A monolithic multi-time-step computational framework for advective-diffusive-reactive transient systems with disparate scales

S. Karimi
Graduate Student, Department of Civil and Environmental Engineering,
University of Houston, Houston, Texas 77204–4003.

K. B. Nakshatrala
Department of Civil and Environmental Engineering
University of Houston, Houston, Texas 77204–4003.
Correspondence to: Professor Kalyana Babu Nakshatrala
e-mail: knakshatrala@uh.edu, phone:+1-713-743-4418
website: http://www.cive.uh.edu/faculty/nakshatrala

ABSTRACT. Developing robust simulation tools for problems involving multiple mathematical scales has been a subject of great interest in computational mathematics and engineering. A desirable feature to have in a numerical formulation for multiscale transient problems is to be able to employ different time-steps (multi-time-step coupling), and different time integrators and different numerical formulations (mixed methods) in different regions of the computational domain. To this end, we present two new monolithic multi-time-step mixed coupling methods for first-order transient systems. We shall employ unsteady advection-diffusion-reaction equation with linear decay as the model problem, which offers several unique challenges in terms of non-self-adjoint spatial operator and rich features in the solutions. We shall employ the dual Schur domain decomposition technique to handle the decomposition of domain into subdomains. It will be shown that the governing equations of the decomposed problem after spatial discretization will be differential/algebraic equations, which is a crucial observation to obtain stable numerical results. Two different methods of enforcing compatibility along the subdomain interface will be used in the time discrete setting. A systematic theoretical analysis (which includes numerical stability, influence of perturbations, bounds on drift along the subdomain interface) will be performed. The first coupling method ensures that there is no drift along the subdomain interface but does not facilitate explicit/implicit coupling. The second coupling method allows explicit/implicit coupling with controlled (but non-zero) drift in the solution along the subdomain interface. Several canonical problems will be solved to numerically verify the theoretical predictions, and to illustrate the overall performance of the proposed coupling methods. Finally, we shall illustrate the robustness of the proposed coupling methods using a multi-time-step transient simulation of a fast bimolecular advective-diffusive-reactive system.

Key words and phrases. multi-time-step schemes; monolithic coupling algorithms; advective-diffusive-reactive systems; partitioned schemes; differential-algebraic equations; Baumgarte stabilization.
1. INTRODUCTION AND MOTIVATION

Advection-diffusion-reaction equations can exhibit several mathematical (i.e., temporal and spatial) scales depending on the relative strengths of advection, diffusion and reaction processes, and on the strength of the volumetric source/sink. The presence of these mathematical scales is evident from the qualitative richness that the solutions of advection-diffusion-reaction equations exhibit. For example, it is well-known that solutions to advection-dominated problems typically exhibit steep gradients near the boundaries [Gresho and Sani, 2000]. Solutions to diffusion-dominated problems tend to be diffusive and smooth [McOwen, 1996], whereas reaction-dominated solutions typically exhibit sharp fronts and complex spatial patterns [Walgraef, 1997]. These scales can be systematically characterized using the well-known non-dimensional numbers – the Péclet number and the Damköhler numbers [Bird et al., 2006]. It needs to be emphasized that these equations, in general, are not amenable to analytical solutions. Therefore, one has to rely on predictive numerical simulations for solving problems of any practical relevance. Due to the presence of disparate mathematical scales in these systems, it is highly desirable to have a stable computational framework that facilitates to employ tailored numerical formulations in different regions of the computational domain.

Several advances have been made in developing numerical formulations for advection-diffusion-reaction equations, especially in the area of stabilized formulations [Codina, 2000; Augustin et al., 2011], and in the area of discrete maximum principles [Burman and Ern, 2002]. However, the main research challenge that still remains is to develop numerical methodologies for these type of problems to adequately resolve different mathematical scales in time and in space. This paper precisely aims at addressing this issue by developing a stable multi-time-step coupling framework for first-order transient systems that allows different time-steps, different time integrators and different numerical formulations in different regions of a computational domain.

Most of the prior works on multi-time-step coupling methods have focused on the second-order transient systems arising from the area of structural dynamics (e.g., see the discussion in [Karimi and Nakshatrala, 2013], and references therein). Some attempts regarding time integration of partitioned first-order systems can be found in [Nakshatrala et al., 2008, 2009]. In the former reference, a multi-time-step staggered coupling method is proposed. The method is considered as a staggered scheme as the Lagrange multipliers are calculated in an explicit fashion (i.e., based on the quantities known at prior time-levels). The stability and accuracy (especially, the control of drift along the subdomain interface) have been improved through the use of projection methods at appropriate time-levels. Since the method is a staggered scheme the obvious drawback is that the overall accuracy is first-order. However, it needs to be emphasized that the method proposed in [Nakshatrala et al., 2008] has better accuracy and stability properties than the previously proposed staggered schemes (e.g., [Piperno et al., 1995; Piperno, 1997]). In the later reference, several monolithic schemes are discussed for first-order transient systems but the treatment is restricted to transient diffusion equations (i.e., self-adjoint spatial operators) and multi-time-stepping is not allowed. Herein, we develop coupling methods that are monolithic and allow multi-time-steps.

Recently, a multi-time-step monolithic coupling method for linear elastodynamics, which is a second-order transient system, has been proposed in [Karimi and Nakshatrala, 2013]. However, developing a multi-time-step coupling method for first-order transient systems (e.g., unsteady advection-diffusion and advection-diffusion-reaction equations) will bring unique challenges. To name a few:
As shown in [Karimi and Nakshatrala, 2013], coupling explicit and implicit time-stepping schemes is always possible in the case of second-order transient systems. We will show later in this paper that such coupling is not always possible for first-order transient systems, and can be achieved only if an appropriate stabilized form of the continuity constraint is employed. We will also show that this explicit/implicit coupling for first-order transient systems will come at an expense of controlled drift.

Spatial operators in advection-diffusion-reaction equations are not self-adjoint. Symmetry and positive definiteness of the discretized operators should be carefully examined to ensure the stability of multi-time-step coupling methods. For second-order transient systems, the overall stability of the coupling method can be achieved provided the stability criterion in each subdomain is satisfied (which depends on the choice of the time-stepping scheme in the subdomain and the choice of the subdomain time-step) [Karimi and Nakshatrala, 2013]. We will show in a subsequent section that ensuring the stability of the time-stepping schemes in subdomains alone will not guarantee the overall stability of the coupling method. There is a need to place additional restrictions on the continuity constraints along the subdomain interface.

The governing equations of decomposed first-order transient systems form a system of differential/algebraic equations (DAEs) in Hessenberg form with a differential index 2. On the other hand, the governing equations for second-order transient systems form a system of DAEs with differential index 3. For more details on DAEs and associated terminology, see the brief discussion provided in subsection 3.1.1 or consult [Hairer and Wanner, 1996].

The current paper builds upon the ideas presented in [Nakshatrala et al., 2009; Karimi and Nakshatrala, 2013]. The central hypothesis on which the proposed multi-time-step coupling framework has been developed is two-fold: (i) The governing equations before the domain decomposition form a system of ordinary differential equations (ODEs). On the other hand, the governing equations resulting from the decomposition of the domain form a system of differential/algebraic equations. It needs to emphasized that many of the popular time-stepping schemes (which are developed for solving ODEs) are not appropriate for solving DAEs [Gear and Petzold, 1984; Petzold, 1992]. At least, the accuracy and the stability properties will be altered considerably. The title of an influential paper in the area of numerical solutions of DAEs by Petzold [Petzold, 1982] clearly conveys the aforementioned sentiment: “Differential/algebraic equations are not ODEs.” Therefore, we shall take a differential/algebraic equations perspective in posing the governing equations of the decomposed problems, and apply time-stepping strategies that are appropriate to solve DAEs. (ii) Developing and performance of multi-time-step coupling methods for first-order transient systems can be different from that of second-order transient systems.

The proposed monolithic multi-time-step coupling framework for first-order transient systems enjoys several attractive features, which will be illustrated in the subsequent sections by both theoretical analysis and numerical results. In the remainder of this paper, we shall closely follow the notation introduced for multi-time-step coupling in [Karimi and Nakshatrala, 2013].

2. CONTINUOUS MODEL PROBLEM: TRANSIENT ADVECTION-DIFFUSION-REACTION EQUATION

We shall consider transient advection-diffusion-reaction equation as the continuous model problem. Our choice provides an ideal setting for developing multi-time-step coupling methods for
First-order transient systems, as the governing equations pose several unique challenges. First, the relative strengths of advection, diffusion, reaction, and volumetric source introduce multiple temporal scales, which compel a need for a multi-time-step computational framework. Second, the spatial operator is not self-adjoint, which adds to the complexity of obtaining stability proofs. It needs to be emphasized that the current efforts on multi-time-step coupling have focused on second-order transient systems, and the stability analyses have been restricted to the cases in which the coefficient (i.e., “stiffness”) matrix is symmetric and positive definite [Karimi and Nakshatrala, 2013]. This will not be the case with respect to the advective-diffusive and advective-diffusive-reactive systems. Third, a numerical method to the chosen model problem can serve as a template for developing multi-time-step coupling methods for more complicated and important problems like transport-controlled bimolecular reactions, which exhibit complex spatial and temporal patterns. None of the prior works on multi-time-step coupling methods have undertaken such a comprehensive study, which this paper strives to achieve.

