well.

$$\min_{\bar{\Omega}} w = \min_{\partial\Omega} w = w_{min} \tag{82}$$

In the case of the concentration dependent diffusion coefficient, as long as $\bar{B}(w) \geq \alpha > 0$ for a constant α, and is a smooth function of w, the proof for the validity of conditions in (81) and (82) easily follows. The equations for the uncoupled case, corresponding to $V = 0$ in (19) for model-1 and $\hat{V} = 0$ in (45) for model-2, can be rewritten into the form in (80) except for the terms $1/(1-\omega)^2$ and $1/(1-\nu)$ that appear as the coefficient of the time derivatives in the corresponding equations. This situation is handled by performing through an invertible nonlinear transformation; for *e.g.*, by defining $w = 1/(1-\omega)$ in (45) and transforming the equations into the primary variable w, one can obtain the form in (80). The final result is that the uncoupled case of the two models satisfy the two conditions in (81) and (82).

These maximum and minimum principles are important as we want the numerically computed solution to satisfy them at least for the uncoupled case. Appendix B describes a projection technique used in this report to ensure the satisfaction of the maximum and the minimum principles.

Stiffness of the Resulting DAEs

The typical constraints on stepsizes while using a time stepping scheme come from requirements of stability and accuracy. Accuracy is quantified in terms of error committed per time step due to the time discretization. Stability of a time stepping scheme refers to a

requirement where the numerical solutions do not blow up in time when the exact solution to the ODE does not. An interesting situation arises when the system of ODEs are stiff. Roughly, stiffness refers to a situation where the stepsize of an explicit time stepping scheme applied to a stable ODE is governed by stability requirements rather than by accuracy. For implicit schemes, stiffness manifests as slow convergence of the iterates to the solution of nonlinear equations while using simple functional iteration. The stiffness effects manifest in the DAEs through the stiffness in the ODEs. For further details on the concepts of stability, accuracy and stiffness in the context of time stepping schemes, the reader is referred to HAIRER, NORSETT, AND WANNER [1993, vol.1], HAIRER AND WANNER [1993, vol.2] and SHAMPINE [1994]. In the discussion presented here, questions on how stiffness manifests in our situation and the bearing it has on the choice of the time stepping schemes are answered.

Stiffness arises from several sources. The first source is the spatial discretization of the weak forms to arrive at the time continuous DAEs. To understand this, a linear case in (80) where $\bar{B} = B_0$ a constant is considered. Building a weak form for this case, and discretizing in space using a uniform mesh of length h, one obtains the following system of ODEs (which accounts for the ODE part of the DAE for the fully coupled case).

$$\frac{d\mathbf{w}}{d\theta} = -\frac{1}{h^2}\mathbf{M}^{-1}\mathbf{K}\mathbf{w} + \mathbf{M}^{-1}\mathbf{f} \tag{83}$$

where the $\mathbf{M}$ is the normalized mass matrix, and $\mathbf{K}$ is the normalized flux or stiffness matrix, $\mathbf{w}$ is the vector of nodal solution values as defined before. The matrices $\mathbf{M}$ and $\mathbf{K}$ are given by

$$\mathbf{M} = \mathop{\mathbf{A}}_{A=1}^{n_{el}} \frac{1}{h}\sum_{A=1}^{n_{en}}\sum_{B=1}^{n_{en}}\int_{L_e} N^A N^B d\bar{X} \tag{84}$$

$$\mathbf{K} = \mathop{\mathbf{A}}_{A=1}^{n_{el}} B_0 h\sum_{A=1}^{n_{en}}\sum_{B=1}^{n_{en}}\int_{L_e} N^A_{,\bar{X}} N^B_{,\bar{X}} d\bar{X} \tag{85}$$

By factoring out the element length h appropriately, elements of the matrices $\mathbf{M}$ and $\mathbf{K}$ are $\mathcal{O}(1)$. The Lipschitz constant associated with the right hand side in (83) is $\mathcal{O}(\frac{1}{h^2})$ and explicit schemes experience a time step restriction due to linear stability of the form

$$\frac{B_0\Delta\theta}{h^2} \leq C \tag{86}$$

where C is $\mathcal{O}(1)$. This restriction on the time step becomes more and more severe as the mesh is refined. Also, the restriction is independent of how fast the solution is changing. For example, when the transient system is approaching the steady state, solutions don't change very rapidly and large time steps would not hurt the accuracy. This makes the time step requirement given by (86) a severe restriction. For the concentration dependent case where $B = B(w)$, the time step restriction (see LANGTANGEN [1999 pp. 348]) is roughly

$$\frac{\bar{B}_{max}\Delta\theta}{h^2} \leq C \tag{87}$$

The important point to recognize here is that the $\frac{1}{h^2}$ factor appearing in both the cases of constant and the concentration dependent diffusion coefficient. For the coupled case in Model-2, at least heuristically, it can be seem that the coupling terms would bring in a $\frac{1}{h^2}$ factor for the time step restriction. However, in Model-1 where derivatives of stretches (of the order of second derivatives of the displacement) occur, it seems that the coupling would bring in a factor of $\frac{1}{h^3}$ for the time step restriction. The analysis for the concentration dependent and the coupled case is purely heuristic and time step restriction may be more complicated than what is provided here. The time step restriction for these cases can only be inferred from actual numerical computation.

The second source of stiffness is the concentration dependent relaxation time in the stress evolution equations, (23) and (44). For concreteness, the evolution equation (44) is recalled here:

$$\frac{\partial \bar{\sigma}}{\partial \theta} + \frac{\bar{\sigma}}{\bar{\tau}_0 \exp(-a_\eta \nu)} = \frac{\partial}{\partial \theta}\left(\bar{E}\frac{1}{1-\nu}\right) \tag{88}$$

To simplify things, a fixed nonzero volume fraction at one end, and a zero volume fraction at the other is assumed as the end boundary conditions. Qualitatively, the coupled case produces sharp fronts with nearly constant volume fraction values behind the front. These fronts travel from the end with nonzero concentration towards the opposite end. Consider a spatial point $\bar{X}_0$ at which the front has passed and the volume fraction ν is close to ν_{max}. Since, ν does not change much in time, its time derivative and hence the right hand side in (88) would approach zero. At the same time, the term $\bar{\sigma}/\bar{\tau}_0 \exp(-a_\eta \nu)$ can become very large compared to $\partial\bar{\sigma}/\partial\theta$, and in such cases $\bar{\sigma}$ would approach a steady state value of zero very rapidly. When the steady state is being approached the solutions being close to zero don't change very much at all. Even in this situation, explicit schemes are governed by a time step restriction of the form

$$\frac{\Delta\theta}{\bar{\tau}_0 \exp(-a_\eta \omega_{max})} \leq C \tag{89}$$

in spite of accuracy not being a consideration. This restriction can again be severe when the denominator in (89) is very small. A similar analysis holds for the evolution of $\bar{q}$ in equation (23) in the case of Model-1.

Even though stiffness may arise due to other sources in the fully coupled case, these two sources are identified as being primary. As far as time stepping schemes are concerned, they should be able to handle these stiffness effects.

3.4.2 Choosing Time Stepping Schemes

For stiff ODEs/DAEs, implicit time stepping schemes which lead to stable numerical solutions are recommended. Implicit schemes require the solution of a nonlinear system of algebraic equations have to be solved to find the solution at each time step. Consider, the

ODE that results from the spatial discretization of (80). The resulting ODE is of the form

$$\frac{d\mathbf{w}}{d\theta} = -\frac{1}{h^2}\mathbf{M}^{-1}\mathbf{K}(\mathbf{w})\mathbf{u} + \mathbf{M}^{-1}\mathbf{f} \tag{90}$$

where again,

$$\mathbf{M} = \mathop{\mathbf{A}}_{A=1}^{n_{el}} \frac{1}{h}\sum_{A=1}^{n_{en}}\sum_{B=1}^{n_{en}}\int_{L_e} N^A N^B d\bar{X} \tag{91}$$

$$\mathbf{K} = \mathop{\mathbf{A}}_{A=1}^{n_{el}} h\sum_{A=1}^{n_{en}}\sum_{B=1}^{n_{en}}\int_{L_e} \bar{B}(\mathbf{w}) N^A_{,\bar{X}} N^B_{,\bar{X}} d\bar{X} \tag{92}$$

Solution of the nonlinear equations by a simple fixed point iteration would lead to a time step restriction of the form (see LAMBERT [1993 pp. 103])

$$\frac{\Delta\theta}{h^2} \leq C \tag{93}$$

where $1/(h^2)$ plays the role of a lipschitz constant and C is some constant. While the restriction in (86) was purely based on stability considerations, the restriction in (93) is governed by the radius of convergence of the iterative scheme. The condition in (93) is a sufficient condition and in practical situations the restriction may not be as severe as the necessary condition in (86) for explicit schemes. In the linear case where $\bar{B}(\mathbf{w}) = B_0$, the sufficient condition in (93) holds for the case of fixed point iteration. In contrast to this, Newton's Method for solving the resulting algebraic equations (linear in this case) has infinite radius of convergence with no restriction on the time step at all. In mildly nonlinear problems, the radius of convergence for the Newton's method is large and the stepsize is usually governed by accuracy requirements of the time integration. This explains the popularity of using Newton (or Quasi Newton) methods for solving implicit algebraic equations arising from stiff ODEs. However, the situation may completely change for highly nonlinear ODEs. In these situations, the radius of convergence may govern the stepsize rather than the accuracy when Newton type methods are used. This means that in addition to solving implicit equations at each time step, one is also forced to take small time steps so that the iterations in Newton type methods converge. This problem can be handled if a good initial guess for the Newton type schemes can be found. Explicit methods which are popularly used for finding predictors in predictor-corrector schemes (see, for example, HAIRER ET AL. [1993]) are not applicable because the problem is stiff. This may be the case also with extrapolation where the initial guess is an extrapolation of the solution at previously computed time steps. Continuation methods are suggested (see RHEINBOLDT [1981]) to improve the radius of convergence, where a sequence of problems of increasing nonlinearity is solved at each time step. Numerical experience with these techniques for the specific problem at hand was not very favorable and is not considered here.

From the above discussion, it is clear that for implicit schemes, the stepsize is governed by two primary considerations–accuracy of the solution, and radius of convergence of the iterative scheme to solve nonlinear equations. Regarding stability, it is assumed that the chosen implicit scheme has a large region of absolute stability. Typically A-stable (linearly stable) or B-stable (nonlinear stable) implicit methods are chosen for stiff problems (see HAIRER AND WANNER [1993] for various notions of stability and for the definition of region of absolute stability). On the other hand, all explicit schemes have a finite region of absolute stability and stiffness forces the time stepping scheme to take small time steps due to stability considerations. The relative costs for using implicit vs explicit schemes for situations where the radius of convergence governs the stepsize can only be quantified by numerical simulations.

The fully coupled problem is highly nonlinear due to nonlinear mobility/diffusion coefficient, nonlinear viscoelastic evolution equations and coupling between the swelling and the diffusion phenomena. In spite of being aware of various issues with explicit and implicit time stepping schemes for highly nonlinear problems, the only way to quantify the performance of the various schemes is through numerical simulations.

Second Order Implicit Schemes

The choice of a second order method ensures that the global error decreases as the square of the step size. Therefore, a target accuracy of the solution can be achieved with fewer steps compared to first order schemes such as Backward Euler. Order of accuracy higher than second order is not preferable as the solution may not have the required number of derivatives in time to ensure that the high order is achieved. Though this is not a concern here, extension to other models with such a restriction would not be possible.

The popular choices for the second order schemes are the family of A-stable second order Implicit-Runge-Kutta (IRK) method and the BDF2 scheme. Both these choices are recommended for the time integration of stiff ODEs as well as index 1 DAEs (see BRENAN, CAMPBELL, AND PETZOLD [1989 Ch. 3 and Ch. 4]). However, the IRK requires the solution of nonlinear algebraic equations at each stage (more than one) within a time step while BDF2 scheme requires the solution of the nonlinear equations once per time step. The second order BDF2 scheme is also A-stable just as the IRK is chosen to be. In addition to A-stability, IRK methods can be chosen to satisfy B-stability which refers to the contractivity property of the numerical method when the exact solution of the ODE system satisfies it. Numerical experience with the BDF2 method showed that the timestep restriction due to radius of convergence kept the numerical solution stable for the nonlinear problem at hand. The storage requirements in terms of solution at previous steps of BDF2 is also comparable to the storage requirements of the various stages within a time step in IRK. These above considerations favored the choice of BDF2 method for the implicit time stepping scheme.

Second Order Implicit-Explicit Schemes

As mentioned earlier, the nonlinearities (nonlinear mobility plus coupling in the mass balance equations) that create the sharp fronts may force a fully implicit scheme to take tiny steps. To handle this, an idea pursued here is to time integrate the mass balance equations at the global level using an explicit scheme, while an implicit BDF2 scheme is proposed for time integrating the local stress evolution equations. The momentum balance which forms the algebraic part of the DAE is solved implicitly for the global mechanical displacements, given the concentrations. These schemes are known as partitioned schemes and were first used for partitioning a system of ODEs into stiff and nonstiff parts and using explicit schemes for the nonstiff components and implicit schemes for the stiff components (see SODERLIND [1980] for one such application).

The reason for choosing a Runge-Kutta (RK) method for the explicit scheme is due to the motivation provided by a result in SHAMPINE [1994, pp. 302] that states: For a linear first order ODE, controlling the local error per step in an explicit RK scheme controls the stepsize to ensure stability. In the same reference, the author also remarks that this result holds true in practice for nonlinear problems as well. Thus, the stability requirements for the explicit part of the time stepping is satisfied by controlling the local error. For the implicit part, the A-stable BDF2 scheme handles the stability requirements.

In the following sections, formulas for the RK method and the BDF2 method for an abstract ODE are provided. Error estimation for time adaptivity for the two methods is described. Details of putting together the RK method and the BDF2 scheme for obtaining the implicit-explicit schemes are outlined. Finally, remarks on the connection to the actual governing equations considered here are made.

Adaptive Time Stepping

An adaptive time stepping strategy is incorporated into both the implicit BDF2 and the implicit-explicit method. This serves to reduce the time step when the stepsize is large due to considerations of accuracy or radius of convergence. But more importantly, it serves to maximize the stepsize without violating the above two considerations, thus making the time integration efficient. In the case of implicit schemes, robustness in terms of convergence of iterations while solving nonlinear equations, can be achieved by say halving the step size without time adaptivity when iterations don't converge to the solution. However, for rationally increasing the step size when it is not necessary that it is too small, a stepsize estimate is needed based on an error criterion. Therefore, adaptive time stepping for the implicit case is indispensable. For the explicit part of the implicit-explicit scheme, stability is guaranteed by controlling the local error at each time step. The control of the local error is efficiently handled by adaptive time stepping.

3.4.3 The Second Order Backward Differentiation Formula (BDF2)

The various steps for implementing the BDF2 method with adaptive time stepping is summarized in a flow chart in Fig. 2. A description of each of the steps summarized in the flow chart is provided in the following.

Time Discretization

Consider a system of ODEs of the form

$$\frac{d\mathbf{Y}}{dt} = \mathbf{F}(\mathbf{Y}) \tag{94}$$

Here, $\mathbf{Y}$ and $\mathbf{F}(\mathbf{Y})$ are vectors in $\mathbb{R}^m$ and t denotes the time. For finding the coefficients of the variable stepsize BDF2, the following expansion for the exact solution $\mathbf{Y}$ is assumed

$$\mathbf{Y}(t_{n+1}) = a\mathbf{Y}(t_n) + b\mathbf{Y}(t_{n-1}) + c\Delta t \frac{d\mathbf{Y}(t_{n+1})}{dt} + \tau_{n+1} \tag{95}$$

Here, τ_{n+1} is the Local Truncation Error (LTE) and is of $\mathcal{O}\Delta t^3$. Using Taylor expansion of the various terms in (95) about t_n, the coefficients a, b and c can be calculated in terms of the current stepsize Δt and the previous stepsize Δt_1 by setting the constant term and the coefficients of the Δt and the Δt^2 terms in the expansion to zero. Consequently,

$$a = 1 + (1 - \kappa)\frac{\Delta t}{\Delta t_1} \tag{96}$$

$$b = (\kappa - 1)\frac{\Delta t}{\Delta t_1} \tag{97}$$

$$c = \kappa \tag{98}$$

$$\tag{99}$$

where $\kappa = (\Delta t + \Delta t_1)/(2\Delta t + \Delta t_1)$. Finally, the variable stepsize BDF2 discretization for equation (94) is given by

$$\frac{\mathbf{Y}_{n+1} - \tilde{\mathbf{Y}}}{\kappa\Delta t} = \mathbf{F}(\mathbf{Y}_{n+1}) \tag{100}$$

Here, $\tilde{\mathbf{Y}} = a\mathbf{Y}_n + b\mathbf{Y}_{n-1}$ and $\mathbf{Y}_{(\cdot)}$ is the numerical solution at time $t_{(\cdot)}$. Note that the backward Euler scheme is simply obtained by setting $a = c = 1$ and $b = 0$ leading to a local truncation error of $\mathcal{O}(\Delta t^2)$. For the ODE system of the form

$$\mathbf{M}(\mathbf{Y})\frac{d\mathbf{Y}}{dt} = \mathbf{F}(\mathbf{Y}) \tag{101}$$

where $\mathbf{M}$ is invertible for all $\mathbf{Y}$, it can be easily shown that the corresponding BDF2 discretization is

$$\mathbf{M}(\mathbf{Y}_{n+1})\frac{\mathbf{Y}_{n+1} - \tilde{\mathbf{Y}}}{\kappa\Delta t} = \mathbf{F}(\mathbf{Y}_{n+1}) \tag{102}$$

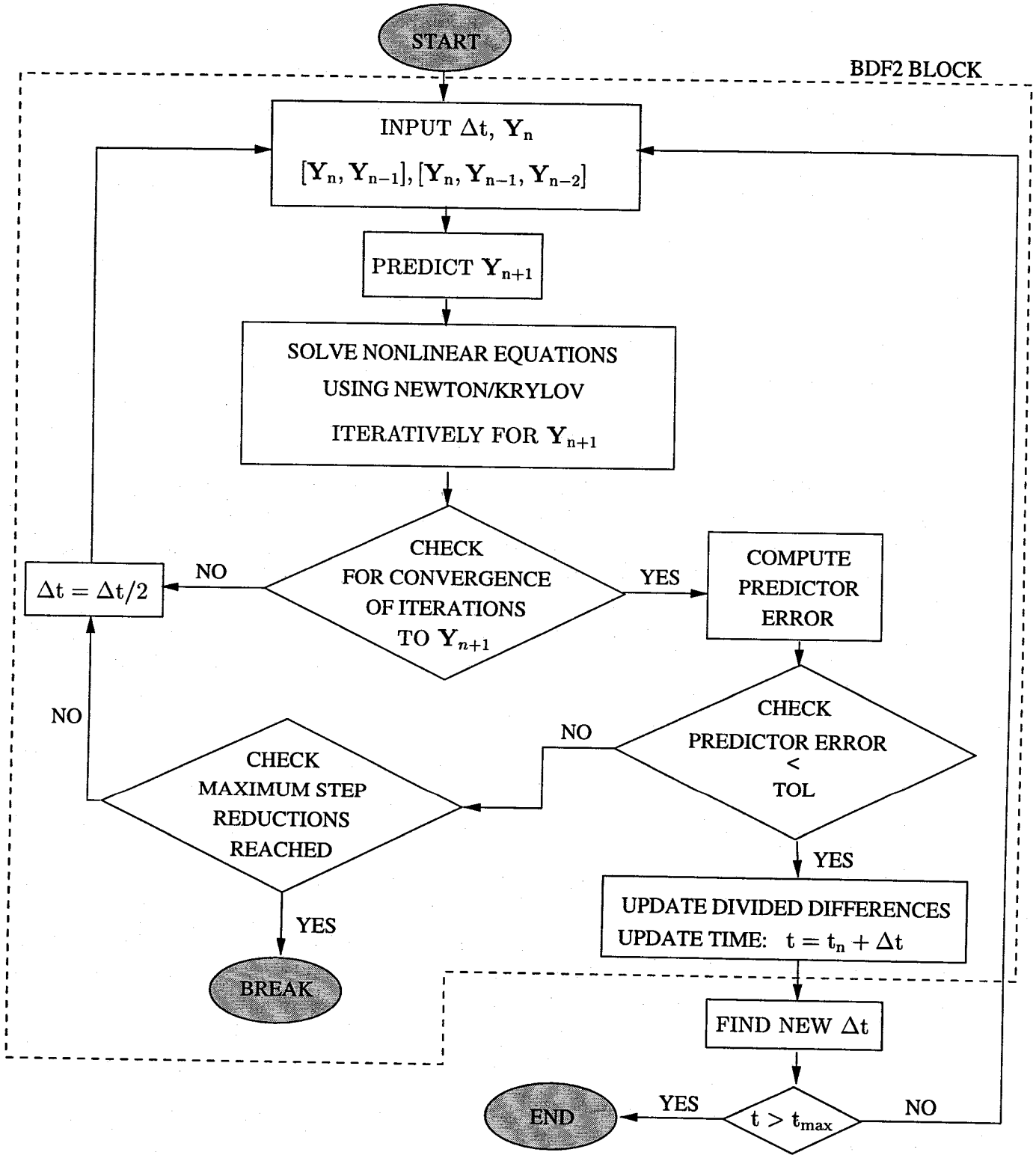


Figure 2: Flow Chart for Adaptive Second Order Backward Differentiation Formula(BDF2)

Equation (94) or (102) (as the case may be) represent the system of nonlinear algebraic equations that arise as a result of the implicit BDF2 time discretization. They are solved for $\mathbf{Y}_{n+1}$ using iterative schemes such as Newton's Method. In Appendix A, a Krylov Newton procedure and the standard Newton's method for the iterative solution of the nonlinear algebraic equations are described. The predictor for these schemes is obtained by a quadratic extrapolation of the numerical solution values $\mathbf{Y}_n$, $\mathbf{Y}_{n-1}$ and $\mathbf{Y}_{n-2}$ which are stored in a standard divided difference form.

Apriori stepsize Selection

For error control and choosing stepsizes, a predictor error defined here is used. It can be argued (see MILLER [1997]) that it is difficult to estimate the LTE term and that estimating predictor error is simple and satisfactory. Once the solution $\mathbf{Y}_{n+1}$ is found, the predictor error (PE) is computed as

$$\text{PE} = \mathbf{Y}_{n+1} - \mathbf{V}_{n+1} \tag{103}$$

where $\mathbf{V}_{n+1}$ is the quadratic extrapolation for the initial guess to Newton-type methods. It can be easily shown that this predictor error can be approximated by

$$\text{PE} = \mathbf{Y}[n+1, n, n-1, n-2](\Delta t)(\Delta t + \Delta t_1)(\Delta t + \Delta t_1 + \Delta t_2) \tag{104}$$

where $\Delta t_2 = t_{n-1} - t_{n-2}$ and $\mathbf{Y}[n+1, n, n-1, n-2]$ is the third divided difference of the numerical solution at the four time values t_{n+1}, t_n, t_{n-1}, t_{n-2}. To estimate the predictor error for the next step, it is assumed that the divided differences at t_{n+2}, t_{n+1}, t_n, t_{n-1} is the same as that at t_{n+1}, t_n, t_{n-1}, t_{n-2}. Consequently, the estimated predicted error for the next time step is

$$\text{Est. PE} = \mathbf{Y}[n+1, n, n-1, n-2](\Delta t_{new})(\Delta t_{new} + \Delta t)(\Delta t_{new} + \Delta t + \Delta t_1) \tag{105}$$

From this, the new stepsize Δt_{new} is estimated by solving the following cubic equation approximately using Newton's method with a crude tolerance.

$$(\Delta t_{new})(\Delta t_{new} + \Delta t)(\Delta t_{new} + \Delta t + \Delta t_1) = \frac{1}{\sqrt{\sum_{i=1}^{m} \left(\dfrac{\mathbf{Y}^i[n+1, n, n-1, n-2]}{rtol\,|\mathbf{Y}^i_{n+1}| + atol} \right)^2}} \tag{106}$$

Here, *atol* and *rtol* are the absolute and relative tolerances on the error in the components of the solution at the current step. The factor $rtol\,|\mathbf{Y}_i| + atol$ for each component i normalizes the predictor error into a dimensionless form. Finally, consistent with the standard practice of increasing the stepsize (see HAIRER ET AL. [1993]) by atmost a modest factor, the new stepsize is chosen as

$$\Delta t_{new} = \min[\Delta t_{new}, 4\,\Delta t] \tag{107}$$

On the other hand, if the stepsize needs to be reduced, the calculated Δt_{new} is ensured to be greater than a minimum allowed stepsize Δt_{min}.

Error Control

The predicted error given by (104) is used to decide whether to accept or reject the current step. Define the dimensionless error quantity ϵ_p as

$$\epsilon_p = \sqrt{\sum_{i=1}^{n} \left(\frac{\mathbf{Y}^i_{n+1} - \mathbf{V}^i_{n+1}}{rtol\,|\mathbf{Y}^i_{n+1}| + atol} \right)^2} \tag{108}$$

Accordingly,

- if $\epsilon_p \leq 1$, accept the current step and update the divided difference table for storing the solution values at the previous steps.
- if $\epsilon_p > 1$, reject the current step and reduce the stepsize by half for the current step and repeat the step.

Stepsizes are halved when the error criterion is not satisfied as in here or when iterations in a Newton-type scheme don't converge within a maximum of MAXITR iterations. To save computation, the iteration scheme is also terminated and the stepsize halved if the convergence rate, which is the ratio of residual error in the current iteration to that of the previous iteration, is larger than CRATE even after MAXRATE number of iterations. This way, the situation where the iterations start to diverge is detected early. In the case of Krylov Newton, the Jacobian is not updated freshly at every time step. Instead, it is updated only when the iterations do not converge in MAXITR iterations while using an old Jacobian. A limit MAXTRY on the number of times the stepsize is allowed to be halved within a time step is specified. Exceeding MAXTRY number of stepsize reductions in a time step forces the program to halt. The iteration sequence is considered successful if the norm of the increment in an iteration is below a tolerance TOL. In this case, the satisfaction of the error criterion is checked before proceeding to the next step. These additional features in the implementation make the computation efficient and robust.

3.4.4 The Second Order Explicit Runge Kutta Method (ERK2)

As in the case of the BDF2 scheme, the various steps for the partitioned implicit-explicit schemes are summarized in a flow chart in Fig. 4. A description of each of the steps summarized in the flow chart is provided next.

Consider a system of ODEs of the form

$$\frac{d\mathbf{Z}}{dt} = \mathbf{G}(\mathbf{Z}) \tag{109}$$

where $\mathbf{Z}$ and $\mathbf{G}(\mathbf{Z})$ are vectors in $\mathbb{R}^n$ and t denotes the time, as before. The Explicit Runge-Kutta method for this system of ODEs is given by its Butcher array as shown in table 2. In table 2 only three stage are considered as we are interested in second order methods.