Consider a chemical species that is transported by both advection and diffusion processes, and simultaneously undergoes a chemical reaction. Let \( \Omega \subset \mathbb{R}^{nd} \) denote the spatial domain, where “nd” denotes the number of spatial dimensions. The boundary is denoted by \( \partial \Omega \), which is assumed to be piece-wise smooth. The gradient and divergence operators with respect to \( \mathbf{x} \in \Omega \) are, respectively, denoted by \( \text{grad}[-] \) and \( \text{div}[-] \). The time is denoted by \( t \in \mathcal{I} := (0, T) \), where \( \mathcal{I} \) is the time interval of interest. Let \( c(\mathbf{x}, t) \) denote the concentration of the chemical species. As usual, the boundary is divided into two parts: \( \Gamma^D \) and \( \Gamma^N \) such that \( \Gamma^D \cup \Gamma^N = \partial \Omega \) and \( \Gamma^D \cap \Gamma^N = \emptyset \). \( \Gamma^D \) is the part of the boundary on which concentration is prescribed (i.e., Dirichlet boundary condition), and \( \Gamma^N \) is that part of the boundary on which flux is prescribed (i.e., Neumann boundary condition). We shall denote the advection velocity vector field by \( \mathbf{v}(\mathbf{x}, t) \). The diffusivity tensor, which is a second-order tensor, is denoted by \( \mathbf{D}(\mathbf{x}) \), and is assumed to be symmetric and uniformly elliptic [Evans, 1998]. The initial boundary value problem for a transient advective-diffusive-reactive system can be written as follows:

\[
\begin{align*}
\frac{\partial c}{\partial t} + \text{div} \left[ \mathbf{v}(\mathbf{x}) \text{grad}[c] \right] + \beta c &= f(\mathbf{x}, t) \quad \text{in } \Omega \times \mathcal{I} \quad (2.1a) \\
c(\mathbf{x}, t) &= c^0(\mathbf{x}, t) \quad \text{on } \Gamma^D \times \mathcal{I} \quad (2.1b) \\
-\mathbf{n}(\mathbf{x}) \cdot \mathbf{D}(\mathbf{x}) \text{grad}[c] &= q^p(\mathbf{x}, t) \quad \text{on } \Gamma^N \times \mathcal{I} \quad (2.1c) \\
c(\mathbf{x}, t = 0) &= c_0(\mathbf{x}) \quad \text{in } \Omega \quad (2.1d)
\end{align*}
\]

where \( \mathbf{n}(\mathbf{x}) \) denotes the unit outward normal to the boundary, \( c_0(\mathbf{x}) \) is the prescribed initial concentration, \( c^0(\mathbf{x}, t) \) is the prescribed concentration on the boundary, \( q^p(\mathbf{x}, t) \) is the prescribed diffusive flux on the boundary, \( f(\mathbf{x}, t) \) is the prescribed volumetric source/sink, and \( \beta \geq 0 \) is the coefficient of decay due to a chemical reaction.

A popular non-dimensional measure to identify the relative dominance of advection is the Péclet number, which can be defined as follows:

\[
P_e(\mathbf{x}, t) := \frac{L \| \mathbf{v}(\mathbf{x}, t) \|}{D(\mathbf{x})} \quad (2.2)
\]

where \( L \) is the characteristic length, \( D \) is the characteristic diffusivity, and \( \| \cdot \| \) denotes the standard 2-norm. In the case of anisotropic diffusion tensor, \( D(\mathbf{x}) \) can be taken as the minimum eigenvalue of the diffusivity tensor at \( \mathbf{x} \) (i.e., \( D(\mathbf{x}) = \min \{ \kappa \mid \det (\mathbf{D}(\mathbf{x}) - \kappa \mathbf{I}) = 0 \} \)). Clearly, the higher the Péclet number the greater will be the relative dominance of advection. A non-dimensional quantity
to measure the relative dominance of the chemical reaction is the Damkohler number, which takes the following form:

\[
D_a := \frac{\beta L^2}{D(x)}
\]  

(2.3)

In the context of numerical solutions, the characteristic length is typically associated with an appropriate measure of the mesh size. A popular choice under the finite element method is \( L = h_e / 2 \), where \( h_e \) is the diameter of the circumscribed circle of the element and the factor 2 is for convenience. This choice gives rise to what is commonly referred to as the element Péclet number (e.g., see [Donea and Huerta, 2003]):

\[
P_e(x, t) = \frac{h_e \|v(x, t)\|}{2D(x)}
\]  

(2.4)

which will be used in subsequent sections, especially, in defining stabilized weak formulations. We shall employ the semi-discrete methodology [Zienkiewicz and Taylor, 1989] based on the finite element method for spatial discretization and the trapezoidal family of time-stepping schemes for the temporal discretization.

2.1. Trapezoidal family of time-stepping schemes. The time interval of interest is divided into \( N \) sub-intervals such that

\[
I = (0, T] = \bigcup_{n=1}^{N} (t_{n-1}, t_n]
\]  

(2.5)

where \( t_0 = 0 \) and \( t_N = T \). To make the presentation simple, we shall assume that the sub-intervals are uniform. That is,

\[
\Delta t = t_n - t_{n-1} \quad \forall n = 1, \cdots, N
\]  

(2.6)

where \( \Delta t \) will be referred to as the time-step. However, it should be noted that the methods presented in this paper can be easily extended to variable time-steps. The nodal vector of the primary variable (which, in our case, will be the concentration) and the corresponding time derivative at discrete time levels are denoted as follows:

\[
d(n) = u(t = t_n), \quad v(n) = \left. \frac{du}{dt} \right|_{t = t_n}
\]  

(2.7)

The trapezoidal family of time-stepping schemes can be compactly written as follows:

\[
d^{(n+1)} = d^{(n)} + \Delta t \left( (1 - \vartheta)v^{(n)} + \vartheta v^{(n+1)} \right)
\]  

(2.8)

where \( \vartheta \in [0, 1] \) is a user-specified parameter. Some popular time-stepping schemes under the trapezoidal family include the forward Euler (\( \vartheta = 0 \)), the midpoint rule (\( \vartheta = 1/2 \)), and the backward Euler (\( \vartheta = 1 \)). The forward Euler is an explicit scheme, and the midpoint and the backward Euler schemes are implicit. The stability and accuracy properties of these time-stepping schemes in the context of ordinary differential equations are well-known (e.g., see [Hairer and Wanner, 2009]).
2.2. Weak formulations. We will now present several weak formulations for the initial boundary value problem given by equations (2.1a)–(2.1d), which will be used in the remainder of the paper. Since we address advection-dominated and reaction-dominated problems, we will present two popular stabilized weak formulations in addition to the Galerkin weak formulation. Let us introduce the following function spaces:

\[ C_t := \{ c(x, \cdot) \in H^1(\Omega) \mid c(x, t) = c^p(x, t) \text{ on } \Gamma^D \} \]  
\[ W := \{ w(x) \in H^1(\Omega) \mid w(x) = 0 \text{ on } \Gamma^D \} \]

where \( H^1(\Omega) \) is a standard Sobolev space \( \text{[Brezzi and Fortin, 1991]} \). For convenience, we shall denote the standard \( L_2 \) inner-product over a set \( K \) as follows:

\[ (a; b)_K \equiv \int_K a \cdot b \, dK \]  

The subscript \( K \) will be dropped if the set is the entire spatial domain (i.e., \( K = \Omega \)).

2.2.1. Galerkin weak formulation. The Galerkin formulation for the initial boundary value problem (2.1a)–(2.1d) can be written as follows: Find \( c(x, t) \in C_t \) such that we have

\[ (w; \partial c/\partial t) + (w; \text{div} [vc]) + (\text{grad} [w] ; D(x) \text{grad} [c]) + (w; \beta c - f) = + (w; q^p)_{\Gamma_N} \quad \forall w(x) \in W \]  

(2.11)

It is well-known that the Galerkin formulation may exhibit numerical instabilities (e.g., spurious node-to-node oscillations) for non-self-adjoint spatial operators like the advective-diffusive and advective-diffusive-reactive systems. The reason can be attributed to the presence of boundary layers and interior layers in the solutions of these systems when advection is relatively more dominant than the diffusion and reaction processes. Designing stable numerical formulations for advection-diffusion and advection-diffusion-reaction problems is still an active area of research (e.g., see \( \text{[Turner et al., 2011; Franca et al., 2006; Gresho and Sani, 2000]} \)). This paper is not concerned with developing a new stabilized formulation.

In order to avoid spurious oscillations and obtain accurate numerical solutions, it is sufficient to have the element Péclet number to be less than unity. To put it differently, if the element Péclet number is greater than unity, the computational mesh may not be adequate to resolve the steep gradients due to boundary layers and internal layers, which are typical in the solutions of advection dominated problems. One can always achieve smaller values for the element Péclet number by refining the computational mesh adequately. However, in some cases, the mesh has to be refined so fine that it may be computationally prohibitive to employ such a mesh.

2.2.2. Stream-Upwind/Petrov-Galerkin (SUPG) weak formulation. In order to alleviate the instability of the Galerkin formulation in (2.11) for advection-dominated problems, many alternative methods have been proposed in the past few decades, for a short description and comparison of these methods see \( \text{[Augustin et al., 2011]} \). In this paper we shall employ the SUPG formulation for enhanced stability of the weak formulation. The SUPG formulation reads as follows: Find \( c(x, t) \in C_t \) such that we have

\[ (w; \partial c/\partial t) + (w; \text{div} [vc]) + (\text{grad} [w] ; D(x) \text{grad} [c]) + (w; \beta c) \]
\[ + \sum_{e=1}^{Nele} (\tau_{\text{SUPG}} \cdot \text{grad} [w] ; \partial c/\partial t + \text{div} [vc - D(x) \text{grad} [c]] + \beta c - f)_{\Omega_e} \]
\[ = (w; f) + (w; q^p)_{\Gamma_N} \quad \forall w(x) \in W \]  

(2.12)
where \( \text{Nele} \) is the number of elements, and \( \tau_{\text{SUPG}} \) is the stabilization parameter under the SUPG formulation. We shall use the stabilization parameter proposed in [John and Knobloch, 2007]:

\[
\tau_{\text{SUPG}} = \frac{h_e}{2\parallel v \parallel} \xi_0 \left( P^h_e \right), \quad \xi_0 (\chi) = \coth (\chi) - \frac{1}{\chi}
\]  

(2.13)

where \( h \) is the element length, and \( \xi_0 \) is known as the upwind function.

2.2.3. Galerkin/least-squares (GLS) weak formulation. The GLS formulation reads as follows: Find \( c(x,t) \in C_t \) such that we have

\[
(w; \partial c/\partial t) + (w; \text{div}[vc]) + (\text{grad}[w]; D(x)\text{grad}[c]) + (w; \beta c)
\]

\[
+ \sum_{e=1}^{\text{Nele}} \left( \frac{w}{\Delta t} + \text{div}[vw - D(x)\text{grad}[w]] + \beta w; \tau_{\text{GLS}} (\partial c/\partial t + \text{div}[vc - D(x)\text{grad}[c]] + \beta c - f) \right)_{\Omega_e}
\]

\[
= (w; f) + (w; q^p)_{\Gamma_N} \forall w(x) \in W
\]

(2.14)

where \( \tau_{\text{GLS}} \) is the stabilization parameter under the GLS formulation, and \( \Delta t \) is the time-step. In this paper, we shall take \( \tau_{\text{GLS}} = \tau_{\text{SUPG}} \), which is a common practice. It should be emphasized that an optimal choice of stabilization parameter for stabilized formulations in two- and three-dimensions is still an active area of research (e.g., see [Augustin et al., 2011]).