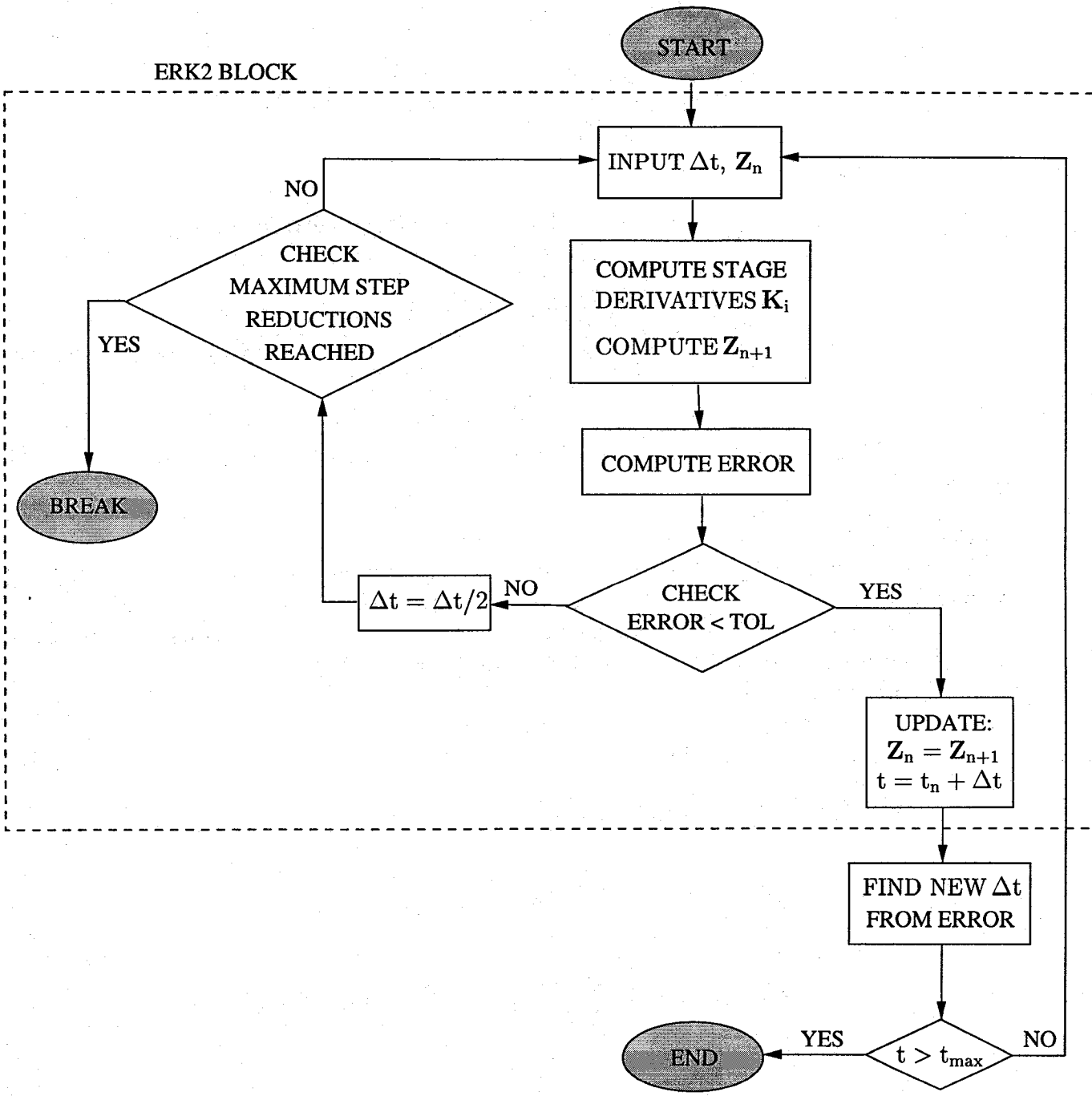


Figure 3: Flow Chart for Adaptive Second Order Runge-Kutta Method (ERK2)

c_1	0		
c_2	α_{21}	0	
c_3	α_{31}	α_{32}	0
1	b_1	b_2	b_3
1	$\hat{b}_1$	$\hat{b}_2$	$\hat{b}_3$

where $\sum_{j=1}^{3} \alpha_{ij} = c_i$

Table 2: Butcher Array for Explicit Runge-Kutta Methods

First order Method	Second Order Method
$\hat{b}_1 + \hat{b}_2 + \hat{b}_3 = 1$	$b_1 + b_2 + b_3 = 1$
	$b_1c_1 + b_2c_2 + b_3c_3 = 1/2$

Table 3: Order conditions for First and Second Order ERK Methods

In fact, the unknown coefficients in the three stages provide enough flexibility to obtain a second order method with a first order method embedded in it for error control.

The conditions on the coefficients in the butcher array for first and second order methods are recalled from HAIRER ET AL. [1993] and are listed in table 3.

The ERK discretization is given by

$$\mathbf{K}_1 = \mathbf{G}(\mathbf{Z}_n) \tag{110}$$

$$\mathbf{K}_i = \mathbf{G}(\mathbf{Z}_n + \Delta t \sum_{j=1}^{i-1} \alpha_{ij}\mathbf{K}_j)$$

$$\mathbf{Z}_{n+1}^{(2)} = \mathbf{Z}_n + \Delta t \sum_{j=1}^{3} b_j\mathbf{K}_j \quad \text{(Second order solution)} \tag{111}$$

$$\mathbf{Z}_{n+1}^{(1)} = \mathbf{Z}_n + \Delta t \sum_{j=1}^{3} \hat{b}_j\mathbf{K}_j \quad \text{(First order solution)}$$

(112)

where $\mathbf{K}_i$s are called the *stage derivatives* at the stage points $c_i\Delta t$ within a time step, and $\mathbf{Z}_{n+1}^{(1)}$ denotes the first order solution and $\mathbf{Z}_{n+1}^{(2)}$ denotes the second order solution, respectively. The solution at the next time step is given in terms of the solution at the previous step and a linear combination of the stage derivatives. The local error (LE) used in error control is

0	0		
2/3	2/3	0	
1	1/4	3/4	0
1	1/4	3/4	0
1	2/3	0	1/3

Table 4: A Choice of coefficients in the Butcher array

calculated by the difference

$$\mathrm{LE} = \mathbf{Z}_{n+1}^{(2)} - \mathbf{Z}_{n+1}^{(1)} \tag{113}$$

The actual coefficients of the Butcher array satisfying the order conditions in table 3 are listed in table 4. Besides being simple, the coefficients are fully compatible with the extensive set of guidelines on developing Runge-Kutta methods provided in SHAMPINE [1994, Sec. 5.3]. Most notably, these coefficients render a very accurate estimate of the Local Error which is $\mathcal{O}(\Delta t^2)$ by minimizing the coefficients of the $\mathcal{O}(\Delta t^3)$ term in the error (*i.e.*, the coefficients almost satisfy the conditions for the method to be of third order).

For the case

$$\mathbf{M}(\mathbf{Z})\frac{d\mathbf{Z}}{dt} = \mathbf{G}(\mathbf{Z}) \tag{114}$$

where $\mathbf{M} \in \mathbb{R}^{n\times n}$ invertible for all $\mathbf{Z}$, the stage derivatives are computed using

$$M(\mathbf{Z}_n)\mathbf{K}_1 = \mathbf{G}(\mathbf{Z}_n) \tag{115}$$

$$M(\mathbf{Z}_n + \Delta t\sum_{j=1}^{i-1}\alpha_{ij}\mathbf{K}_j)\mathbf{K}_i = \mathbf{G}(\mathbf{Z}_n + \Delta t\sum_{j=1}^{i-1}\alpha_{ij}\mathbf{K}_j) \tag{116}$$

and the rest of the procedure for calculating $\mathbf{Z}_{n+1}$ and the local error, remains the same.

Error Control and Time adaptivity

Similar to the normalization of the prediction error in the BDF2 scheme, the local error can be normalized to obtain a dimensionless error ϵ_p.

$$\epsilon_p = \sqrt{\sum_{i=1}^{m}\left(\frac{LE^{(i)}}{rtol\,|\mathbf{Z}_{n+1}^i| + atol}\right)^2} \tag{117}$$

where *atol* and *rtol* were defined earlier. However, in this report, an alternate normalization recommended in SHAMPINE AND REICHELT [1997] along with the $\|\cdot\|_\infty$ for calculating ϵ_p is used.

$$\mathbf{V}^i = \frac{\mathrm{LE}^{(i)}}{\max[rtol\,|\mathbf{Z}^i_{n+1}|, atol]} \tag{118}$$

$$\epsilon_p = \|\mathbf{V}\|_\infty \tag{119}$$

Once ϵ_p is computed, the step is accepted if $\epsilon_p \leq 1$ and the new step size is estimated using

$$\Delta t_{new} = \min[5\,\Delta t, 0.8\,\epsilon_p^{-\frac{1}{p}}] \tag{120}$$

where p denotes the order of the method which here is 2. The factor 0.8 ensures that the next step would not fail just because it is a little too big. If $\epsilon_p > 1$, the step is rejected and the stepsize is halved before the step is repeated. A maximum MAXTRY number of stepsize reductions are allowed within a time step. In the case of stepsize reduction or choosing a new Δt, the stepsize is ensured to remain greater than a lower bound Δt_{min}. The various steps for the ERK method are summarized through the flow chart in Fig. 3.

3.4.5 PARTITIONED IMPLICIT-EXPLICIT SCHEMES

In this section, the details of combining the fully implicit BDF2 scheme and the second order ERK scheme by partitioning a system of ODEs are discussed. These details closely follow those presented in SODERLIND [1980].

Consider a system of ODEs which are partitioned into the form

$$\frac{d}{dt}\begin{pmatrix}\mathbf{Z}\\ \mathbf{Y}\end{pmatrix} = \begin{pmatrix}\mathbf{G}(\mathbf{Z},\mathbf{Y})\\ \mathbf{F}(\mathbf{Z},\mathbf{Y})\end{pmatrix} \tag{121}$$

where $\mathbf{Z}$ and $\mathbf{G}(\mathbf{Z},\mathbf{Y})$ are vectors in $\mathbb{R}^n$, while $\mathbf{Y}$ and $\mathbf{F}(\mathbf{Z},\mathbf{Y})$ are vectors in $\mathbb{R}^m$. Further, the vector $\mathbf{X} = [\mathbf{Z},\ \mathbf{Y}]^T \in \mathbb{R}^{n+m}$ defines the full system of ODEs. The partition of the ODE in $d\mathbf{Z}/dt$ is discretized using the ERK2 scheme with control of the local error, while that in $d\mathbf{Y}/dt$ is discretized using BDF2.

Explicit Second Order Runge-Kutta Step

As before, the ERK2 step is given by

$$\mathbf{K}_1 = \mathbf{G}(\mathbf{Z}_n, \mathbf{Y}_n) \tag{122}$$

$$\mathbf{K}_i = \mathbf{G}(\mathbf{Z}_n + \Delta t \sum_{j=1}^{i-1} \alpha_{ij}\mathbf{K}_j, \mathbf{Y}_{n+c_i})$$

$$\mathbf{Z}_{n+1}^{(2)} = \mathbf{Z}_n + \Delta t \sum_{j=1}^{3} b_j \mathbf{K}_j \quad \text{(Second order solution)} \tag{123}$$

$$\mathbf{Z}_{n+1}^{(1)} = \mathbf{Z}_n + \Delta t \sum_{j=1}^{3} \hat{b}_j \mathbf{K}_j \quad \text{(First order solution)}$$

(124)

The only change is that the quantities $\mathbf{Y}_{n+c_i}$, corresponding to the values $\mathbf{Y}$ at the stage points $t_n + c_i\Delta t$, are required. The $\mathbf{Y}$ system has solution values at the previous steps stored because of the multistep BDF2 scheme used for it. Using a linear polynomial $Q(t)$ through $\mathbf{Y}_n$ and $\mathbf{Y}_{n-1}$, the values $\mathbf{Y}_{n+c_i}$ are predicted by extrapolation. In other words, $\mathbf{Y}_{n+c_i}$ are replaced by $Q(t_n + c_i\Delta t)$ in (122). This replacement by extrapolation is equivalent to taking an ordinary ERK2 step on the system

$$\frac{d\mathbf{Z}}{dt} = \mathbf{G}(\mathbf{Z}, Q(t)) \tag{125}$$

The error associated with the extrapolation in $Q(t)$ is $\mathcal{O}(\Delta t^2)$. Further, the BDF2 step implies that the values $\mathbf{Y}$s themselves have a global error of $\mathcal{O}(\Delta t^2)$. Assuming that $\partial G/\partial \mathbf{Y}$ is continuous, the $\mathcal{O}(\Delta t^2)$ error from the extrapolation and the global error in $\mathbf{Y}$ result in $\mathcal{O}(\Delta t^3)$ perturbations in computing $\mathbf{Z}_{n+1}$. This is due to an additional of factor Δt that is picked while computing $\mathbf{Z}_{n+1}$ from $\mathbf{K}_i$s and $\mathbf{Z}_n$. Therefore, the extra error caused by extrapolation and the global error in $\mathbf{Y}$ is of the same order of magnitude as the local truncation error associated with the ERK2 scheme. The case where the ODE system in $\mathbf{Z}$ is of the form

$$\mathbf{M}(\mathbf{Z})\frac{d\mathbf{Z}}{dt} = \mathbf{G}(\mathbf{Z}, \mathbf{Y}) \tag{126}$$

is handled as before.

Implicit Second Order BDF2 Step

The time discretization for the BDF2 step is given by

$$\frac{\mathbf{Y}_{n+1} - \tilde{\mathbf{Y}}}{\kappa\Delta t} = \mathbf{F}(\mathbf{Z}_{n+1}, \mathbf{Y}_{n+1}) \tag{127}$$

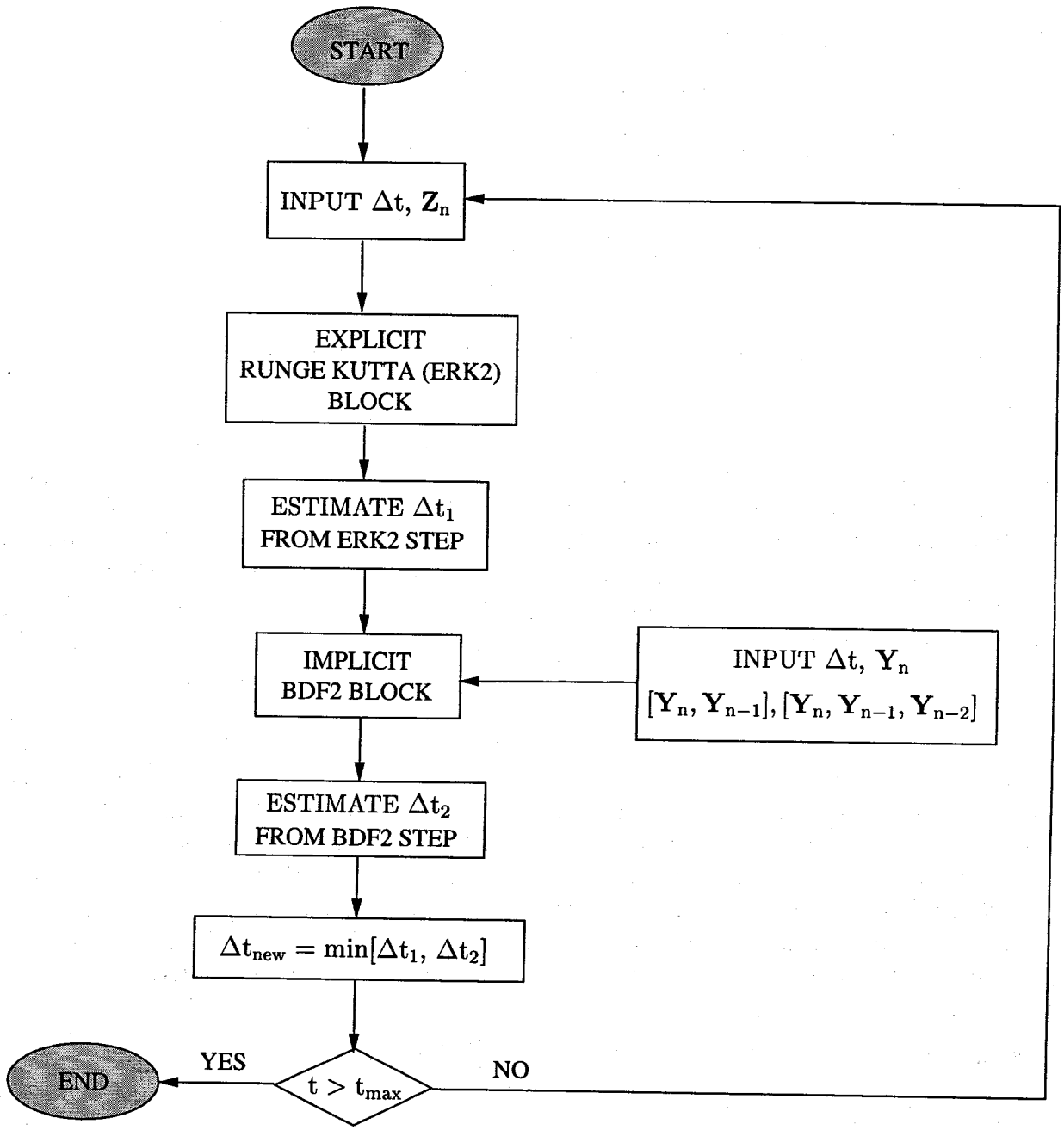


Figure 4: Flow Chart for the Implicit-Explicit ERK2-BDF2 Scheme

where $\tilde{\mathbf{Y}} = a\mathbf{Y}_n + b\mathbf{Y}_{n-1}$ and the coefficients a and b are defined in (96), and $\mathbf{Z}_{n+1}$ is already computed from the ERK2 step. Substituting the exact solution $\mathbf{Y}(t_{n+1})$, $\mathbf{Y}(t_n)$, $\mathbf{Y}(t_{n-1})$ and $\mathbf{Z}(t_{n+1})$ into (127), and performing a simple consistency analysis (see LAMBERT [1993]), it can easily be shown that the local truncation error for the BDF2 step is also $\mathcal{O}(\Delta t^3)$.