3. PROPOSED MULTI-TIME-STEP COMPUTATIONAL FRAMEWORK

The proposed multi-time-step computational framework is built based on the semi-discrete methodology [Zienkiewicz and Taylor, 1989] and the dual Schur domain decomposition method [Toselli and Widlund, 2004]. The semi-discrete methodology converts the partial differential equations into a system of ordinary differential equations. For spatial discretization of the problem at hand, one can use either the Galerkin formulation or a stabilized formulation depending on the relative strengths of transport processes and the decay coefficient due to chemical reactions. The dual Schur domain decomposition is an elegant way to handle decomposition of the computational domain into subdomains through Lagrange multipliers.

3.1. Domain decomposition and the resulting equations. In order to facilitate multi-time-step coupling, we shall decompose the computational domain into \( S \) non-overlapping subdomains such that we have

\[
\overline{\Omega} = \bigcup_{i=1}^{S} \overline{\Omega}_i \quad \text{and} \quad \Omega_i \cap \Omega_j = \emptyset \text{ for } i \neq j
\]

(3.1)

where a superposed bar denotes the set closure. Furthermore, the meshes in all subdomains are assumed to be conforming along the subdomain interface (see Figure 1). We shall use signed Boolean matrices to write the compatibility constraints along the subdomain interface, as they provide a systematic way of writing the interface constraints as a system of linearly independent equations. In addition, the mathematical structure of the resulting equations is suitable for a mathematical analysis. The entries of a signed Boolean matrix are either -1, 0, or 1, and each row has at most one non-zero entry. However, it needs to be emphasized that a signed Boolean matrix is never constructed explicitly in a computer implementation, as it is computationally not efficient to store such a matrix. For more details on signed Boolean matrices see [Nakshatrala et al., 2008].
In a time-continuous setting, the governing equations after spatial discretization can be written as follows:

\[ M_i \dot{u}_i(t) + K_i u_i(t) = f_i(t) + C^T_i \lambda(t) \quad i = 1, \ldots, S \] (3.2a)

\[ \sum_{i=1}^S C_i u_i(t) = 0 \] (3.2b)

where a superposed dot denotes a derivative with respect to time, the subscript \( i \) denotes the subdomain number, the nodal concentration vector of the \( i \)-th subdomain is denoted by \( u_i \), the capacity matrix of the \( i \)-th subdomain is denoted by \( M_i \), and the transport matrix of the \( i \)-th subdomain is denoted by \( K_i \). \( \lambda \) denotes the vector of Lagrange multipliers, and \( C_i \) denotes the signed Boolean matrix for the \( i \)-th subdomain. Let the number of degrees-of-freedom in the \( i \)-th subdomain be denoted by \( N_i \), and the number of degrees-of-freedom on the subdomain interface be denoted by \( N_\lambda \). The size of \( u_i \) is \( N_i \times 1 \), and both the capacity and transport matrices of the \( i \)-th subdomain will be of the size \( N_i \times N_i \). The size of \( \lambda \) will be \( N_\lambda \times 1 \), and the size of the signed Boolean matrix \( C_i \) will be \( N_\lambda \times N_i \).

It is imperative to note that the governing equations (3.2a) – (3.2b), which arise from domain decomposition, form a system of differential/algebraic equations (DAEs). For completeness and future reference we now present the necessary details about differential/algebraic equations.

**3.1.1. Differential/algebraic equations.** A differential/algebraic equation is an equation involving a set of independent variable, an unknown function of the independent variables, and derivatives of the functions with respect to the independent variables. Clearly, ordinary differential equations, and algebraic equations form subclasses of differential/algebraic equations. In this paper, we are concerned with first-order differential/algebraic equations. Mathematically, a DAE in first-order form takes the following general form:

\[ f(\dot{x}(t), x(t), t) = 0 \quad t \in I \] (3.3)

where \( t \) is the independent variable, and \( x(t) \) is the unknown function. It is well-known that solving a system of differential/algebraic equations numerically can be more difficult than solving a system of ordinary differential equations [Hairer and Wanner, 1996; Petzold, 1982]. A notion which is popularly employed to measure the difficulty of obtaining numerical solutions to a particular DAE is the *differential index*. The differential index of a DAE is the number of times one has to take derivatives of equation (3.3) in order to be able to derive an ODE by mere algebraic manipulations. It is obvious that a system of ODEs will have differential index of zero. A special form of DAEs which is of interest to us in this paper is the Hessenberg index-2 DAE. It has the following form:

\[ \dot{x} = f(x, y, t) \] (3.4a)

\[ 0 = g(x) \] (3.4b)

which is in fact an ODE system along with an algebraic constraint, and \( y \) is an algebraic variable. This paper concerns with differential/algebraic equations of differential index two or lower. Many of the constrained mechanical systems can be modeled using DAEs (e.g., see Geradin and Cardona, 2001). In the case of coupling algorithms the compatibility of subdomains along the interfaces will appear as an algebraic constraint to the ODEs obtained from a finite element discretization. It is not possible to solve differential/algebraic equations analytically unless in some very special cases, and one has to resort to numerical solutions. In this paper, we shall restrict to time-stepping schemes from the trapezoidal family. However, the corresponding properties when applied
to differential/algebraic equations can be different, and for a detailed discussion on this topic see [Hairer and Wanner, 1996].

3.2. Time discretization. We now construct two multi-time-step coupling methods that can handle multiple subdomains, and can allow the use of different time-steps, different time-integrators and/or different numerical formulation in different subdomains. To this end, the time interval of interest is divided into non-overlapping intervals whose end points will be referred to as system time-levels. The algebraic compatibility constraints will be enforced at the system time-levels. For convenience, we shall assume the system time-levels are uniform. The $n$-th system time-level will be denoted by $t_n$ and can be written as follows:

$$t^{(n)} = n\Delta t \quad n = 0, 1, \ldots, N$$  \hfill (3.5)

where $\Delta t$ is called the system time-step. The numerical time-integration of each subdomain will advance by the subdomain time-step. The subdomain time-step of the $i$-th subdomain will be denoted by $\Delta t_i$. Note that $\Delta t \geq \Delta t_i \forall i$. Furthermore, we shall assume that the ratio between the system and subdomain time-step is a natural number, and is denoted by $\eta_i$. That is,

$$\eta_i = \frac{\Delta t}{\Delta t_i}$$  \hfill (3.6)

Figure 2 presents a pictorial description of the system and subdomain time-steps. In the rest of the paper, we will use the following notation to show the value of a variable at a time-level:

$$x^{(n+\frac{j}{m})} = x(t^{(n)} + j\Delta t_i)$$  \hfill (3.7)

$$t^{(n+\frac{j}{m})} = t^{(n)} + j\Delta t_i$$  \hfill (3.8)

Note that because of the enforcement of compatibility constraint at system time-levels only, the Lagrange multipliers can only be calculated at system time-levels. Herein, we will use the Taylor expansion to approximate the Lagrange multipliers in a system time-step. This approximation will be in the following form:

$$\lambda^{(n+\frac{j}{m})} = \left(1 - \frac{j + 1}{\eta_i}\right)\lambda^{(n)} + \left(\frac{j + 1}{\eta_i}\right)\lambda^{(n+1)}$$  \hfill (3.9)

As discussed earlier, coupling explicit and implicit time-stepping schemes is not straightforward in the case of first-order transient systems as compared with second-order systems. The proposed computational framework will employ different compatibility constraints in order to enforce continuity and to make an explicit/implicit coupling possible.

3.3. Mathematical statements of the proposed coupling methods. The compatibility constraints along the subdomain interface will be enforced at system time-levels. Mathematically, the time discretization of compatibility constraints reads as follows:

$$\sum_{i=1}^{S} C_i d_i^{(n+1)} = 0 \quad \forall n$$ \quad \text{d-continuity method} \hfill (3.10)

$$\sum_{i=1}^{S} C_i \left(\psi_i^{(n+1)} + \frac{\alpha}{\Delta t} d_i^{(n+1)}\right) = 0 \quad \forall n$$ \quad \text{Baumgarte stabilization} \hfill (3.11)
where $\alpha > 0$ is the Baumgarte stabilization parameter. The proposed coupling method based on $d$-continuity will read as follows: Find $(v_i^{(n+1)/\eta_i}, d_i^{(n+1)/\eta_i}, \lambda^{(n+1)})$ for $n = 1, ..., N; j = 0, ..., \eta_i - 1; i = 1, ..., S$ such that we have

\begin{align}
M_i v_i^{(n+1)/\eta_i} + K_i d_i^{(n+1)/\eta_i} &= f_i^{(n+1)/\eta_i} + C_i^T \lambda^{(n+1)/\eta_i} \\
d_i^{(n+1)/\eta_i} &= d_i^{(n+1)/\eta_i} + \Delta t_i \left( -\vartheta_i (1-\vartheta_{i,j}) v_i^{(n+1)/\eta_i} + \vartheta_i v_i^{(n+1)/\eta_i} \right) \\
\lambda^{(n+1)/\eta_i} &= \left( 1 - \dfrac{j+1}{\eta_i} \right) \lambda^{(n)} + \left( \dfrac{j+1}{\eta_i} \right) \lambda^{(n+1)} \\
\sum_{i=1}^{S} C_i (v_i^{(n+1)} + \alpha \dfrac{\Delta t_i}{\Delta t} d_i^{(n+1)}) &= 0
\end{align}

(3.12a) (3.12b) (3.12c) (3.12d)

The proposed coupling method based on the Baumgarte stabilization will read as follows: Find $(v_i^{(n+1)/\eta_i}, d_i^{(n+1)/\eta_i}, \lambda^{(n+1)})$ for $n = 1, ..., N; j = 0, ..., \eta_i - 1; i = 1, ..., S$ such that we have

\begin{align}
M_i v_i^{(n+1)/\eta_i} + K_i d_i^{(n+1)/\eta_i} &= f_i^{(n+1)/\eta_i} + C_i^T \lambda^{(n+1)/\eta_i} \\
d_i^{(n+1)/\eta_i} &= d_i^{(n+1)/\eta_i} + \Delta t_i \left( -\vartheta_i (1-\vartheta_{i,j}) v_i^{(n+1)/\eta_i} + \vartheta_i v_i^{(n+1)/\eta_i} \right) \\
\lambda^{(n+1)/\eta_i} &= \left( 1 - \dfrac{j+1}{\eta_i} \right) \lambda^{(n)} + \left( \dfrac{j+1}{\eta_i} \right) \lambda^{(n+1)} \\
\sum_{i=1}^{S} C_i (v_i^{(n+1)} + \alpha \dfrac{\Delta t_i}{\Delta t} d_i^{(n+1)}) &= 0
\end{align}

(4.1a) (4.1b) (4.1c) (4.1d)

We shall perform a systematic theoretical analysis of the proposed multi-time-step coupling methods in the next section.