From the results on the explicit and the implicit steps, it is clear that the local truncation errors in the $\mathbf{Z}$ and $\mathbf{Y}$ systems individually are of $\mathcal{O}(\Delta t^3)$. This implies a $\mathcal{O}(\Delta t^3)$ local truncation error in the full system in $\mathbf{X}$. The result is a partitioned scheme that is globally of second order.

3.4.6 Time Stepping Schemes for Model-1

Fully Implicit BDF2 method

The spatially-discrete equations in (70) corresponding to mass and momentum balance, along with the stress evolution equations (23) at the quadrature points are discretized in time to obtain

$$\text{Mass Balance:} \quad \mathbf{M}(\boldsymbol{\omega}_{n+1})\frac{\boldsymbol{\omega}_{n+1} - \tilde{\boldsymbol{\omega}}}{\kappa\Delta t} + \mathbf{R}(\boldsymbol{\omega}_{n+1}, \mathbf{u}_{n+1}, \tilde{\bar{\boldsymbol{\lambda}}}) = \mathbf{f} \tag{128a}$$

$$\text{Momentum Balance:} \quad \mathbf{S}(\boldsymbol{\omega}_{n+1}, \mathbf{u}_{n+1}) = \mathbf{s} \tag{128b}$$

$$\text{Stress Evolution:} \quad \frac{\bar{q}^{(i)}_{n+1} - \tilde{\bar{q}}^{(i)}}{\kappa\Delta t} + \frac{\bar{q}^{(i)}_{n+1}}{\bar{\tau}_0 \exp(-a_\eta \omega^{(i)}_{n+1})} = \beta\left(1 + \frac{1}{(\lambda^{(i)}_{n+1})^2}\right)\left(\frac{\lambda^{(i)}_{n+1} - \tilde{\lambda}^{(i)}}{\kappa\Delta t}\right) \tag{128c}$$

where $\tilde{\bar{q}}^{(i)} = a\bar{q}^{(i)}_n + b\bar{q}^{(i)}_{n-1}$ and $\tilde{\lambda}^{(i)} = a\lambda^{(i)}_n + b\lambda^{(i)}_{n-1}$, and i denotes the index for the quadrature points.

The concentration $\omega^{(i)}_{n+1}$ at the i^{th} quadrature point is obtained by interpolating the nodal concentration values of the element that contains the quadrature point through an expansion in terms of the nodal shape functions. Similarly, the stretch $\lambda^{(i)}_{n+1}$ at the quadrature points are obtained by taking the derivative of the interpolating displacement field at the quadrature point. The stretches $\lambda^{(i)}_n$, $\lambda^{(i)}_{n-1}$, and the internal variables $\bar{q}^{(i)}_n$ and $\bar{q}^{(i)}_{n-1}$ required for calculating $\tilde{\lambda}^{(i)}$ and $\tilde{\bar{q}}^{(i)}$ are stored as history at each quadrature point. The quantity $\tilde{\bar{\boldsymbol{\lambda}}}$ appearing in the argument list in (128a) is explained next.

Recalling Remark 4 on page 21, the weak form of the mass balance depends on the derivative of the stretch. To handle this, a discrete H^1 projection of the stretches is computed, *i.e.*,

$$\bar{\boldsymbol{\lambda}} = \mathbf{H}^{-1}\left(\mathop{\mathbf{A}}_{1}^{n_{el}} \sum_{A=1}^{n_{en}} \int_{L_e} N^A \lambda \, d\bar{X}\right) \tag{129}$$

where $\bar{\boldsymbol{\lambda}}$ denotes the nodal values of stretch interpolation. The projection matrix $\mathbf{H}$ is given

by

$$\mathbf{H} = \mathop{\mathbf{A}}_{1}^{n_{el}} \sum_{A=1}^{n_{en}} \sum_{B=1}^{n_{en}} \int_{L_e} N^A N^B \, d\bar{X} \tag{130}$$

The stretch derivatives at time t_{n+1} are replaced by the extrapolation $\tilde{\bar{\boldsymbol{\lambda}}}$ of the nodal projections $\bar{\boldsymbol{\lambda}}_n$ and $\bar{\boldsymbol{\lambda}}_{n-1}$ at time t_n and t_{n-1}, respectively.

The nonlinear algebraic equations in (128a),(128b) and (128c) are solved iteratively by Newton (or Krylov Newton) scheme. Stepsize adaptivity and error control are accomplished by specifying the individual tolerances $atol_u$, $rtol_u$ for the displacement vector $\mathbf{u}$, $atol_\omega$, $rtol_\omega$ for the concentration vector $\boldsymbol{\omega}$, and $atol_q^{(i)}$, $rtol_q^{(i)}$ for stress at each of the quadrature points. Using these tolerances, efficient adaptive time stepping is incorporated by requiring that the error criterion for each type (displacement, concentration, stress at each quadrature point) of the unknown quantities is satisfied.

Partitioned Implicit-Explicit method

The time discretization for the stress evolution and the momentum balance remains the same as in equations (128b) and (128c). The mass balance equation in (128a) is discretized using ERK2. This requires values of displacements $\mathbf{u}$ and nodal stretch projections $\boldsymbol{\lambda}$ at the intermediate stage points. They are calculated by extrapolating the solution values, $\mathbf{u}_n$, $\mathbf{u}_{n-1}$ for $\mathbf{u}$ and $\boldsymbol{\lambda}_n$, $\boldsymbol{\lambda}_{n-1}$ for $\boldsymbol{\lambda}$, at the previous time steps as discussed in Sec. ??.

From an earlier discussion, it seems that the stiffness would severely restrict the stepsize in the explicit part of the partitioned method and in turn the overall scheme. Therefore, the question arises as to why this scheme is considered at all. The reason for considering the partitioned scheme is twofold: a) to evaluate the efficiency of the scheme b) to provide a benchmark for comparing against the efficiency of fully implicit adaptive schemes like the BDF2 method.

3.4.7 Time Stepping Schemes for Model-2

Fully Implicit BDF2 method

The spatially-discrete equations in (77) corresponding to mass balance, along with the stress evolution equations (44) at the quadrature points are discretized in time to obtain

$$\text{Mass Balance:} \quad \mathbf{M}(\boldsymbol{\nu}_{n+1}) \frac{\boldsymbol{\nu}_{n+1} - \tilde{\boldsymbol{\nu}}}{\kappa \Delta t} + \mathbf{R}(\boldsymbol{\nu}_{n+1}, \bar{\sigma}_{n+1}^{(i)}) = \mathbf{f} \tag{131a}$$

$$\text{Stress Evolution:} \quad \frac{\bar{\sigma}_{n+1}^{(i)} - \tilde{\bar{\sigma}}^{(i)}}{\kappa \Delta t} + \frac{\bar{\sigma}_{n+1}^{(i)}}{\bar{\tau}_0 \exp(-a_\eta \nu_{n+1}^{(i)})} = \bar{E} \left(\frac{1}{1 - (\nu_{n+1}^{(i)})^2} \right) \left(\frac{\nu_{n+1}^{(i)} - \tilde{\nu}^{(i)}}{\kappa \Delta t} \right) \tag{131b}$$

As before, the stress evolution equation at each quadrature point is indexed by i in equation (131b). The volume fraction $\nu_{n+1}^{(i)}$ at the i^{th} quadrature point is obtained by interpolating

the nodal volume fraction values of the element that contains the quadrature point through the isoparametric mapping. The stresses $\bar{\sigma}_n^{(i)}$ and $\bar{\sigma}_{n-1}^{(i)}$ required for calculating $\tilde{\bar{\sigma}}^{(i)}$ are stored as history at each quadrature point.

The nonlinear algebraic equations in (131a) and(131b) are solved iteratively by Newton or Krylov Newton scheme. Stepsize adaptivity and error control are accomplished by specifying the individual tolerances $atol_\nu$, $rtol_\nu$ for the concentration vector $\boldsymbol{\nu}$, and $atol_\sigma^{(i)}$, $rtol_\sigma^{(i)}$ for stress at each of the quadrature points

Partitioned Implicit-Explicit method

The time discretization for the stress evolution remains the same as in equation (131b). The mass balance equation in (131a) is discretized using ERK2. This requires values of stress at quadrature points $\bar{\sigma}^{(i)}$ at the intermediate stage points in a time step. They are calculated by extrapolating the quadrature stress values, $\bar{\sigma}_n^{(i)}$, $\bar{\sigma}_{n-1}^{(i)}$ at the previous time steps.

3.5 Linearization

The implicit BDF2 scheme and the implicit part of the partitioned Implicit-Explicit method, result in nonlinear algebraic equations that needs to be solved. They are solved using Newton or Krylov Newton schemes (see Appendix A). These schemes require the Jacobian or the derivative of the system of nonlinear equations. Although cumbersome, the linearization procedure for finding the Jacobian matrix is fairly standard and will not be discussed here. It will only be mentioned that the Jacobian matrix for the fully implicit BDF2 scheme turns out to be unsymmetric for both the models. The Jacobian matrix corresponding to the implicit part of the partitioned scheme turns out to be symmetric.

4 Discussion of Numerical Results

Numerical results from the time integration of the spatially discretized governing equations for the two Case II diffusion models are presented here. The performance of the fully implicit and the partitioned implicit-explicit time integration schemes developed earlier are compared based on their ability to handle the various numerical issues discussed in the previous section. Furthermore, the importance of using adaptive time stepping for our problem is illustrated.

4.1 Numerical Results for Model-1

The numerical simulation of Case II diffusion using Model-1 is considered first. The material properties chosen for this model are listed in Table 5. A polymer of dimensionless reference length $\bar{L} = 1$ is considered as the domain of interest for the IBVP. The boundary conditions for the diffusion problem are: a fixed concentration $\omega \mid_{\bar{X}=0} = 0.3832$ at the left end, and a zero flux $\check{f} \mid_{\bar{X}=1} = 0$ at the right end. Correspondingly, the maximum concentration ω_{max} occurs that the left end boundary with the value $\omega_{max} = 0.3832$. Although the maximum principle was not proved for the fully coupled problem, the sign of the flux in the coupled case (on which the proof depends) can be seen to be the same as that of the uncoupled case thereby implying, at least heuristically, the satisfaction of the maximum principle. Numerical simulations (not presented here) for very fine spatial discretizations and tiny time steps over a short time also confirm this implication. Given the value of ω_{max}, a value of $a_d = 20$ corresponds to a change in the normalized mobility $\bar{B}$ by three orders of magnitude which models sharp changes in mobility in the presence of the solvent. The coefficient $a_\eta = 50$ corresponds to a decrease in the the relaxation time $\bar{\tau}$ by nine orders of magnitude for modeling the glass to rubber transition. For the mechanical problem, the bar is assumed to be fixed at the right end, *i.e.*, $u \mid_{\bar{X}=1} = 0$, and free to expand at the left end. The mechanical boundary conditions correspond to those in free swelling. Further, zero displacements, zero concentrations and zero stresses in the polymer are chosen as the initial conditions for the IBVP.

The IBVP incorporating Model-1 is discretized in space with 60 elements using the finite element method. The resulting DAEs are solved using the fully implicit BDF2 scheme and the partitioned ERK2-BDF2 scheme with adaptive time stepping. The tolerances $atol_u = 5 \times 10^{-3}$, $rtol_u = 5 \times 10^{-2}$, $atol_\omega = 5 \times 10^{-3}$ and $rtol_\omega = 5 \times 10^{-2}$ are chosen for adaptive time stepping in both the BDF2 and ERK2-BDF2 schemes. The tolerances $atol_q^{(i)}$ and $rtol_q^{(i)}$ for integration of the stress at each quadrature point (Gauss point) are set to a high value so that the stepsizes don't depend on them (see Remark 6). For the solution of the nonlinear equations, the values MAXTRY = 9, MAXITR = 8, TOL = 10^{-13} and MAXFAIL = 4 were chosen. In the Krylov Newton Method, the maximum number of vectors in the subspace $\mathcal{V}$ (see Appendix A) is set to 4. Old Krylov vectors are discarded if the sine of the angle between any two vectors is smaller than 0.001.

In Fig. 5, a numerical simulation illustrating the formation and propagation of sharp fronts

Parameter	Value	Parameter	Value
a_d	20	$\bar{E}^{\infty}$	40
a_η	50	$\bar{E}_l$	200
$\bar{V}$	1	$\bar{E}^0$	40
γ	0.36	ζ	0.52
δ	0.03	β	10
$\bar{\tau}_0$	2.5	–	–

Table 5: Material Properties for Model-1

using the BDF2 and the ERK2-BDF2 schemes is presented. In these plots, profiles of the concentration ratio M/M_{max}, calculated from ω/ω_{max}, are plotted on the deformed length of the polymer at every 0.015 units of time. The entire simulation takes 0.33 units of time. From the plots, it is clear that the fronts remain sharp and, except at the beginning, move almost at constant speed. Due to swelling, the free (left) end of the polymer moves to the left corresponding to a steady-state elongation of the total length of the polymer by 30%. At steady state, the solvent concentration ω takes the value ω_{max} everywhere, consistent with the zero flux boundary condition at the right end.

The BDF2 and the ERK2-BDF2 schemes for the full simulation in Fig. 5 are compared in Table 6. Newton's method is used to solve the implicit nonlinear equations that arise at each time step in these schemes. While the implicit scheme required just 315 steps for the full simulation, the partitioned scheme required as many as 67788 steps. From the discussion in the previous section, a result like this is not totally unexpected. The stability requirements due to stiffness indeed places very severe stepsize restrictions on the explicit ERK2 method and in turn on the overall partitioned scheme. It turns out that in this problem, the reduction in the radius of convergence due to the nonlinearities while using the fully implicit BDF2 is not very significant. The stepsizes in the BDF2 case are in fact several orders larger than those in the partitioned scheme thereby making the partitioned scheme quite ineffective here.

For assessing the accuracy of the partitioned scheme, the concentration profiles at various time values obtained from the BDF2 method and ERK2-BDF2 scheme are plotted in Fig. 6. The mesh size, material parameters, and the various tolerances are exactly the same as before. In the figure, the concentration profiles obtained from the partitioned scheme are indistinguishable from those obtained from the A-stable BDF2 method. This implies that controlling the solution error in the ERK2 method has indeed ensured that the numerical solution to the concentration field is stable even when the ODEs are nonlinear. The need of choosing a stepsize apriori to ensure stability of the ERK2 scheme is thus obviated. In the partitioned scheme the concentration and the displacement fields are essentially solved independently at each time step. In spite of this, the results closely agree with those from

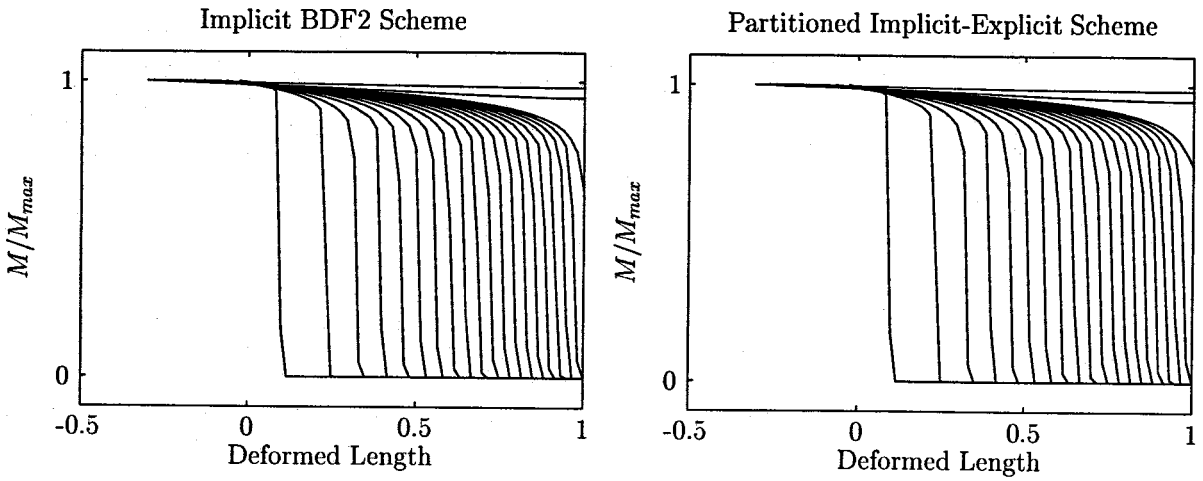


Figure 5: Concentration Profiles at every 0.015 units of normalized time θ starting from $\theta = 0.005$ at the left most front to $\theta = 0.325$ close to steady state.

the BDF2 scheme where the two fields are solved together. The possible reason for good accuracy here is because the stepsizes are small.

Effect of Stiffness due to Mesh Size

The effect of stiffness due to mesh size is examined by considering different spatial discretizations with 30, 60 and 90 elements. The results for the different meshes using a BDF2 scheme with Newton's method from $\theta = 0$ to $\theta = 0.025$ are tabulated in Table 7. The corresponding results for the ERK2-BDF2 scheme are summarized in Table 8. The cost increase for the BDF2 scheme in terms of function evaluations and Jacobian evaluations and inversions due to mesh refinement is not significant at all. On the other hand, the cost of function and Jacobian evaluations for the ERK2-BDF2 scheme increases more rapidly than $1/h^2$ to ensure stability as the mesh size h is refined. This makes the partitioned scheme ineffective even for practical mesh sizes considered here. Furthermore, it is interesting to note that the fraction of the total number of attempted steps failing convergence of the Newton's method in the BDF2 scheme increases from 5% for 30 elements to 17% for 90 elements. In other words, the stepsize requirements for satisfying the fixed error tolerance has less and less correlation with the stepsize requirements from the radius of convergence as the mesh is refined.

The cost statistics from using the Krylov Newton scheme with the BDF2 method for solving the nonlinear equations could not be obtained for this example. This is because the Krylov method experienced repeated failure of convergence of the iterates to the solution even when MAXTRY reductions in the stepsize were tried. However, convergence was not a

BDF2 Scheme With Newton's Method		ERK2-BDF2 with Newton's Method	
Total # of BDF2 Steps Attempted	360	Total # of Steps Attempted	67793
Total # of Successful Steps	315	Total # of Successful ERK Steps	67788
Total # of Function Evaluations	1787	Total # of ERK Function Evaluations	203380
Total # of Jacobian evaluations	1787	Steps Failing ERK Error Criteria	5
Steps Failing Error Criteria	0	Total # of Successful BDF2 Steps	67788
Steps Failing Convergence of Newton's Method	45	Function evaluations for BDF2	135182
		Jacobian evaluations for BDF2	135182
–	–	Steps Failing BDF2 Error Criteria	0
–	–	Steps Failing Newton's Method	0

Table 6: Statistics for Implicit BDF2 scheme and partitioned ERK2-BDF2 scheme applied to Model-1 with 60 elements for a full simulation from $\theta = 0$ to $\theta = 0.33$

problem when Krylov method was used to solve the nonlinear equations in the implicit part of the ERK2-BDF2 scheme.

Adaptive Time Stepping and the Stepsize-Error Tolerance Dilemma

The full stepsize sequence for the BDF2 scheme with 60 elements is plotted on a semi-log scale in Fig. 7. The method is started with a small initial stepsize of order 10^{-6} from which it proceeds by choosing stepsizes adaptively. The stepsizes at the end of the simulation are of order 10^{-3}. From the plot for the stepsize sequence, it is clear that adaptive time stepping continuously tries to maximize the stepsizes during the course of the simulation, at the same time satisfying the error criterion and the radius of convergence requirements.

From the compilation of statistics for this simulation in Table 6, around 12% of all the attempted steps failed. In all the cases, the failures were due to nonconvergence of the iterations while using the Newton's method. Therefore, choosing stepsizes larger than the stepsizes chosen here would be impossible as the Newton's Method would not converge. From the statistics, it is also interesting to note that none of the stepsizes failed due to violation of the error criterion. If crude tolerances are chosen, the stepsizes in some of the time steps where Newton's iterations converge may be large. But in that case, numerical experience showed that a larger percentage of the attempted steps would fail as the stepsize selection based on error criterion would have no correlation with the stepsize requirements based on the radius of convergence. This may lead to inefficiency because the adaptive scheme would have to try several stepsizes for which Newton's method would fail before finding the

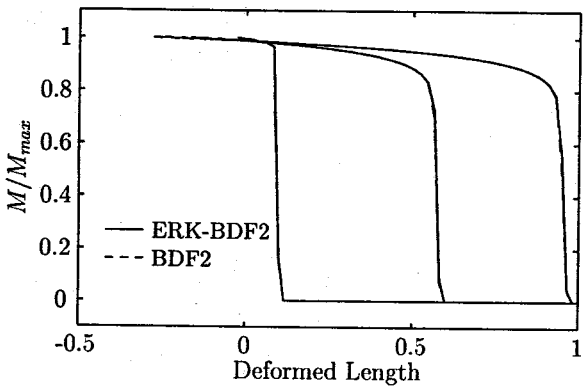


Figure 6: Comparison of partitioned ERK-BDF2 and the BDF2 scheme for Model-1; Looking from the left, the fronts correspond to $\theta = 0.005$, $\theta = 0.1$ and $\theta = 0.265$.

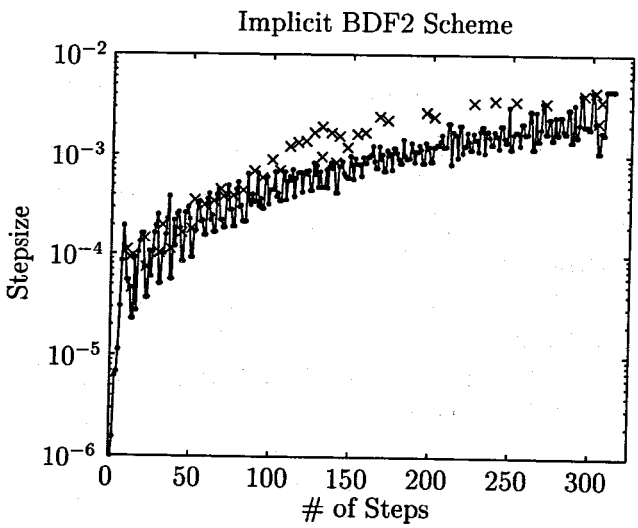


Figure 7: Stepsize sequence using BDF2 method for Model-1; Stepsizes are plotted in the interval [0, 0.342]. (×) denotes failed stepsizes.

	30 Elements	60 Elements	90 Elements
Total # of Steps Attempted	58	125	186
Total # of Successful Steps	55	119	153
Total # of Function Evaluations	272	618	908
Total # of Jacobian evaluations	272	618	908
Steps Failing Error Criteria	0	0	0
Steps Failing Convergence of Newton	3	16	33

Table 7: Statistics for BDF2 scheme using Newton's method applied to Model-1 for various mesh sizes from $\theta = 0$ to $\theta = 0.025$

right stepsize. Therefore, there is a delicate balance between achieving robustness in terms of convergence of Newton's method, accuracy in terms of satisfying an error criterion and efficiency in terms of being able to take the largest possible stepsizes.

For a coarse time step simulation with large stepsizes, where solution accuracy is not of concern, it is absolutely important to ensure that the Newton's method has a very good initial guess. In this case, choosing the converged solution at the previous step or a linear combination of converged solutions at the previous steps (as in the case of extrapolation) as the initial guess for the Newton's method is found to be insufficient. The effect of a bad initial guess is illustrated in Fig. 8. The data in the figure is taken from one of the actual simulations using Model-2. The situation considered here corresponds to a time step where the converged solution at the previous step is not a good initial guess. The Newton iterations diverge in this time step. In the figure, the norm of the residual $\|\mathbf{R}(\mathbf{Y}_k + s\Delta\mathbf{Y}_K)\|$ is plotted against $s \in [0, 1]$ where k is the iteration number (see Appendix B for notation). The residual norm increases in the direction $\Delta\mathbf{Y}_K$ in a very haphazard way. The consequence is divergence of the iterations. This is unlike the situation where the initial guess is within the radius of convergence and the residual norm typically decreases with s . The only solution to divergence of iterations is abandoning the current stepsize and choosing a smaller one.

The stepsize sequence for the ERK2-BDF2 scheme with 60 elements is shown in Fig. 9. A sequence of 250 attempted stepsizes in the time interval [0.328917705, 0.329993138], close to the steady state, and a sequence of 250 stepsizes in the interval [0.171909993, 0.173074479] near half-time of the simulation are plotted. The stepsizes are 3 orders of magnitude smaller than those in the BDF2 method. Due to stiffness, the stepsizes don't change even when the concentration solution does not change much while approaching the steady state.

The time integration of the stress equations is always done using the implicit BDF2 scheme. The influence of stiffness due to nonlinear relaxation times is not examined here as the BDF2 scheme was found to handle this source of stiffness effectively.

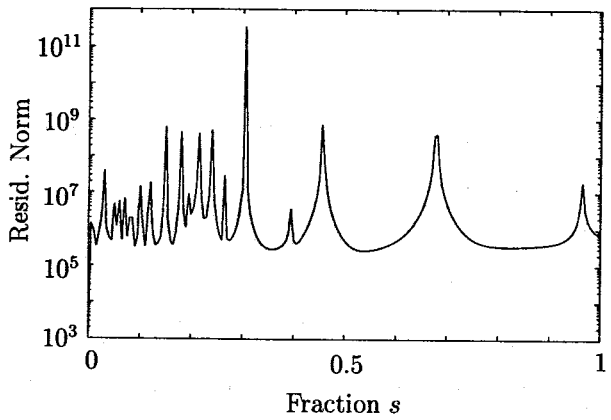


Figure 8: Variation of the Residual Norm for large $\Delta\theta$ along a direction provided by the Newton's Method; s is fraction of the Newton's step in an iteration with $s = 1$ being the full Newton Step.

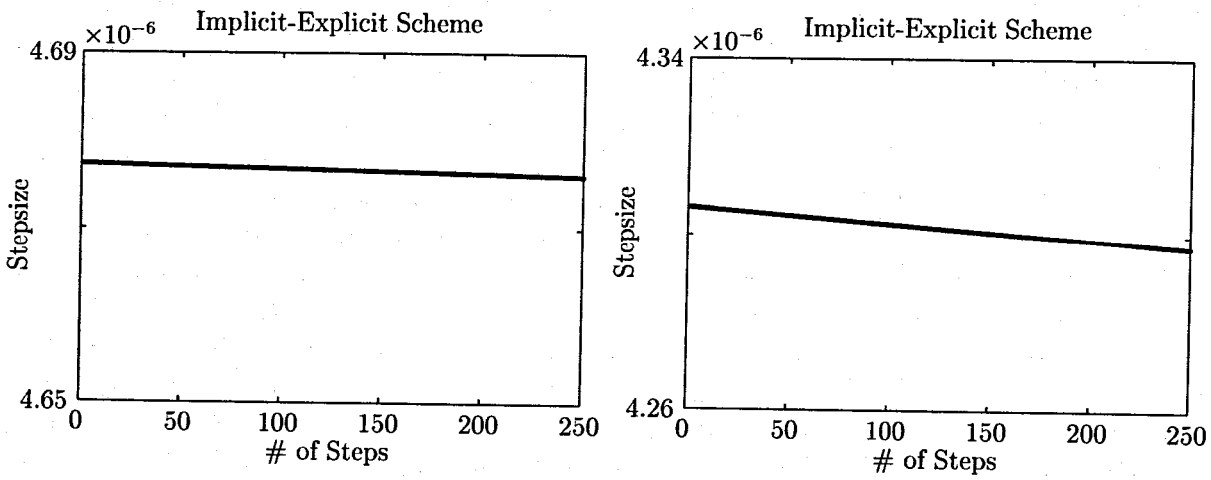


Figure 9: Stepsize sequence using ERK-BDF2 method for Model-1; Left: Stepsize in the time interval [0.171909993, 0.173074479]. Right: Stepsize in the time interval [0.328917705, 0.329993138]. (×) denotes failed stepsizes.