4. A THEORETICAL STUDY ON THE PROPOSED METHODS

4.1. Notation. The jump and average operators over the $i$-th subdomain time-step are, respectively, defined as follows:

\begin{align}
\left[ x^{(n+1)/\eta_i} \right]_i &= x^{(n+1)/\eta_i} - x^{(n+1)/\eta_i} \\
\left[ x^{(n+1)/\eta_i} \right]_i &= \dfrac{1}{2} \left( x^{(n+1)/\eta_i} + x^{(n+1)/\eta_i} \right)
\end{align}

(4.1a) (4.1b)

One can similarly define the jump and average operators over a system time-step as follows:

\begin{align}
\left[ x^{(n)} \right] &= x^{(n+1)} - x^{(n)} = \sum_{j=0}^{\eta_i-1} \left[ x^{(n+1)/\eta_i} \right]_i \\
\left\langle x^{(n)} \right\rangle &= \dfrac{1}{2} \left( x^{(n+1)} + x^{(n)} \right)
\end{align}

(4.2a) (4.2b)

Let $S$ be a symmetric matrix, then we have the following identity:

\begin{align}
\left\langle x \right\rangle^T S x = \dfrac{1}{2} x^T S x
\end{align}

(4.3)
The trapezoidal family of time-stepping schemes applied over a subdomain time-step can be compactly written as follows:

\[
\left[ d_i^{(n+\frac{1}{2})} \right] = \Delta t_i \left( \left[ v_i^{(n+\frac{1}{2})} \right] + \left( \vartheta_i - \frac{1}{2} \right) \left[ v_i^{(n+1)} \right] \right) \quad (4.4)
\]

### 4.2. Stability analysis.

Consistency of the proposed coupling methods is trivial by construction. Hence, for convergence, it is necessary and sufficient to show that the proposed coupling methods are stable. We now show that both the proposed coupling methods are indeed stable using the energy method [Wood, 1990]. For numerical stability analysis, it is common to assume that forcing function is zero. Therefore, we take \( f_i(t) = 0 \) in all the subdomains. Before we can provide stability proofs for the proposed coupling methods, we need to present an important property that the transport matrices enjoy under the three weak formulations that are outlined in the previous section. This property will play a crucial role in the stability analysis. We provide a proof for the Galerkin weak formulation.

**Lemma 1.** Consider the Galerkin weak formulation given by equation (2.11). If the advection velocity satisfies \( \text{div}[\mathbf{v}] \geq 0 \), and the diffusivity tensor \( \mathbf{D}(\mathbf{x}) \) is symmetric and positive definite, then the symmetric part of the transport matrix resulting after the finite element discretization will be positive semi-definite.

**Proof.** Let us denote the spatial operator of the advective-diffusive system as follows:

\[
\mathcal{L}[c] := \text{div}[\mathbf{v}c] - \text{div}[\mathbf{D}(\mathbf{x})\text{grad}[c]]
\]

(4.5)

It is easy to show that the adjoint of the spatial operator takes the following form:

\[
\mathcal{L}^*[c] = -\mathbf{v} \cdot \text{grad}[c] - \text{div}[\mathbf{D}^T(\mathbf{x})\text{grad}[c]]
\]

(4.6)

Noting the symmetry of diffusivity tensor, the symmetric part of the spatial operator takes the following form:

\[
\mathcal{L}[c] = \mathcal{L}[c] + \mathcal{L}^*[c] = \frac{1}{2} \text{div}[\mathbf{v}]c - \text{div}[\mathbf{D}(\mathbf{x})\text{grad}[c]]
\]

(4.7)

The coefficient (i.e., “stiffness”) matrix corresponding to the operator \( \mathcal{L}[c] \) over a finite element \( \Omega_e \) can be written as follows:

\[
\mathbf{K}_e = \int_{\Omega_e} \frac{1}{2} \text{div}[\mathbf{v}] \mathbf{N}^T(\mathbf{x}) \mathbf{N}(\mathbf{x}) \, d\Omega + \int_{\Omega_e} \mathbf{B}(\mathbf{x}) \mathbf{D}(\mathbf{x}) \mathbf{B}^T(\mathbf{x}) \, d\Omega
\]

(4.8)

where \( \mathbf{N}(\mathbf{x}) \) is the row vector containing shape functions, and \( \mathbf{B}(\mathbf{x}) \) is the matrix containing the derivatives of shape functions with respect to \( \mathbf{x} \). Since \( \text{div}[\mathbf{v}] \geq 0 \) and \( \mathbf{D}(\mathbf{x}) \) is positive definite, the matrix \( \mathbf{K}_e \) will be positive semi-definite. Since \( \mathbf{D}(\mathbf{x}) \) is symmetric, the matrix \( \mathbf{K}_e \) is symmetric. The assembly procedure preserves the positive semi-definiteness when the local matrices are mapped to a global matrix.

One can similarly show that the symmetric part of the transport matrix under the GLS formulation is also positive semi-definite. On the other hand, the symmetric part of the transport matrix under the SUPG formulation will be positive semi-definite only if the diffusivity tensor is constant, and low-order simplicial elements (e.g., two-node element, three-node triangle element, four-node tetrahedron element) are employed.
THEOREM 2 (Stability of the d-continuity coupling method). Under the proposed multi-time-step method with d-continuity, the rate variables \( v_i \) will remain bounded if \( 1/2 \leq \vartheta_i \leq 1 \forall i. \)

PROOF. The governing equations under the d-continuity coupling method can be written as follows:

\[
M_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + K_i \left[ d_i^{(n+\frac{1}{2})} \right]_i = \frac{1}{\eta_i} C_i^T \left[ \lambda^{(n)} \right]_i
\]

\[
\sum_{i=1}^{S} C_i \left[ d_i^{(n)} \right] = 0
\]  

(Note that we have used the fact that the Lagrange multipliers are interpolated linearly over a system time-step. For convenience, let us denote)

\[ Q_i := M_i + 2 \left( \vartheta_i - \frac{1}{2} \right) \Delta t_i \text{sym}[K_i] \]

Clearly, the matrix \( Q_i \) is symmetric, as the matrix \( M_i \) is symmetric. Since the matrix \( M_i \) is positive definite, the symmetric part of \( K_i \) is positive semi-definite, \( \vartheta_i \geq 1/2 \), and \( \Delta t_i > 0 \); one can conclude that the matrix \( Q_i \) is positive definite.

Premultiplying both sides of equation (4.9a) by \( \left[ d_i^{(n+\frac{1}{2})} \right]_i \) and using equation (4.4), we have the following equation:

\[
\left( v_i^{(n+\frac{1}{2})} \right)_i^T Q_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + \left( \vartheta_i - \frac{1}{2} \right) \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \left( M_i + (\vartheta_i - \frac{1}{2}) \Delta t_i \text{sym}[K_i] \right) \left[ v_i^{(n+\frac{1}{2})} \right]_i
+ \Delta t_i \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \text{sym}[K_i] \left[ v_i^{(n+\frac{1}{2})} \right]_i = \frac{1}{\Delta t} \left[ \lambda^{(n)} \right]_i^T C_i \left[ d_i^{(n+\frac{1}{2})} \right]_i
\]

(4.11)

Since the symmetric part of \( K_i \) is positive semi-definite, and \( \Delta t_i > 0 \), we have the following inequality:

\[
\left( v_i^{(n+\frac{1}{2})} \right)_i^T Q_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + \left( \vartheta_i - \frac{1}{2} \right) \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \left( M_i + (\vartheta_i - \frac{1}{2}) \Delta t_i \text{sym}[K_i] \right) \left[ v_i^{(n+\frac{1}{2})} \right]_i
\]

\[
\leq \frac{1}{\Delta t} \left[ \lambda^{(n)} \right]_i^T C_i \left[ d_i^{(n+\frac{1}{2})} \right]_i
\]

(4.12)

Noting that \( \vartheta_i \geq 1/2 \), the matrix \( M_i \) is positive definite, the symmetric part of \( K_i \) is positive semi-definite, and \( \Delta t_i > 0 \); one can obtain the following inequality:

\[
\left( v_i^{(n+\frac{1}{2})} \right)_i^T Q_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \leq \frac{1}{\Delta t} \left[ \lambda^{(n)} \right]_i^T C_i \left[ d_i^{(n+\frac{1}{2})} \right]_i
\]

(4.13)

Summing over all the subdomain time levels within a system time-step, summing over the number of subdomains, and noting the compatibility conditions under the d-continuity coupling method given by equation (4.9b); we obtain the following inequality:

\[
\sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left( v_i^{(n+\frac{1}{2})} \right)_i^T Q_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \leq 0
\]
By noting the symmetry of the matrix $Q_i$, the above inequality can be rewritten as following:

$$\sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left[ \left( v_i^{(n+\frac{j}{\eta_i})} \right)^T Q_i v_i^{(n+\frac{j}{\eta_i})} \right] \leq 0 \quad (4.15)$$

By executing the telescopic summation, we obtain the following:

$$\sum_{i=1}^{S} \left[ \left( v_i^{(n)} \right)^T Q_i v_i^{(n)} \right] \leq 0 \quad (4.16)$$

This further implies that

$$\sum_{i=1}^{S} v_i^{(n)}^T Q_i v_i^{(n)} \leq \sum_{i=1}^{S} v_i^{(n-1)}^T Q_i v_i^{(n-1)} \leq \cdots \leq \sum_{i=1}^{S} v_i^{(0)}^T Q_i v_i^{(0)} \quad (4.17)$$

Noting the boundedness of $v_i^{(0)}$ and the matrices $Q_i$ ($i = 1, \ldots, S$) are positive definite, one can conclude the boundedness of $v_i^{(n)} \forall i$ and $\forall n$. This concludes the proof. \(\square\)

**Remark 3.** One cannot relax the condition $\vartheta_i \geq 1/2$. It should be noted that one would obtain numerical instability if this condition is violated. This will be the case even if one does not employ subcycling [Nakshatrala et al., 2009].

We now assess the stability of the proposed coupling method based on the Baumgarte stabilization. We are able to construct a proof only for the case in which the matrices $K_i$ are symmetric. This means that the proof does not hold for the case in which advection is present. However, the numerical results presented in a subsequent section show that the coupling method based on the Baumgarte stabilization provide stable solutions even in the presence of advection. It is therefore a good research problem to theoretically assess the stability of the coupling method based on the Baumgarte stabilization.

**Theorem 4 (Stability of the proposed method with Baumgarte stabilization).** Under the proposed multi-time-step method with Baumgarte stabilization, the rate variables $v_i$, will remain bounded if one chooses $\Delta t \leq \Delta t_i^{\text{critical}}$ and $\alpha \leq \alpha_{\text{max}}$ where

$$\Delta t_i^{\text{critical}} := \begin{cases} \frac{2}{(1-2\vartheta_i)\omega_i} & \text{if } 0 \leq \vartheta_i < 1/2 \\ \infty & \text{if } 1/2 \leq \vartheta_i \leq 1 \end{cases} \quad (4.18a)$$

$$\alpha_{\text{max}} := \min \left\{ \frac{2\eta_i}{1-2\vartheta_i} : 0 \leq \vartheta_i < 1/2 \right\} \quad (4.18b)$$

and $\omega_i = \max \{ \omega : \det (\omega I_i - M_i^{-1} K_i) = 0 \}$. It is assumed that the matrices $K_i$ ($i = 1, \ldots, S$) are symmetric and positive semi-definite.

**Proof.** The governing equations under the proposed coupling method based on the Baumgarte stabilization can be written as follows:

$$M_i \left[ v_i^{(n+\frac{j}{\eta_i})} \right]_i + K_i \left[ d_i^{(n+\frac{j}{\eta_i})} \right]_i = \frac{1}{\eta_i} C_i^T \left[ \chi_i^{(n)} \right] \quad (4.19a)$$

$$\sum_{i=1}^{S} C_i \left[ v_i^{(n)} \right]_i + \frac{\alpha}{\Delta t} \left[ d_i^{(n)} \right]_i = 0 \quad (4.19b)$$
Premultiplying both sides of equation (4.19a) by \( \left[ v_i^{(n+\frac{1}{2})} \right]_i + \alpha \Delta t \left[ d_i^{(n+\frac{1}{2})} \right]_i \), we obtain the following:

\[
\begin{align*}
\left[ v_i^{(n+\frac{1}{2})} \right]_i^T & \cdot M_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + \alpha \Delta t \left[ d_i^{(n+\frac{1}{2})} \right]_i^T \cdot M_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot K_i \left[ d_i^{(n+\frac{1}{2})} \right]_i \\
+ \alpha \Delta t \left[ d_i^{(n+\frac{1}{2})} \right]_i^T & \cdot K_i \left[ d_i^{(n+\frac{1}{2})} \right]_i = \frac{1}{\eta_i} \left[ \lambda^{(n)} \right]_i^T \cdot C_i \left( \left[ v_i^{(n+\frac{1}{2})} \right]_i + \alpha \Delta t \left[ d_i^{(n+\frac{1}{2})} \right]_i \right)
\end{align*}
\]

Employing equation (4.4) yields:

\[
\begin{align*}
\left[ v_i^{(n+\frac{1}{2})} \right]_i^T & \cdot \left( 1 + \alpha \left( \vartheta_i - \frac{1}{2} \right) \frac{\Delta t_i}{\Delta t} \right) M_i + \Delta t_i \left( \vartheta_i - \frac{1}{2} \right) \left( 1 + \alpha \left( \vartheta_i - \frac{1}{2} \right) \frac{\Delta t_i}{\Delta t} \right) K_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \\
+ \left[ v_i^{(n+\frac{1}{2})} \right]_i^T & \cdot \left( \alpha \frac{\Delta t_i}{\Delta t} M_i + \Delta t_i \left( 1 + 2 \alpha \left( \vartheta_i - \frac{1}{2} \right) \frac{\Delta t_i}{\Delta t} \right) K_i \right) \left[ v_i^{(n+\frac{1}{2})} \right]_i + \alpha \Delta t_i^2 \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot K_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \\
= \frac{1}{\eta_i} \left[ \lambda^{(n)} \right]_i^T \cdot C_i \left( \left[ v_i^{(n+\frac{1}{2})} \right]_i + \alpha \Delta t \left[ d_i^{(n+\frac{1}{2})} \right]_i \right)
\end{align*}
\]

Noting that the parameters \( \alpha, \Delta t_i, \Delta t \) are strictly positive and \( K_i \) is a positive semi-definite matrix, we have the following inequality:

\[
\begin{align*}
\left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot P_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot U_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \leq \left[ \lambda^{(n)} \right]_i^T \cdot C_i \left( \left[ v_i^{(n+\frac{1}{2})} \right]_i + \alpha \Delta t \left[ d_i^{(n+\frac{1}{2})} \right]_i \right)
\end{align*}
\]

where

\[
P_i := \left( \eta_i + \alpha \left( \vartheta_i - \frac{1}{2} \right) \right) M_i + \Delta t_i \left( \vartheta_i - \frac{1}{2} \right) \left( \eta_i + \alpha \left( \vartheta_i - \frac{1}{2} \right) \right) K_i
\]

\[
U_i := \alpha M_i + \Delta t_i \left( \eta_i + 2 \alpha \left( \vartheta_i - \frac{1}{2} \right) \right) K_i
\]

Summing over all the subdomains (i.e., summing over \( i \)) and summing over the subdomain time-levels (i.e., summing over \( j \)), one can obtain the following inequality:

\[
\sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot P_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + \sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot U_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \leq \left[ \lambda^{(n)} \right]_i^T \cdot \sum_{i=1}^{S} C_i \left( \left[ v_i^{(n)} \right]_i + \alpha \Delta t \left[ d_i^{(n)} \right]_i \right)
\]

The compatibility condition along the subdomain interface in the form given by equation (4.24) implies that

\[
\sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot P_i \left[ v_i^{(n+\frac{1}{2})} \right]_i + \sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot U_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \leq 0
\]

From the hypothesis of the theorem, it is easy to show that the matrix \( P_i \) is positive semi-definite. This implies that we have the following inequality:

\[
\sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left[ v_i^{(n+\frac{1}{2})} \right]_i^T \cdot U_i \left[ v_i^{(n+\frac{1}{2})} \right]_i \leq 0
\]
It is easy to check that \( U_i \) is symmetric, which implies the following:

\[
\sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \left(v_i^{(n+\frac{j}{\eta_i})} \right)^T U_i \left(v_i^{(n+\frac{j}{\eta_i})} \right) = \sum_{i=1}^{S} \sum_{j=0}^{\eta_i-1} \frac{1}{2} \left[v_i^{(n+\frac{j}{\eta_i})} \right]^T U_i v_i^{(n+\frac{j}{\eta_i})} = \frac{1}{2} \sum_{i=1}^{S} \left[v_i^{(n)} \right]^T U_i v_i^{(n)} \leq 0 \quad \forall n
\]

This further implies that

\[
\sum_{i=1}^{S} v_i^{(n)} U_i v_i^{(n)} \leq \sum_{i=1}^{S} v_i^{(n-1)} U_i v_i^{(n-1)} \leq \cdots \leq \sum_{i=1}^{S} v_i^{(0)} U_i v_i^{(0)} \quad (4.28)
\]

Noting that the matrices \( U_i \) (\( i = 1, \cdots, S \)) are positive definite, and the initial rates \( v_i^{(0)} \) are bounded; one can conclude that the rate variables will remain bounded at all time-levels.

4.3. Bounds on drifts in concentrations and rate variables. A well-known phenomenon appearing in numerical solutions of DAEs is the drift in the compatibility/constraint equations [Hairer and Wanner, 1996]. In our case, the drift will manifest as discontinuity in the primary and/or rate variables along the subdomain interface. The drifts will be different for two proposed coupling methods, as they differ in handling compatibility conditions along the subdomain interface. Herein, we shall ignore subcycling (i.e., \( \eta_i = 1 \ \forall i \)), and assume that \( \vartheta_i = \vartheta \ \forall i \). The following notation is employed:

\[
d_{\text{drift}}^{(n)} := \sum_{i=1}^{S} C_i d_i^{(n)}, \quad v_{\text{drift}}^{(n)} := \sum_{i=1}^{S} C_i v_i^{(n)}
\]

Under the \( d \)-continuity coupling method, by construction of the method, there is no drift in the primary variable (i.e., concentration) along the subdomain interface at all system time levels. The drift in the rate satisfy the following recursive relation:

\[
v_{\text{drift}}^{(n+1)} = \left(1 - \frac{1}{\vartheta}\right) v_{\text{drift}}^{(n)} \quad (4.30)
\]

Note that if the implicit Euler method (i.e., \( \vartheta = 1 \)) is employed then the drifts at system time-levels will be zero in both concentrations and rate variables.

Under the proposed coupling method with Baumgarte stabilization, the following recursive relations hold:

\[
d_{\text{drift}}^{(n+1)} = \frac{1}{1 + \alpha \vartheta} d_{\text{drift}}^{(n)} + \frac{\Delta t (1 - \vartheta)}{1 + \alpha \vartheta} v_{\text{drift}}^{(n)} \quad (4.31a)
\]

\[
v_{\text{drift}}^{(n+1)} = -\frac{\alpha}{\Delta t (1 + \alpha \vartheta)} d_{\text{drift}}^{(n)} - \frac{\alpha (1 - \vartheta)}{1 + \alpha \vartheta} v_{\text{drift}}^{(n)} \quad (4.31b)
\]

which imply that choosing larger \( \alpha \) will decrease drifts in both the concentration and its rate. It should be noted that subcycling, and mixed methods can have adverse effects on the drifts. That is, the drifts can be worse than predictions made by the above bounds. However, the above bounds can be valuable to check a computer implementation, and can show a general trend of the drifts in the numerical time integration process. In a subsequent section, some numerical results are presented to corroborate the aforementioned theoretical predictions.
4.4. Influence of perturbations. In this section, we will study the propagation of perturbations over a system time-step. This analysis will help us better understand how perturbations in input (in this case previous time-level) will affect the solution at the next time-level. In the following theorem, we will consider application of the proposed method to non-linear DAEs of form

\[ M_i \dot{u}_i (t) = h_i \left( u_i (t), t \right) + C_i^T \lambda (t) \quad \forall i \]  

(4.32)

\[ \sum_{i=1}^{S} C_i u_i (t) = 0 \]  

(4.33)

Theorem 5. Let \( \left( v_i^{(n+j+1/\eta_i)}, d_i^{(n+j+1/\eta_i)}, \lambda^{(n+1)} \right) \) with \( j = 1, \ldots, \eta_i - 1 \) and \( i = 1, \ldots, S \) be the solution of the following system

\[ \frac{\delta d_i^{(n+1)}}{\delta t} = M_i^{-1} h_i \left( d_i^{(n+1/\eta_i)}, t^{(n+1/\eta_i)} \right) + M_i^{-1} C_i^T \lambda^{(n+1)} \]  