	30 Elements	60 Elements	90 Elements
Total # of Steps Attempted	769	4141	10284
Total # of Successful ERK Steps	764	4136	10279
Total # of ERK Function Evaluations	2308	12424	30853
Steps Failing ERK Error Criteria	5	5	5
Krylov Newton For the Implicit Part			
Total # of Successful BDF2 Steps	764	4136	10279
Function evaluations for BDF2	1657	8343	19104
Jacobian evaluations for BDF2	709	4082	10224
Steps Failing BDF2 Error Criteria	0	0	0
Failed Krylov (New Jacobian)	0	0	0
Failed Krylov (old Jacobian)	709	4082	10224
Standard Newton For the Implicit Part			
Total # of Successful BDF2 Steps	764	4136	10279
Function evaluations for BDF2	1601	8345	20629
Jacobian evaluations for BDF2	1601	8345	20629
Steps Failing BDF2 Error Criteria	0	0	0
Failed Newton	0	0	0

Table 8: Statistics for ERK2-BDF2 scheme applied to Model-1 for various mesh sizes from $\theta = 0$ to $\theta = 0.025$

Remark 6

In the simulations described in this section, the tolerances $atol_q^{(i)}$ and $rtol_q^{(i)}$ for stress at each quadrature point are not used to determine whether the current stepsize is accepted or rejected. These tolerances are also not used to predict a stepsize for the next step. The reason for this is as follows: The error criterion for the BDF2 scheme applied to the stress equations is based on the predictor error. When a sharp concentration front happens to pass a quadrature point in a time step, the viscoelastic stress $\bar{q}^{(i)}$ at the quadrature point would change rapidly in that time step. The resulting predictor error would also be large. The predictor error at the other quadrature points would still be small. Therefore, basing the decision of accepting or rejecting a stepsize on violation of the error criterion at a few quadrature points which the front might pass in each time step turns out to be very conservative. Due to a lack of a better criterion to decide on which quadrature points need to be picked for checking the error conditions, the error criteria for $\bar{q}$ at the quadrature points

Parameter	Value
a_d	20
a_η	50
χ	0.9
$\bar{E}$	30
$\bar{\tau}_0$	2.5

Table 9: Material Properties for Model-2

are completely ignored. The same discussion holds for doing away with the error criteria for $\bar{\sigma}$ in the case of Model-2.

4.2 Numerical Results for Model-2

The numerical simulation of Case II diffusion using Model-2 is considered here. The material properties chosen for this model are listed in Table 9. As in the case of Model-1, the domain of interest is the interval $[0, 1]$. The boundary conditions for the diffusion problem are: a fixed volume fraction $\nu\,|_{\bar{X}=0} = 0.3832$ at the left end, and a zero flux $\breve{f}\,|_{\bar{X}=1} = 0$ at the right end. Similar to the previous model, the maximum volume fraction ν_{max} occurs that the left end boundary with the value $\nu_{max} = 0.3832$. There are no boundary conditions for the mechanical behavior that are needed to be specified for the IBVP. Furthermore, zero volume fractions and zero stresses in the polymer are chosen as the initial conditions for the IBVP.

The results of time integration using the fully implicit BDF2 scheme and the partitioned ERK2-BDF2 scheme with 60 elements is shown in Fig. 10. The tolerances $atol_\nu = 2 \times 10^{-3}$ and $rtol_\nu = 2 \times 10^{-2}$ are chosen for adaptive time stepping in both the BDF2 and ERK2-BDF2 schemes. As before, the tolerances $atol_\sigma^{(i)}$ and $rtol_\sigma^{(i)}$ for integration of the stress at each quadrature point (Gauss point) are set to a high value so that the stepsizes don't depend on them. The values of the parameters MAXTRY, MAXITR, TOL and MAXFAIL are chosen to be the same as in the case of Model-1. Similarly, the various input parameters for Krylov Newton implementation are also chosen to be the same as before. In Fig. 10, volume fraction profiles are plotted on the deformed length of the polymer. The deformed length is calculated using the ideal mixing assumption, where the stretches λ computed from ν are integrated over the length to obtain the elongation. The right end of the polymer is assumed to be fixed at $\bar{X} = 1$ for carrying out the integration. The volume fraction profiles are plotted at every 0.3 units of time starting at $\theta = 0.1$ for the left most front. The entire simulation takes 5.5 units of time. Sharp fronts of the volume fraction are formed as in Model-1 and move from left to right. The diffusion problem finally reaches steady state

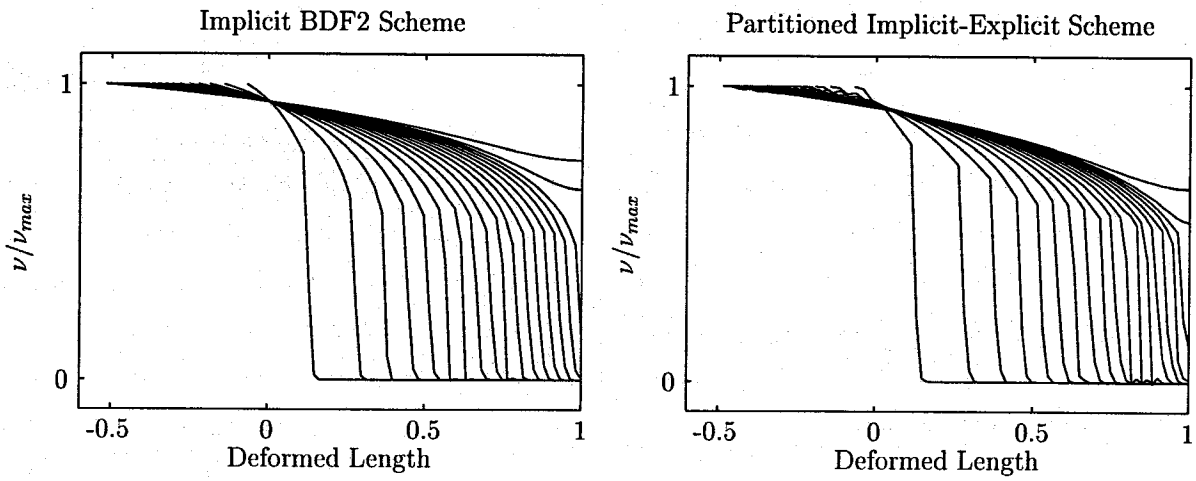


Figure 10: Concentration Profiles at every 0.1 units of normalized time θ starting from $\theta = 0.1$ for the left most front to $\theta = 6.5$ close to steady state.

where $\nu = \nu_{max}$ everywhere as governed by the zero flux boundary condition at the right end. At steady state the net elongation of the polymer due to swelling is about 60%.

The BDF2 and the ERK2-BDF2 schemes for the full simulation in Fig. 10 are compared in Table 10. Newton's method is used to solve the implicit nonlinear equations that arise using the BDF2 scheme. The implicit equation for the scalar variable stress at each quadrature point arising in the ERK2-BDF2 and the BDF2 scheme is solved in closed form. The BDF2 scheme took 1490 steps while the ERK2-BDF2 scheme took 13068 steps. Unlike in the case of Model-1, where the implicit BDF2 scheme performed far better than the partitioned scheme, the implicit BDF2 scheme performs marginally better than the ERK2-BDF2 scheme here. In fact, the cost of 6278 function evaluations and 6278 Jacobian evaluations and inversions for the implicit scheme may be comparable to 39577 function evaluations using the ERK2-BDF2 scheme. For large systems of equations, formation and the inversion of the Jacobian can be very expensive and this may outweigh the benefits of using an implicit scheme.

The accuracy of the partitioned scheme is assessed by plotting the normalized volume fraction profiles at various time values obtained from the BDF2 method and ERK2-BDF2 scheme Fig. 11. The profiles from the two methods don't quite match for the two schemes possibly because the stepsizes in the partitioned scheme are not small enough. Nevertheless, the results are still in good agreement.

Effect of Stiffness due to Mesh Size

The effect of stiffness due to mesh size is examined by considering different spatial discretizations with 30, 60 and 120 elements. The results for the different meshes using BDF2

BDF2 Scheme With Newton's Method		ERK2-BDF2 with Newton's Method	
Total # of BDF2 Steps Attempted	1530	Total # of Steps ERK Attempted	13189
Total # of Successful Steps	1490	Total # of Successful ERK Steps	13068
Total # of Function Evaluations	6278	Total # of ERK Function Evaluations	39577
Total # of Jacobian evaluations	6278	Steps Failing ERK Error Criteria	121
Steps Failing Error Criteria	40	Total # of Successful BDF2 Steps	–NA–
Steps Failing Convergence of Newton	1	Function evaluations for BDF2	–

Table 10: Statistics for Implicit BDF2 scheme and partitioned ERK2-BDF2 scheme applied to Model-2 with 60 elements for a full simulation from $\theta = 0$ to $\theta = 5.5$

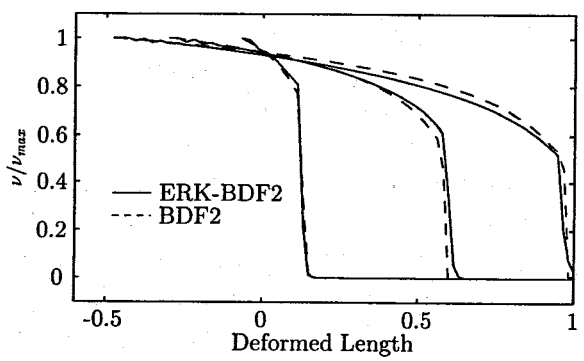


Figure 11: Comparison of partitioned ERK2-BDF2 and the BDF2 scheme for Model-2; Looking from the left, the fronts correspond to $\theta = 0.1$, $\theta = 1.7$ and $\theta = 4.7$.

	30 Elements	60 Elements	120 Elements
Total # of Steps Attempted	165	605	1913
Total # of Successful Steps	163	582	1842
Total # of Function Evaluations	655	2475	8092
Total # of Jacobian evaluations	655	2475	8092
Steps Failing Error Criteria	2	21	27
Steps Failing Convergence of Newton	0	1	44

Table 11: Statistics for BDF2 scheme using Newton's method applied to Model-2 for various mesh sizes from $\theta = 0$ to $\theta = 0.5$

	30 Elements	60 Elements	120 Elements
Total # of Steps Attempted	284	1026	3377
Total # of Successful Steps	165	577	1977
Total # of Function Evaluations	1539	5686	19158
Total # of Jacobian evaluations	119	426	1274
Steps Failing Error Criteria	1	3	4
Failing Krylov (old Jacobian)	117	424	1272
Failing Krylov (new Jacobian)	1	22	124

Table 12: Statistics for BDF2 scheme using Krylov Newton's method applied to Model-2 for various mesh sizes from $\theta = 0$ to $\theta = 0.5$

scheme with Newton's method from $\theta = 0$ to $\theta = 0.5$ are tabulated in Table 11 while those from using the Krylov Newton method are listed in Table 12. Furthermore, the results for the ERK2-BDF2 scheme are summarized in Table 13. The important observations from these tables are as follows: stiffness due to the mesh size h increases the cost of the ERK2-BDF2 scheme more rapidly than $1/h^2$ as the mesh size h is refined. The mesh size has a significant influence even on the implicit BDF2 scheme where the number of function evaluations increase 4 times every time the mesh size is halved. The number of stepssizes failing due to violation of error criterion is comparable to the number of stepsizes due to nonconvergence of the iterations when Newton's method is used with the BDF2 scheme. On the other hand, most of the failed stepsizes are due to nonconvergence of the iterations when Krylov Newton method is used with the BDF2 scheme. In other words, Krylov Newton method has a smaller radius of convergence. However, the Krylov Newton requires significantly fewer number of

	30 Elements	60 Elements	120 Elements
Total # of ERK Steps Attempted	226	1077	5180
Total # of Successful ERK Steps	199	968	4714
Total # of ERK Function Evaluations	760	3568	17008
Steps Failing ERK Error Criteria	27	109	466

Table 13: Statistics for ERK2-BDF2 scheme applied to Model-2 for various mesh sizes from $\theta = 0$ to $\theta = 0.5$

Jacobian evaluations compared to the Newton scheme resulting in enormous savings in the computational cost. In conclusion, for the cases considered here where an error tolerance is specified, BDF2 scheme with Krylov Newton ranks the best followed by BDF2 with full newton and ERK2-BDF2 scheme whose performances are comparable.

Adaptive Time Stepping

The adaptive time stepping algorithm for the BDF2 scheme with Newton's method is illustrated by plotting stepsize sequences in the time intervals [0.8688, 1.6979] and [3.0973, 5.4993] in Fig. 12. Similarly, the stepsize sequences for the ERK2-BDF2 scheme in the time intervals [2.7911, 2.8943] and [5.3976, 5.4993] are plotted in Fig. 13. In the BDF2 scheme, the stepsizes at the end of the simulation are about an order of magnitude larger compared to their values in the interval [0.8688, 1.6979]. This is to be expected as the solution to the volume fraction field changes slowly in most part of the domain while close to steady state. On the other hand, the ERK2-BDF2 scheme suffers from stability requirements due to stiffness and the stepsizes remain the same as at half-time even when the steady state is approached.

Handling Oscillations due to Sharp Fronts

The near vertical solvent concentration profiles in the sharp fronts lead to undershoots in the numerical solution due to oscillations. These oscillations are illustrated in Fig. 14 for Model-2 where a finite element discretization with 60 elements is used. The undershoots that result in negative concentrations are not only unphysical but also detrimental to carrying out the time integration. From numerical experience, negative concentrations resulted in repeated stepsize failures when solving nonlinear equations in implicit schemes thereby forcing the integration scheme to stop. On taking a closer look at the various quantities such as the flux that depend nonlinearly on the concentration, it was found that they were totally wrong when evaluated for negative concentrations – sometimes even with a wrong sign. The effect of undershoots on the explicit schemes was not very significant and the explicit schemes recovered without the integration scheme being forced to stop.