(4.34a)

\[ d_i^{(n+1/\eta_i)} = d_i^{(n+1/\eta_i)} + \Delta t \left( 1 - \phi \right) v_i^{(n+1/\eta_i)} + \Delta t \phi \delta d_i^{(n+1)} + \Delta t \epsilon_{d_i} \]  

(4.34b)

\[ \lambda^{(n+1/\eta_i)} = \left( 1 - \frac{j+1}{\eta_i} \right) \lambda^{(n)} + \left( \frac{j+1}{\eta_i} \right) \lambda^{(n+1)} + \Delta t \Delta \lambda \]  

(4.34c)

\[ \sum_{i=1}^{S} C_i d_i^{(n+1)} = \epsilon_\lambda \quad \text{or} \quad \sum_{i=1}^{S} C_i \left( v_i^{(n+1)} + \frac{\alpha}{\Delta t} \delta d_i^{(n+1)} \right) = \frac{1}{\Delta t} \epsilon_\lambda \]  

(4.34d)

(d-continuity) Baumgarte stabilization

in which we have assumed that \( \Delta \lambda = O(\Delta t), \epsilon_{d_i} = O(\Delta t_i), \) and \( \epsilon_\lambda = O(\Delta t^2) \). Furthermore,

\[ v_i^{(n)} - v_i^{(n)} = O(\Delta t_i), \quad d_i^{(n)} - d_i^{(n)} = O(\Delta t_i^2), \quad \lambda^{(n)} - \lambda^{(n)} = O(\Delta t) \]  

(4.35)

Let the functions \( M_i^{-1} h_i \) (\( i = 1, \ldots, S \)) be Lipschitz continuous, then the following inequalities will hold:

\[ \left\| \delta d_i^{(n+1)} \right\| \leq C_d \left( \sum_{i=1}^{S} \left( \left\| \delta d_i^{(n)} \right\| + \Delta t \left\| \epsilon_{d_i} \right\| \right) + \Delta t \left\| \delta \lambda^{(n)} \right\| + \phi \left\| \epsilon_\lambda \right\| + \Delta t^2 \left\| \Delta \lambda \right\| \right) \]  

(4.36a)

\[ \left\| \delta v_i^{(n+1)} \right\| \leq C_v \left( \sum_{i=1}^{S} \left( \frac{1}{\Delta t} \left\| \delta d_i^{(n)} \right\| + \left\| \epsilon_{d_i} \right\| \right) + \left\| \delta \lambda^{(n)} \right\| + \frac{\phi}{\Delta t} \left\| \epsilon_\lambda \right\| + \Delta t \left\| \Delta \lambda \right\| \right) \]  

(4.36b)

\[ \left\| \delta \lambda^{(n+1)} \right\| \leq C_\lambda \left( \sum_{i=1}^{S} \left( \frac{1}{\Delta t} \left\| \delta d_i^{(n)} \right\| + \left\| \epsilon_{d_i} \right\| \right) + \left\| \delta \lambda^{(n)} \right\| + \frac{\phi}{\Delta t} \left\| \epsilon_\lambda \right\| + \Delta t \left\| \Delta \lambda \right\| \right) \]  

(4.36c)

where \( C_d, C_v, C_\lambda \) are constants, \( \delta \square = \square - \square \), and

\[ \phi = \begin{cases} 1 & \text{d - continuity method} \\ \Delta t & \text{Baumgarte stabilization method} \end{cases} \]  

(4.37)

Proof. From equation (4.34) we can write:

\[ \delta v_i^{(n+1/\eta_i)} = M_i^{-1} \left( h_i \left( d_i^{(n+1/\eta_i)}, t^{(n+1/\eta_i)} \right) - h_i \left( d_i^{(n+1/\eta_i)}, t^{(n+1/\eta_i)} \right) \right) + M_i^{-1} C_i^T \left( 1 - \frac{j+1}{\eta_i} \right) \delta \lambda^{(n)} + \left( \frac{j+1}{\eta_i} \right) \delta \lambda^{(n+1)} + \Delta t \Delta \lambda \quad \forall i \]  

(4.38)
Lipschitz continuity of functions $M_i^{-1} h_i$ and $M_i^{-1} C_i^T$ can be used to obtain the following inequality:

$$\left\| M_i^{-1} h_i \left( d_i^{\left( n+\frac{j+1}{\eta_i}\right)}, t^{\left( n+\frac{j+1}{\eta_i}\right)} \right) - M_i^{-1} h_i \left( d_i^{\left( n+\frac{j}{\eta_i}\right)}, t^{\left( n+\frac{j}{\eta_i}\right)} \right) \right\| \leq L_i^h \left\| d_i^{\left( n+\frac{j}{\eta_i}\right)} \right\|$$

By taking norms on both sides in equation (4.38), and applying the triangle inequality, we obtain the following:

$$M_i^{-1} C_i^T \left( \lambda^{\left( n+\frac{j}{\eta_i}\right)} - \lambda^{\left( n+\frac{j}{\eta_i}\right)} \right) \right\| \leq L_i^\lambda \left\| \lambda^{\left( n+\frac{j}{\eta_i}\right)} \right\|$$

Equations (4.40) and (4.41) imply the following:

$$\left\| \delta v_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| \leq L_i^h \left\| \delta d_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| + \left( 1 - \frac{j}{\eta_i} \right) L_i^\lambda \left\| \delta \lambda^{(n)} \right\| + \Delta t L_i^\lambda \left\| \Delta \lambda \right\|$$

Note that $0 \leq (j + 1)/\eta_i \leq 1$ for all $j$. Using equation (4.34b) one can obtain the following inequality:

$$\left\| \delta d_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| \leq \left\| \delta d_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| + \Delta t_i \left( 1 - \vartheta_i \right) \left\| \delta v_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| + \Delta t_i \vartheta_i \left\| \delta v_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| + \Delta t_i \left\| \delta \lambda^{(n)} \right\|$$

Equations (4.40) and (4.41) imply the following:

$$(1 - \Delta t_i L_i^h) \left\| \delta d_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| \leq \left( 1 + \Delta t_i L_i^h \right) \left\| \delta d_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| + 2 \Delta t_i L_i^\lambda \left\| \delta \lambda^{(n)} \right\| + 2 \Delta t_i L_i^\lambda \left\| \delta \lambda^{(n+1)} \right\|$$

We shall assume that the subdomain time-steps $\Delta t_i$ are sufficiently small that $1 - \Delta t_i L_i^h > 0$ holds. Then, the propagation of perturbations over a subdomain time-step will satisfy the following inequality:

$$\left\| \delta d_i^{\left( n+\frac{j+1}{\eta_i}\right)} \right\| \leq \left( 1 + \frac{\Delta t_i L_i^h}{1 - \Delta t_i L_i^h} \right) \left\| \delta d_i^{\left( n+\frac{j}{\eta_i}\right)} \right\| + \frac{2 \Delta t_i L_i^\lambda}{1 - \Delta t_i L_i^h} \left\| \delta \lambda^{(n)} \right\| + \frac{2 \Delta t_i L_i^\lambda}{1 - \Delta t_i L_i^h} \left\| \delta \lambda^{(n+1)} \right\|$$

Applying the above equation in a recursive manner, the following inequality can be obtained over a system time-step:

$$\left\| \delta d_i^{(n+1)} \right\| \leq \left( 1 + \frac{\Delta t_i L_i^h}{1 - \Delta t_i L_i^h} \right)^{\eta_i} \left\| \delta d_i^{(n)} \right\| + \left\{ k \sum_{k=0}^{\eta_i-1} \left( 1 + \frac{\Delta t_i L_i^h}{1 - \Delta t_i L_i^h} \right)^{k} \right\} \left( \frac{2 \Delta t_i L_i^\lambda}{1 - \Delta t_i L_i^h} \right) \left\| \delta \lambda^{(n)} \right\|$$

$$+ \frac{2 \Delta t_i L_i^\lambda}{1 - \Delta t_i L_i^h} \left\| \delta \lambda^{(n+1)} \right\| + \frac{\Delta t_i}{1 - \Delta t_i L_i^h} \left\| \delta \lambda \right\| + \frac{\Delta t_i}{1 - \Delta t_i L_i^h} \left\| \epsilon_{di} \right\|$$
Similarly, one can derive the following inequality for the rate variables:

\[
\| \delta v_i^{(n+1)} \| \leq L_i^h \left( \frac{1 + \Delta t_i L_i^{f}}{1 - \Delta t_i L_i^{h}} \right)^{n_i} \| \delta d_i^{(n)} \| \\
+ \left\{ L_i^h \left( \sum_{k=0}^{n_i-1} \left( \frac{1 + \Delta t_i L_i^{h}}{1 - \Delta t_i L_i^{h}} \right)^k \right) \right\} \left( \frac{2 \Delta t_i L_i^{\lambda} + L_i^{\Delta \lambda}}{1 - \Delta t_i L_i^{\lambda}} + \Delta t L_i^{\lambda} \right) \| \delta \lambda^{(n)} \| + \| \delta \lambda^{(n+1)} \| \\
+ \left\{ L_i^h \left( \sum_{k=0}^{n_i-1} \left( \frac{1 + \Delta t_i L_i^{h}}{1 - \Delta t_i L_i^{h}} \right)^k \right) \right\} \frac{\Delta t_i}{1 - \Delta t_i L_i^{h}} \| \varepsilon_i \| \right\}
\]

(4.45)

From the perturbed constraint equations, the following inequalities holds for the \( d \)-continuity method:

\[
\| \varepsilon_{\lambda} \| = \left\| \sum_{i=1}^{S} C_i d_i^{(n+1)} \right\| \leq \sum_{i=1}^{S} \| d_i^{(n+1)} \| \quad (4.46)
\]

and the following inequality holds for the coupling method based on the Baumgarte stabilization:

\[
\| \varepsilon_{\lambda} \| = \left\| \sum_{i=1}^{S} C_i \left( v_i^{(n+1)} + \frac{\alpha}{\Delta t} d_i^{(n+1)} \right) \right\| \leq \sum_{i=1}^{S} \left( \| v_i^{(n+1)} \| + \frac{\alpha}{\Delta t} \| d_i^{(n+1)} \| \right) \quad (4.47)
\]

By substituting inequalities (4.44) and (4.45) in the above equation, one can obtain the desired inequality for \( \| \delta \lambda^{(n+1)} \| \). By substituting the resulting inequality in equations (4.44) and (4.45), one can obtain the desired inequalities for \( \| \delta d_i^{(n+1)} \| \) and \( \| \delta v_i^{(n+1)} \| \).