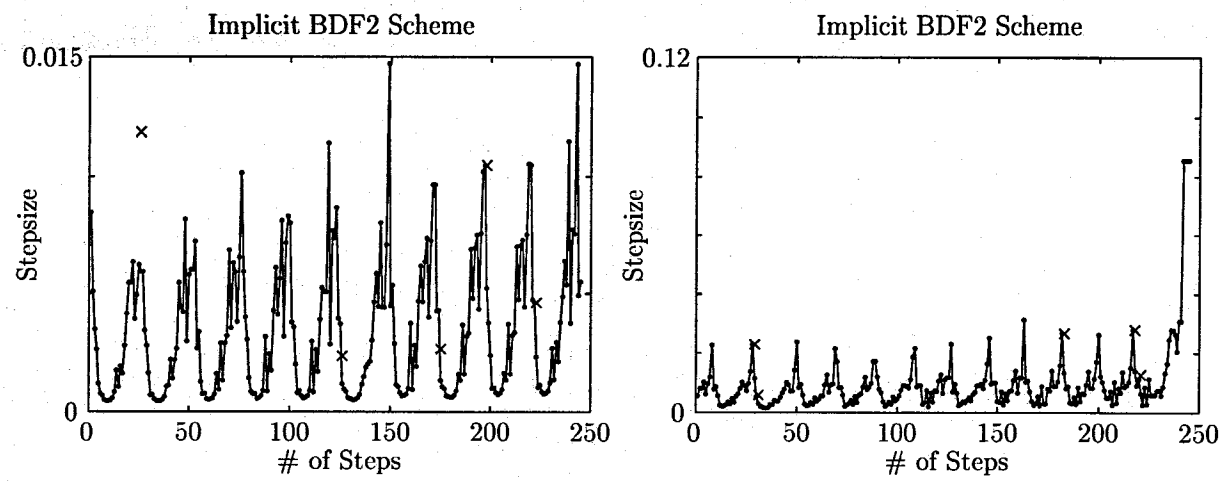


Figure 12: Stepsize sequence using BDF2 scheme with Newton's method for Model-2; Left: Stepsize in the interval [0.8688, 1.6979]. Right: Stepsize in the interval [3.0973, 5.4993]. (×) denotes failed stepsizes.

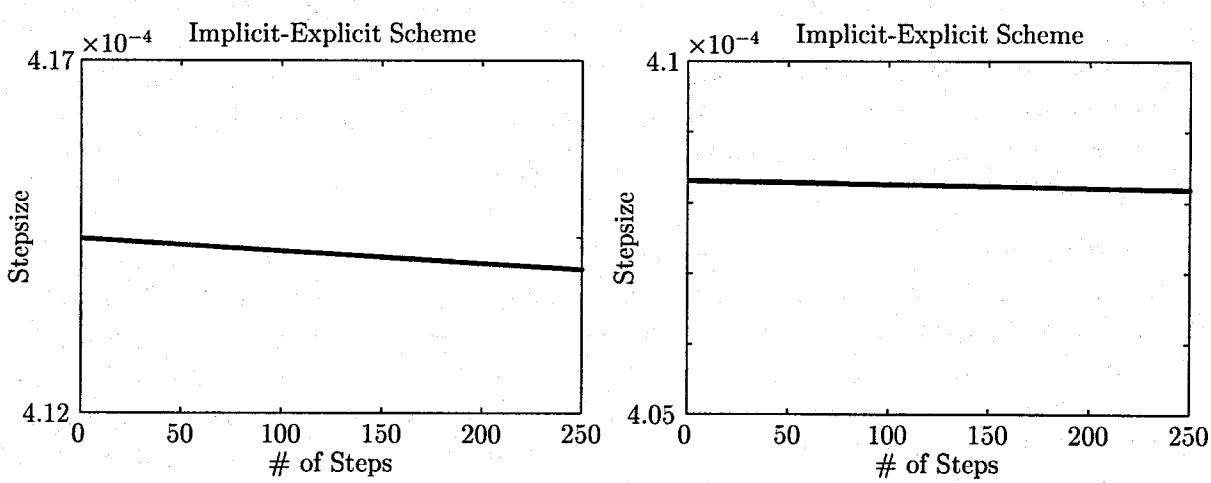


Figure 13: Stepsize sequence using ERK-BDF2 method for Model-2; Left: Stepsize in the interval [2.7911, 2.8943]. Right: Stepsize in the interval [5.3976, 5.4993]. (×) denotes failed stepsizes.

The undershoots here are handled by implementing the projection scheme described in Appendix B. The results for Model-2 in Fig. 10 using 60 elements are obtained by using the projection technique for both the BDF2 and the ERK2-BDF2 schemes and can be compared with Fig. 14.

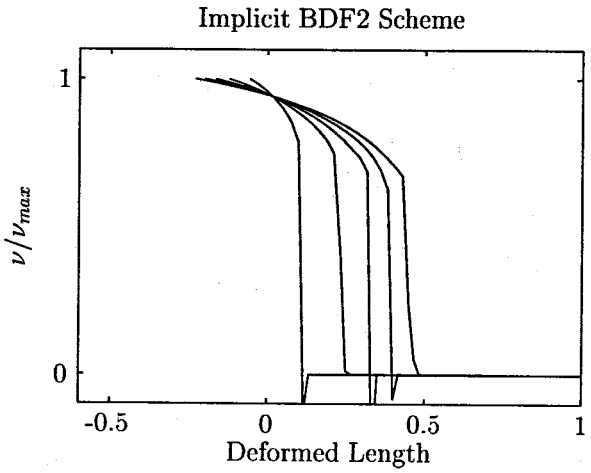


Figure 14: Oscillations in Coarse Meshes due to Sharp Fronts.

Surprisingly, in the case of Model-1, no undershoots were detected even for coarse meshes. In addition to the projection scheme used here, lumped mass matrices were used in all the simulations for both the models. The main motivation for using lumped mass matrices in the numerical simulation of the two models is that they are shown to satisfy the maximum and minimum principles (see THOMEE [1980]), at least in the uncoupled case.

Finally, a remark on the error tolerances is made here.

Remark 7

Error tolerances, $atol_\nu$ and $rtol_\nu$ specified here ensure stability for the explicit part of the partitioned scheme. If larger tolerances are used, stepsizes would be larger and the numerical solution would still grow in time in a controlled manner without blowing up due to instability. However, oscillations start appearing in the spatial solution as a manifestation of instability. In fact, one can see mild oscillations in the spatial solution to the volume fraction in Fig. 10 close to the left end. Therefore, in the explicit schemes oscillations could also occur if crude error tolerances for ensuring stability are used.

5 Summary and Conclusions

In this report, two models for simulating Case II diffusion in 1-D were presented. These simple models in 1-D facilitated a good understanding of what features models for Case II diffusion should have. Most notably, the numerical simulations illustrated that nonlinear mobility dependent on the solvent concentration and coupling of the chemical potential with deformations or stress-like quantities are crucial for the single most important feature of Case II diffusion – formation and propagation of sharp solvent fronts. The two models were presented in a general framework involving balance laws and constitutive relations. This framework has helped in seeing clearly the differences between fully coupled models with independent displacement and concentration fields and those where simplifying assumptions such as ideal mixing on the mechanical behavior are made. All the balance laws and constitutive relations were presented in a dimensionless form. This provides the facility to simulate Case II diffusion in various solvent-polymer combinations by simply scaling the dimensionless time and the dimensionless length. The normalization of the governing equations also avoided numerical complications associated with handling different orders of magnitudes of the solutions to the different fields due to a bad choice of units.

These models were successfully implemented in a numerical setting. As a first step towards this, a mathematical understanding of the qualitative behavior of the solutions was presented. The relevant concepts included the maximum and minimum principles and stiffness in the DAEs obtained from spatial discretization using the finite element method. Although implicit schemes were the recommended choice for handling stiffness, an alternative partioned implicit-explicit scheme was explored to see how it compared with the implicit schemes. Similarly, Krylov Newton method which is a cheaper alternative to Newton's method for solving nonlinear equations was examined. A motivation was provided for using adaptive time stepping for time integration of the spatially discrete governing equations. Through carefully chosen numerical examples, the efficacy of using adaptive time stepping to strike a balance between efficiency, accuracy and robustness requirements was illustrated. Although adaptive time stepping was found to be very useful, a genuine need for good initial guesses for the iterative schemes (such as Newton) for solving nonlinear equations when using coarse time steps with implicit methods was recognized.

An assessment of the various numerical techniques can be made as follows: To varying degrees, implicit schemes performed better than the partitioned implicit-explicit schemes. In the case of Model-1, the implicit BDF2 scheme was clearly more efficient than the partitioned ERK2-BDF2 scheme. Stiffness played a significant role in forcing small stepsizes on the partitioned scheme while the stepsize restrictions on the implicit scheme due to radius of convergence was not significant. Krylov Newton method exhibited very small radius of convergence and could not be used in the implicit BDF2 method at all. In the case of Model-2, the implicit BDF2 scheme with Krylov Newton method was found to be more efficient than the implicit BDF2 scheme with Newton's method or the ERK2-BDF2 method. In this case, the cost of the ERK2-BDF2 method was also comparable to the implicit BDF2 scheme with

Newton's method. Furthermore, a technique for handling unphysical undershoots that lead to negative concentrations was described. It was found that convergence of iterations while solving nonlinear equations in the implicit BDF2 scheme crucially depended on avoiding the undershoots altogether.

In conclusion, a sound understanding of various aspects of modeling Case II behavior and its numerical implementation was accomplished by considering the 1-D case. With this background, it is hoped to understand the extensions to the 2-D and the 3-D cases better.

APPENDIX A: NEWTON AND KRYLOV NEWTON METHODS

In this Appendix, a Krylov Newton method developed in CARLSON AND MILLER [1998a] for the solution of nonlinear algebraic equations is described. This is followed by a short comparison of the Krylov method with the standard Newton's Method.

Consider a system of nonlinear equations of the form

$$\mathbf{R}(\mathbf{Y}) = 0 \tag{A-1}$$

As a first step, these equations are scaled through a premultiplication by $-J^{-1}$, where $J = \mathbf{R}'(\bar{\mathbf{Y}})$ denotes the Jacobian of $\mathbf{R}$ evaluated at some value $\bar{\mathbf{Y}}$. Consequently, one obtains

$$\mathbf{f}(\mathbf{Y}) = -J^{-1}\mathbf{R}(\mathbf{Y}) = 0 \tag{A-2}$$

It is easily seen that the zero of $\mathbf{f}$ is also a zero of $\mathbf{R}$ as long as J is invertible. Scaling by the inverse of the jacobian normalizes the equations. In fact, the jacobian $\mathbf{f}'(\bar{\mathbf{Y}})$ of the scaled equations is approximately the identity $-\mathbf{I}$ near the root $\mathbf{Y}$ if the jacobian J has been freshly updated. The nonlinear Krylov Newton method is used to solve the equations in (A-2) and assumes the following

1. $\mathbf{f}(\mathbf{X} + \mathbf{Z}) = \mathbf{f}(\mathbf{X}) - \mathbf{AZ}$ for all $\mathbf{X}$ near the root $\mathbf{Y}$ where $\mathbf{A}$ is an unknown constant nonsingular matrix.

2. $\mathbf{AZ} \approx \mathbf{IZ}$ for small corrections $\mathbf{Z}$.

Starting with a good initial guess $\mathbf{Y}_0$, the goal here is to find a sequence of iterates $\mathbf{Y}_k$ that converge to the solution $\mathbf{Y}$, cheaply. This is done by building a subspace $\mathcal{V}$ of much smaller dimension than $\mathbf{Y}$ and looking for a solution in the subspace.

The method starts by evaluating $\mathbf{f}$ at $\mathbf{Y}_0$ to obtain $\mathbf{f}_0$. The first member $\mathbf{v}_1$ of the subspace $\mathcal{V}$ is set to $\mathbf{f}_0$. The next iterate $\mathbf{Y}_1$ is obtained by setting $\mathbf{Y}_1 = \mathbf{Y}_0 + \mathbf{v}_1$. From assumption 1, this implies that

$$\mathbf{Av}_1 = \mathbf{f}_0 - \mathbf{f}_1 \tag{A-3}$$

where $\mathbf{f}_1 = \mathbf{f}(\mathbf{Y}_1)$. Following this, the iterate $\mathbf{Y}_2$ is obtained in terms of the residual correction equation

$$\mathbf{f}(\mathbf{Y}_2) = \mathbf{f}(\mathbf{Y}_1 + \mathbf{v}_2) = \mathbf{f}_1 - \mathbf{Av}_2 = 0 \tag{A-4}$$

In the first pass, the above residual equation is approximately solved by letting $\mathbf{w}_2$ be that element of the subspace $\mathcal{V} = \text{span}\{\mathbf{v}_1\}$, that best approximates $\mathbf{f}_1$ in the L_2 norm. This is a simple linear least squares problem and in the case where the subspace $\mathcal{V}$ has one vector, $\mathbf{w}_2$ is simply given by

$$\mathbf{w}_2 = [(\mathbf{Av}_1)^T(\mathbf{Av}_1)]^{-1}\mathbf{f}_1 \tag{A-5}$$

From $\mathbf{w}_2$, an approximation $\mathbf{Y}_2^* = \mathbf{Y}_1 + \mathbf{w}_2$ to $\mathbf{Y}_2$ is made which would have a residual

$$\mathbf{q}_2 = \mathbf{f}(\mathbf{Y}_2^*) = \mathbf{f}(\mathbf{Y}_1 + \mathbf{w}_2) = \mathbf{f}_1 - \mathbf{Aw}_2 \tag{A-6}$$

A further correction $\mathbf{z}$ to $\mathbf{Y}_2^*$ is made by satisfying

$$\mathbf{f}(\mathbf{Y}_2^* + \mathbf{z}) = \mathbf{q}_2 - \mathbf{A}\mathbf{z} = 0 \tag{A-7}$$