**Remark 6.** The difference in the order of the perturbation in the algebraic constraints in (4.34) arises due to the difference in the differential index of the governing DAEs. That is, the \( d \)-continuity method form a system of DAEs of index 2, whereas the coupling method based on the Baumgarte stabilization form a system of DAEs of index 1.

### 5. Benchmark Problems for Verification

In this section, several benchmark problems are solved to illustrate the accuracy of the proposed coupling methods, to verify numerically the theoretical predictions, and to check the computer implementation.

#### 5.1. Split degree-of-freedom problem

The governing equations of a coupled system of two subdomains, as shown in Figure 3, take the following form:

\[
m_1 \ddot{u}_1(t) + k_1 u_1(t) = f_1(t) + \lambda(t) \quad (5.1a)
\]

\[
m_2 \ddot{u}_2(t) + k_2 u_2(t) = f_2(t) - \lambda(t) \quad (5.1b)
\]

\[
u_1(t) = u_2(t) \quad (5.1c)
\]

where \( \lambda(t) \) is the Lagrange multiplier. The following parameters have been used in this numerical simulation:

\[
m_1 = 100, \ m_2 = 1, \ k_1 = 1, \ k_2 = 100, \ f_1 = f_2 = 0 \quad (5.2)
\]
We shall solve the DAEs given by equations (5.1a)–(5.1c) using the proposed multi-time-step coupling methods, subject to the initial condition \( u_1(t = 0) = u_2(t = 0) = 1 \).

5.1.1. Performance of the \( d \)-continuity method. Figure 4 shows the results of numerical solution to (5.1) using the proposed coupling method with \( d \)-continuity. Different system and subdomain time-steps have been chosen. Implicit Euler method (i.e., \( \vartheta_1 = 1 \)) is used to integrate the first subdomain, and the second subdomain is integrated using the midpoint rule (i.e., \( \vartheta_2 = 1/2 \)). As shown earlier, the proposed method is stable under \( d \)-continuity if \( \vartheta_i \geq 1/2 \) in all subdomains. Enforcing \( d \)-continuity, assures the continuity of primary variable (concentrations in this paper) along the interface at all system time-levels. The proposed methods shows good compatibility with the exact solution.

5.1.2. Performance of the Baumgarte stabilization. Baumgarte stabilization allows coupling explicit and implicit time-integrators in different subdomains. Midpoint rule is employed in the first subdomain (i.e., \( \vartheta_1 = 1/2 \)). In this problem explicit Euler method is used in the second subdomain (i.e., \( \vartheta_2 = 0 \)). As it can be seen in Figure 5, choice of system time-step \( \Delta t \), and Baumgarte stabilization parameter \( \alpha \), influence the accuracy of the numerical result. The drift in the primary variables, \( u_1 \) and \( u_2 \), is nonzero. One can observe in Figure 5 that increasing the Baumgarte stabilization parameter \( \alpha \), or decreasing the system time-step \( \Delta t \) can improve the accuracy.

5.2. One-dimensional problem. We will consider an unsteady diffusion with decay in one-dimension, which is an extension of the steady-state version considered in [Farrell et al., 1995]. The governing equations can be written as follows:

\[
\frac{\partial u}{\partial t} + u - \varepsilon \frac{\partial^2 u}{\partial x^2} = 1 \quad x \in (0, 1), \ t \in (0, T) \\
u(x = 0, t) = u(x = 1, t) = 0 \quad t \in (0, T) \\
u(x, t = 0) = 0 \quad x \in (0, 1)
\]

(5.3a)(5.3b)(5.3c)

It is well-known that the solution of this singularly perturbed problem will exhibit boundary layers for small values of \( \varepsilon \). Herein, we have taken \( \varepsilon = 0.01 \). We shall demonstrate the benefits of using the proposed multi-time-step coupling methods to problems in which the behavior of the solution can be very different in various regions of the computational domain.

The domain is decomposed into three subdomains, as shown in Figure 6. Subdomains 1 and 3 are the regions in which the boundary layers will appear. Note that these subdomains are meshed using much finer elements than subdomain 2. The obtained numerical results obtained using the proposed multi-time-step coupling methods are shown in Figures 7 and 8. Results are in good agreement with the exact solution, and the boundary layers are captured accurately by the proposed coupling methods. This numerical experiment illustrates the following attractive features of the proposed coupling methods:

(a) The system time-step can be much larger than subdomain time-steps.
(b) For fixed subdomain time-steps, smaller system time-step will result in better accuracy.
(c) Under the coupling method based on the Baumgarte stabilization and fixed subdomain time-steps, decreasing system time-step and/or increasing the Baumgarte stabilization parameter will result in improved accuracy.
(d) Utilizing smaller time-steps in individual subdomains improves the accuracy of results in the respective subdomain.
5.3. Two-dimensional problem. A transient version of the well-known problem proposed by Hemker [Hemker, 1996] will be considered. The governing equations take the following form:

\[
\begin{align*}
\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} - \varepsilon \text{div} [\nabla u] &= 0 \quad \text{in } \Omega \quad (5.4a) \\
u(x,y,t) &= 1 \quad \text{on } \Gamma_D^1 \quad (5.4b) \\
u(x,y,t) &= 0 \quad \text{on } \Gamma_D^2 \quad (5.4c) \\
-\varepsilon \nabla u \cdot \vec{n}(x) &= 0 \quad \text{on } \Gamma_N \quad (5.4d) \\
u(x,y,t=0) &= 0 \quad \text{in } \Omega \quad (5.4e)
\end{align*}
\]

where \( \varepsilon \in (0,1] \), \( \Omega = \Phi \setminus \{(x,y) \mid (x,y) \in \Phi, \ x^2 + y^2 \leq 1 \} \), and \( \Gamma_D^1 = \{(x,y) \mid (x,y) \in \Phi, \ x^2 + y^2 = 1 \} \), \( \Gamma_D^2 = \{(x,y) \mid (x,y) \in \Phi, \ x = -2 \} \), \( \Gamma_N = \partial \Omega \setminus \Gamma_D^1 \cup \Gamma_D^2 \), in which \( \Phi = [-2,12] \times [-4,4] \). Computational domain, mesh, and domain decomposition are shown in Figures 9 and 10. In this problem, the advection velocity is \( \mathbf{v} = (1,0) \), and \( \varepsilon = 0.01 \). The problem at hand is a singularly perturbed equation and is known to exhibit both boundary and interior layers. Furthermore, the standard Galerkin formulation is known to produce spurious oscillations for small values of \( \varepsilon \) [Gresho and Sani, 2000].

The numerical results obtained using the Galerkin weak formulation are shown in Figure 11. As expected, spurious oscillations occur at the vicinity of the circle. The minimum value observed in both cases is -0.439. The spurious oscillations and the violation of the non-negative constraint is because of using the Galerkin weak formulation, and is not due to the use of proposed multi-time-step coupling methods. To corroborate this claim, Figure 12 shows the results where tailored weak formulations are employed in different subdomains. The GLS formulation is used in subdomain 1, the SUPG formulation is employed in subdomain 2, and the Galerkin formulation in subdomain 3. There are no spurious oscillations and the minimum value observed is -0.062. This example demonstrates choice of disparate time-steps, and different numerical formulations in different spatial regions of the computational domain.

6. MULTI-TIME-STEP TRANSIENT ANALYSIS OF A TRANSPORT-CONTROLLED BIMOLECULAR REACTION

In this section, we shall apply the proposed multi-time-step coupling methods to a transport-controlled bimolecular reaction. This problem is of tremendous practical importance in areas such as transverse mixing-limited chemical reactions in groundwater and aquifers, and mixing-controlled bioreactive transport in heterogeneous porous media arising in bioremediation. We shall now document the most important equations of the mathematical model. A more detailed discussion about the model can be found in Nakshatrala et al. [2013], which however did not address multi-time-step coupling methods.

6.1. Mathematical model. Consider the following irreversible chemical reaction:

\[
n_A A + n_B B \rightarrow n_C C \quad (6.1)
\]

where \( A, B \) and \( C \) are the chemical species participating in the reaction, and \( n_A, n_B \) and \( n_C \) are their respective (positive) stoichiometry coefficients. The fate of the reactants and the product are governed by coupled system of transient advection-diffusion-reaction equations. We shall assume the part of the boundary on which the Dirichlet boundary condition is enforced to be the same for the reactants and the product. Likewise is assumed for the Neumann boundary conditions.
can then find two invariants that are unaffected by the underlying reaction, which can be obtained via the following linear transformation:

\[ c_F := c_A + \left( \frac{n_A}{n_C} \right) c_C \]  
\[ c_G := c_B + \left( \frac{n_B}{n_C} \right) c_C \]  

The evolution of these invariants is given by the following uncoupled transient advection-diffusion equations:

\[ \frac{\partial c_i}{\partial t} + \text{div} [ \mathbf{v} c_i - \mathbf{D}(x) \text{grad} [c_i]] = f_i(x, t) \quad \text{in } \Omega \times \mathcal{I} \]  
\[ c_i(x, t) = c_i^0(x, t) + \left( \frac{n_j}{n_C} \right) c_j^0(x, t) \quad \text{on } \Gamma^D \times \mathcal{I} \]  
\[ -\mathbf{n}(x) \cdot \mathbf{D}(x) \text{grad} [c_i] = h_i^0(x, t) := h_i^0(x, t) + \left( \frac{n_j}{n_C} \right) h_j^0(x, t) \quad \text{on } \Gamma^N \times \mathcal{I} \]  
\[ c_i(x, t = 0) = c_i^0(x) := c_i^0(x) + \left( \frac{n_j}{n_C} \right) c_j^0(x) \quad \text{in } \Omega \]

where \( i = F \) or \( G \). We shall restrict to fast bimolecular reactions. That is, the time-scale of the chemical reaction is much smaller than the time-scale of the transport processes. For such situations, one can assume that the chemical species \( A \) and \( B \) cannot coexist at a spatial point and for a given instance of time. This implies that the concentrations of the reactants and the product can be obtained from the concentrations of the invariants through algebraic manipulations. To wit,

\[ c_A(x, t) = \max \left\{ c_F(x, t) - \left( \frac{n_A}{n_B} \right) c_G(x, t), 0 \right\} \]  
\[ c_B(x, t) = \left( \frac{n_B}{n_A} \right) \max \left\{ -c_F(x, t) + \left( \frac{n_A}{n_B} \right) c_G(x, t), 0 \right\} \]  
\[ c_C(x, t) = \left( \frac{n_C}{n_A} \right) (c_F(x, t) - c_A(x, t)) \]

Note that the solution procedure is still nonlinear, as the \( \max \{ \cdot, \cdot \} \) operator is nonlinear.