Now, using assumption 2 which says $\mathbf{A}\mathbf{z} \approx \mathbf{I}\mathbf{z}$, one obtains

$$\mathbf{A}\mathbf{z} = \mathbf{q}_2 \tag{A-8}$$

Finally, the second member $\mathbf{v}_2$ of the collection of vectors that span the subspace $\mathcal{V}$ is calculated as

$$\mathbf{v}_2 = \mathbf{w}_2 + \mathbf{q}_2 \tag{A-9}$$

Knowing $\mathbf{v}_2$, $\mathbf{A}\mathbf{v}_2$ is obtained by the difference

$$\mathbf{A}\mathbf{v}_2 = \mathbf{f}_1 - \mathbf{f}_2 \tag{A-10}$$

where $\mathbf{f}_2 = \mathbf{f}(\mathbf{Y}_2) = \mathbf{f}(\mathbf{Y}_1 + \mathbf{v}_2)$. This constitutes the end of two steps where the vectors $\{\mathbf{v}_1, \mathbf{v}_2\}$ and their $\mathbf{A}$ values $\{\mathbf{A}\mathbf{v}_1, \mathbf{A}\mathbf{v}_2\}$ have been accumulated. In a similar way, at the end of k steps, the vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ that span the subspace $\mathcal{V}$ and their $\mathbf{A}$ values $\{\mathbf{A}\mathbf{v}_1, \ldots, \mathbf{A}\mathbf{v}_k\}$ would have been accumulated. The desired correction $\mathbf{v}_{k+1}$ is obtained by solving the residual correction equation

$$\mathbf{f}(\mathbf{Y}_{k+1}) = \mathbf{f}(\mathbf{Y}_k + \mathbf{v}_{k+1}) = \mathbf{f}_k - \mathbf{A}\mathbf{v}_{k+1} = 0 \tag{A-11}$$

As before, $\mathbf{v}_{k+1}$ is approximated by $\mathbf{w}_{k+1}$, where $\mathbf{w}_{k+1}$ is the element in the span of $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$, that best approximates $\mathbf{f}_k$. The linear least squares problem in this case is

$$\mathbf{w}_{k+1} = [\mathbf{K}]^{-1}\mathbf{f}_k \tag{A-12}$$

where $\mathbf{K}$ is a symmetric matrix whose elements are inner products of the form $(\mathbf{A}\mathbf{v}_i)^T(\mathbf{A}\mathbf{v}_i)$. As previously done, a further correction to $\mathbf{w}_{k+1}$ is made finding the residual

$$\mathbf{q}_{k+1} = \mathbf{f}(\mathbf{Y}_k + \mathbf{w}_{k+1}) = \mathbf{f}_k - \mathbf{A}\mathbf{w}_{k+1} \tag{A-13}$$

Using assumption 2 as before, yields

$$\mathbf{A}\mathbf{z} = \mathbf{q}_{k+1} \tag{A-14}$$

and the correction $\mathbf{v}_{k+1}$ is given by

$$\mathbf{v}_{k+1} = \mathbf{w}_{k+1} + \mathbf{q}_{k+1} \tag{A-15}$$

Finally, the $\mathbf{A}$ value of $\mathbf{v}_{k+1}$ is computed using

$$\mathbf{A}\mathbf{v}_{k+1} = \mathbf{f}_k - \mathbf{f}_{k+1} \tag{A-16}$$

It is important to remark that $\mathbf{A}$ is never explicitly constructed. In all the steps, $\mathbf{Av}_i$ are directly computed.

As a word of caution, when $\mathbf{f}$ is strongly nonlinear (as in our case), some of the vectors $\mathbf{Av}_i$s may become nearly linearly dependent and result in the ill-conditioning of the matrix $\mathbf{K}$. In that situation, the oldest vectors $\mathbf{v}_i$ and their $\mathbf{A}$ values $\mathbf{Av}_i$ are repeatedly discarded to form a smaller collection of the $\mathbf{v}_k$s and $\mathbf{Av}_k$s until the matrix $\mathbf{K}$ is well conditioned again. In any case, the dimension of the subspace $\mathcal{V}$ is typically restricted to a small number (say around 10) even for very large sized problems.

The Krylov Newton method described above derives its name from the fact that the subspace $\mathcal{V} = \text{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ coincides with the Krylov subspace $\mathcal{M} = \text{span}\{\mathbf{f}_0, \mathbf{Af}_0, \ldots, \mathbf{A}^{k-1}\mathbf{f}_0\}$ when $\mathbf{A}$ is positive definite. The procedure involves simple function evaluations except for a jacobian evaluation to scale the residual equations. This is in contrast to the standard Newton scheme where the iterates $\mathbf{Y}_k$ are calculated through the following steps.

$$\Delta\mathbf{Y}_k = -(\mathbf{R}'(\mathbf{Y}_k))^{-1}\mathbf{R}(\mathbf{Y}_k) \tag{A-17}$$

$$\mathbf{Y}_{k+1} = \mathbf{Y}_k + \Delta\mathbf{Y}_k \tag{A-18}$$

Here, $\mathbf{R}'(\mathbf{Y}_k)$ is the Jacobian matrix that needs to be evaluated and inverted at each iteration k in addition to a function evaluation of $\mathbf{R}(\mathbf{Y}_k)$. On the other hand, the Krylov Newton method requires the jacobian evaluation and inversion only once followed by several function evaluations to find the iterates $\mathbf{Y}_k$. However, the possible disadvantage with the Krylov method is that the radius of convergence for the iteration scheme may be considerably small compared to the Newton scheme where the jacobian is updated at every iteration.

APPENDIX B: A PROJECTION METHOD FOR PREVENTING OSCILLATIONS

A projection method for preventing overshoots and undershoots of the numerical solution to the concentration field due to oscillations is described in this appendix. Several techniques for obtaining monotone solutions, without oscillations, that satisfy the maximum and the minimum principles are developed for advection diffusion equations (see for *e.g.*, MIZUKAMI AND HUGHES [1985], ROY AND BAKER [1997], SHEU ET AL. [1997]). However, none of the techniques were found to be applicable in our case. On the other hand, the work of LAYTON AND POLMAN [1996] provides a projection technique which is applicable to any problem with overshoots and undershoots in the numerical solution though it was developed in the context of advection-diffusion equations. This technique is described here.

We begin by assuming that the discrete solution $\mathbf{w}^h$ is required to satisfy

$$w_{max} \geq \mathbf{w}^h \geq w_{min} \tag{B-1}$$

where w_{min} and w_{max} are the lower and upper bounds for $\mathbf{w}^h$. Denote $\Pi\mathbf{w}^h$ as the unique element in the space $\mathbb{K}^h$ satisfying

$$\|\mathbf{w}^h - \Pi\mathbf{w}^h\|_2 = \inf_{\chi \in \mathbb{K}^h} \|\mathbf{w}^h - \chi\|_2 \tag{B-2}$$

where $\|\cdot\|_2$ denotes the standard L_2 norm, and,

$$\mathbb{K}^h = \{\mathbf{w}^h \in \mathbf{X}^h \mid w_{max} \geq \mathbf{w}^h \geq w_{min}\} \tag{B-3}$$

The projection $\Pi\mathbf{w}^h$ is shown to satisfy (see LAYTON AND POLMAN [1996]) the error estimate

$$\|\mathbf{w} - \Pi\mathbf{w}^h\|_2 \leq \|\mathbf{w} - \mathbf{w}^h\|_2 + \inf_{\chi \in \mathbb{K}^h} \|\mathbf{w} - \chi\|_2 \tag{B-4}$$

where $\mathbf{w}$ is the exact solution that is monotone and satisfies the condition in equation (B-1). This error estimate is independent of the problem at hand.

In the context of the Finite Element method where the discrete solutions $\mathbf{w}^h$ are given in terms of interpolation of nodal values, the constraints in (B-1) simply translate to constraints of the form

$$w_{max} \geq w^A \geq w_{min} \tag{B-5}$$

for each of the nodal values w^A. The discrete spatial solution $\mathbf{w}^h$ obtained by the finite element method can be expressed as $\mathbf{w}^h = \sum_{A=1}^{N} \phi^A w^A$. Here, ϕ^A denotes a global shape function such that its restriction to an element containing node A is the same as isoparametric shape function associated with the node in the element. Defining,

$$c_{AB} = \int_l \phi^A \phi^B dX \tag{B-6}$$

the iterates w_i^A are given by

$$w_{i+\frac{1}{2}}^A = -\frac{1}{c_{AA}} \sum_{B=1}^{A-1} c_{AB} w_{i+\frac{1}{2}}^B + \sum_{B=A+1}^{N} c_{AB} w_i^B - f_A \tag{B-7}$$

$$w_{i+1}^A = P_N[(1-\zeta) w_i^A + \zeta w_{i+\frac{1}{2}}^A] \tag{B-8}$$

Here, the quantity $f_A = \sum_{B=1}^{N} c_{AB} w^B$ where w^B is the unprojected nodal value of the finite element solution. The projection operator $P_N : \mathbb{R}^N \to \prod_1^N [w_{min}, w_{max}]$, where $\prod$ denotes the cartesian product, is defined by

$$(P_N \mathbf{x})_A = \begin{cases} x_A & \text{if } w_{max} \geq x^A \geq w_{min} \\ w_{max} & \text{if } x_A \geq w_{max} \\ w_{min} & \text{if } x_A \leq w_{min} \end{cases} \tag{B-9}$$

The iterates w_{i+1}^A converge usually in $\mathcal{O}(1)$ iterations to the projected nodal values. In the application considered here, the projection scheme is invoked at the end of each time step. The oscillation free solutions form the initial values for the solution at the next time step.

REFERENCES

Argon, A. S., R. E. Cohen, and A. C. Patel [1999]. A mechanistic model of Case II diffusion of a diluent into a glassy polymer. *Polymer 40*, 6991–7012.

Brenan, K. E., S. L. Campbell, and L. R. Petzold [1989]. *Numerical solution of initial-value problems in differential-algebraic equations.* Elsevier Science Pub. Co., North-Holland, New York.

Carbonell, R. G. and G. C. Sarti [1990]. Coupled deformation and mass-transport processes in solid polymers. *Industrial and Engineering Chemistry Research 29*(7), 1194–1204.

Carlson, N. N. and K. Miller [1998a]. Design and application of a gradient-weighted moving finite element code I: In one dimension. *SIAM Journal of Scientific Computing 19*(3), 728–765.

Carlson, N. N. and K. Miller [1998b]. Design and application of a gradient-weighted moving finite element code II: In two dimensions. *SIAM Journal of Scientific Computing 19*(3), 766–798.

Carranza, F. L., B. Fang, and R. B. Haber [1998]. An adaptive space-time finite element model for oxidation-driven fracture. *Computer Methods in Applied Mechanics and Engineering 157*, 399–423.

Cohen, D. S. and A. B. White [1991]. Sharp fronts due to diffusion and viscoelastic relaxation in polymers. *SIAM Journal of Applied Mathematics 51*(2), 472–483.

Croffie, E. [1999]. Moving boundary models and methods for deep submicron resist process simulation, Report No. UCB/ERL M99/26. Technical report, Electronics Research Laboratory, College of Engineering, U.C. Berkeley.

Cussler, E. L. [1984]. *Diffusion, mass transfer in fluid systems.* Cambridge University Press, New York.

Govindjee, S. [1991, June]. *Physical and Numerical Modelling in Filled Elastomeric Systems.* Ph. D. thesis, Stanford University.

Hairer, E., S. P. Norsett, and G. Wanner [1993]. *Solving ordinary differential equations I: Nonstiff problems.* Springer-Verlag, New York.

Hairer, E. and G. Wanner [1993]. *Solving ordinary differential equations II: Stiff and Differential-Algebraic problems.* Springer-Verlag, New York.

Hildebrand, J. H. [1947]. The entropy of solution of molecules of different size. *Journal of Chemical Physics 15*(5), 225–228.

Hughes, T. J. R. [2000]. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis.* Dover Publications, Inc., New York.

John, F. [1982]. *Partial Differential Equations.* Springer-Verlag, New York.

Lambert, J. D. [1993]. *Numerical Methods for Ordinary Differential Equations: The initial value problem.* John Wiley and Sons, New York.

Langtangen, H. P. [1999]. *Computational Partial Differential Equations: Numerical Methods and Diffpack Programming.* Springer-Verlag, Berlin.

Layton, W. and B. Polman [1996]. Oscillation absorption finite element methods for convection-diffusion problems. *SIAM Journal of Scientific Computing 17*(6), 1328–1346.

Miller, K. [1997]. Class notes for MATH 228A, U.C. Berkeley. Unpublished.

Mizukami, A. and T. J. R. Hughes [1985]. A Petrov-Galerkin finite-element method for convection-dominated flows an accurate upwinding technique for satisfying the maximum principle. *Computer Methods in Applied Mechanics and Engineering 50*(2), 181–193.

Ogden, R. W. [1997]. *Nonlinear Elastic Deformations.* Dover Publication, Inc., New York.

Rheinboldt, W. C. [1981]. Numerical analysis of continuation methods for nonlinear structural problems. *Computers and structures 13*, 103–113.

Roy, S. and A. J. Baker [1997]. Nonlinear, subgrid embedded finite-element basis for accurate, monotone, steady CFD solutions. *Numerical Heat Transfer, Part B 31*, 135–175.

Shampine, L. F. [1994]. *Numerical Solution of Ordinary Differential Equations.* Chapman and Hall, New York.

Shampine, L. F. and M. W. Reichelt [1997]. The MATLAB ODE suite. *SIAM Journal of Scientific Computing 18*(1), 1–22.

Sheu, T. W. H., S. F. Tsai, and M. M. T. Wang [1997]. A monotone finite element method with test space of Legendre polynomials. *Computer Methods in Applied Mechanics and Engineering 143*, 349–372.

Soderlind, G. [1980]. DASP3–A program for the numerical integration of partitioned stiff ODEs and Differential-Algebraic systems, TRITA-NA-8008. Technical report, The Royal Institute of Technology, Stockholm, Sweden.

Thomas, N. and A. H. Windle [1978]. Case-II swelling of PMMA sheet in methanol. *Journal of Membrane Science 3*(2–4), 337–342.

Thomee, V. [1980]. *Galerkin Finite Elements for Parabolic Problems.* Lecture Notes in Mathematics. Springer-Verlag, Berlin.

Vrentas, J. S. and C. M. Vrentas [2001]. Viscoelastic diffusion. *Journal of Polymer Science: Part B: Polymer Physics 39*, 1529–1547.

Weitsman, Y. [1987]. Stress assisted diffusion in elastic and viscoelastic materials. *Journal of Mechanics and Physics of Solids 35*(1), 73–93.

Wu, J. C. and N. A. Peppas [1993a]. Modeling of penetrant diffusion in glassy polymers with an integral sorption Deborah number. *Journal of Polymer Science, Part B (Polymer Physics) 31*, 1503–1518.

Wu, J. C. and N. A. Peppas [1993b]. Numerical simulation of anomalous penetrant diffusion in polymers. *Journal of Applied Polymer Science 49*, 1845–1856.