We shall employ the proposed multi-time-step computational framework to solve equations (6.3a)–(6.3d) to obtain concentrations of the invariants. Using the calculated values, we then find the concentrations for the reactants and the product using equations (6.4a)–(6.4c). The Galerkin formulation is employed all the subdomains. The negative values for the concentration are clipped at every subdomain time-step in the numerical simulations.

**6.2. Numerical results for a diffusion-controlled bimolecular reaction.** Consider a reaction chamber with \( L_x = L_y = 1 \), as shown in Figure 13. The computational domain is meshed using 5442 four-node quadrilateral elements. As shown in this figure, the domain is decomposed into four non-contiguous subdomains using METIS [Karypis and Kumar, 1999]. The diffusivity tensor is taken as follows:

\[ \mathbf{D}(x, y) = \begin{bmatrix} \gamma x^2 + y^2 & -(1 - \gamma) xy \\ -(1 - \gamma) xy & x^2 + \gamma y^2 \end{bmatrix} \]  

where \( \gamma = 0.001 \). Baumgarte stabilization is employed to enforce compatibility along the subdomain interfaces with \( \alpha = 100 \). Implicit Euler method is employed in subdomains 1 and 3, and midpoint
rule is employed in subdomains 2 and 4. The system time-step is taken as $\Delta t = 10^{-3}$, and the subdomain time-steps are taken as $\Delta t_1 = \Delta t_3 = 5 \times 10^{-4}$, and $\Delta t_2 = \Delta t_4 = 10^{-3}$.

Numerical results for the concentrations of the invariants, reactants and product are shown in Figures 14 and 15. As discussed earlier, Baumgarte stabilized coupling method can result in drift in the primary variable but it can be controlled using the stabilization parameter $\alpha$. Equation (4.31) can serve as a valuable tool assessing the overall behavior of drifts with respect to system time-step, and the Baumgarte stabilization parameter $\alpha$. Drifts for several choices of $\alpha$ and $\Delta t$ are shown in Figure 16. Note that equation (4.31) assumes no subcycling, and no mixed time-integration.

6.3. Numerical results for a fast bimolecular reaction with advection. Consider a reaction chamber with $L_x = 4$ and $L_y = 1$, as shown in Figure 17(a). The computational domain is meshed using three-node triangular elements, and is decomposed into four non-contiguous subdomains, as shown in Figure 17(b). There are 4151 interface constraints to ensure continuity of concentration along the subdomain interface. The diffusivity tensor is taken as follows:

$$D(x) = \alpha_T \|v\|I + \frac{\alpha_L - \alpha_T}{\|v\|} v \otimes v$$

where $I$ is the second-order identity tensor, $\otimes$ is the tensor product, $\| \cdot \|$ is the 2-norm, $v(x)$ is the velocity, and $\alpha_L$ and $\alpha_T$ are, respectively, the longitudinal and transverse diffusivities. This form of diffusivity tensor is commonly employed in subsurface hydrology [Pinder and Celia, 2006]. We define the velocity through the following stream function:

$$\psi(x, y) = -y - \sum_{k=1}^{3} A_k \cos \left( \frac{p_k \pi x}{L_x} - \frac{\pi}{2} \right) \sin \left( \frac{q_k \pi y}{L_y} \right)$$

The components of the advection velocity can then be calculated as follows:

$$v_x(x, y) = -\frac{\partial \psi}{\partial y}, \quad v_y(x, y) = +\frac{\partial \psi}{\partial x}$$

The following parameters are used in the numerical simulation:

$$p_1 = 4, \quad p_2 = 5, \quad p_3 = 10, \quad q_1 = 1, \quad q_2 = 5, \quad q_3 = 10, \quad A_1 = 0.08, \quad A_2 = 0.02, \quad A_3 = 0.01$$

The diffusivities are taken as $\alpha_L = 1$ and $\alpha_T = 10^{-4}$, and the prescribed concentrations on the boundary are taken as $c_A^p = 1.0$ and $c_B^p = 1.5$. The numerical results for the concentration of the product at various time levels obtained using the $d$-continuity coupling method are shown in Figure 18. In all the numerical results, there is no noticeable drift along the subdomain interface, and the proposed multi-time-step computational framework performed well.

7. CONCLUDING REMARKS

We presented a stable multi-time-step computational framework for transient advective-diffusive-reactive systems. The computational domain can be divided into an arbitrary number of subdomains. Different time-stepping schemes under the trapezoidal family can be used in different subdomains. Different time-steps and different numerical formulations can be employed in different subdomains. Unlike many of the prior works on multi-time-step methods (e.g., staggered schemes proposed in [Piperno, 1997]), no preferential treatment is given to the subdomain with the largest subdomain time-step, and thereby eliminating the associated subdomain-dependent solutions.

Under the framework, we proposed two different monolithic coupling methods, which differ in the way compatibility conditions are enforced along the subdomain interface. Under the first
method (i.e., d-continuity method), the continuity of the primary variable is enforced along the subdomain interface at every system time-step. An attractive feature of the d-continuity method is that, by construction, there is no drift in the primary variable along the subdomain interface. However, one cannot couple explicit and implicit schemes under the d-continuity method. But this method has good stability characteristics. The second method is based on an extension of the classical Baumgarte stabilization \cite{Baumgarte1972, Nakshatrala2009} to first-order transient systems. Under this method one can couple explicit and implicit schemes. However, there can be drift in the primary variable along the subdomain interface. But this drift is bounded and small, which we have shown both theoretically and numerically. The other salient features of the proposed coupling methods are as follows. There is no limitation on the number of subdomains or on the subcycling ratios $\eta_i$. Since no preference is given to any subdomain, the numerical solutions under the proposed coupling methods will not be affected by the way the computational domain is decomposed into subdomains. This is also evident from the numerical results presented in this paper. The coupling methods are shown to be stable, which has been illustrated both mathematically and numerically.

Based on the above discussion, we shall make the following two recommendations for a multi-time-step analysis of first-order transient systems:

(i) If it is not needed to couple explicit/implicit but just want to use different time-steps and different numerical formulations in different regions then it is recommended to use the proposed d-continuity method. If one want to employ couple explicit and implicit schemes, then one has to use the proposed coupling method based on Baumgarte stabilization.

(ii) Accuracy can be improved by decreasing the system time-step.

A possible research work can be towards the implementation of the proposed multi-time-step coupling methods in a parallel computing environment and on graphical processing units (GPUs).

**APPENDIX: A COMPACT NOTATION FOR COMPUTER IMPLEMENTATION**

Herein, we present a compact matrix form for the proposed coupling methods. In fact, we present in a more general setting by considering nonlinear first-order transient DAEs of the following form:

$$M_i \dot{u}_i(t) = h_i(u_i(t), t) + C_i^T \lambda(t) \quad \forall i$$  \hspace{1cm} (7.1)

$$\sum_{i=1}^{S} C_i u_i(t) = 0$$  \hspace{1cm} (7.2)

We will employ a Newton-Raphson-based approach to solve the given system of equation. Other methods of solving nonlinear equations (e.g., Picard method) can also be utilized. However for simplicity of the presentation, we shall ignore further details on solution techniques for solving nonlinear equations.

The resulting system of equations will have to be solved in iterations until some suitable convergence condition is met. Let us denote the differentiation operator with respect to $u$ by $D$ (i.e., $D := \partial/\partial u$). Let $\boxplus_{i}^{(n+j+1)/\eta_i}$ denote the nodal values of $\boxplus$ in the $i$-th subdomain, at time-level
\(n + \frac{i+1}{\eta_i}\), and after \(\nu\) iterations. The following notation will be useful:

\[
\begin{align*}
    h_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} &= h_{i,\nu} \left( d_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} \right) \\
    Dh_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} &= Dh_{i} \left( d_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} \right)
\end{align*}
\]  \(7.3\)
\(7.4\)

The unknowns at all subdomain time-levels for the \(i\)-th subdomain and for a given Newton-Raphson iteration number \(\nu\) can be grouped as follows:

\[
\begin{align*}
    X_{i,\nu+1}^{\left(n + \frac{1}{\eta_i}\right)} &= \begin{bmatrix} v_{i,\nu+1}^{\left(n + \frac{1}{\eta_i}\right)} \\ d_{i,\nu+1}^{\left(n + \frac{1}{\eta_i}\right)} \end{bmatrix},
    X_{i,\nu+1}^{(n+1)} &= \begin{bmatrix} X_{i,\nu+1}^{\left(n + \frac{1}{\eta_i}\right)} \\ X_{i,\nu+1}^{\left(n + \frac{2}{\eta_i}\right)} \\ \vdots \\ X_{i,\nu+1}^{(n+1)} \end{bmatrix},
    X_{i,\nu+1}^{(n+1)} &= \begin{bmatrix} \varphi_{i,\nu+1}^{(n+1)} \\ \varphi_{i,\nu+1}^{(n+1)} \\ \varphi_{i,\nu+1}^{(n+1)} \\ \vdots \\ \varphi_{i,\nu+1}^{(n+1)} \end{bmatrix}
\end{align*}
\]  \(7.5\)

Using equation \(7.3\) and the trapezoidal time-stepping schemes, the following linearized matrices can be constructed:

\[
\begin{align*}
    L_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} &= \begin{bmatrix} M_i & -Dh_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} \\ -\vartheta_i \Delta t_i I_i & I_i \end{bmatrix}, \\
    \mathbb{R}_i &= \begin{bmatrix} \mathcal{O}_i & \mathcal{O}_i \\ (1 - \vartheta_i) \Delta t_i I_i & I_i \end{bmatrix}, \\
    \mathbb{C}_i &= [C_i, O_i]
\end{align*}
\]  \(7.6\)

where \(\mathcal{O}_i\) and \(I_i\) are, respectively, zero matrix and identity matrix of size \(N_i \times N_i\), and the matrix \(O_i\) is a zero matrix of size \(N \times N_i\). The forcing function and the results from the previous iteration can be compactly assumed into the following vector:

\[
\begin{align*}
    \mathbf{F}_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} &= h_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} - Dh_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} \mathbf{d}_{i,\nu}^{\left(n + \frac{i+1}{\eta_i}\right)} \mathbf{0}
\end{align*}
\]  \(7.7\)

Now, let the square matrices \(A_{i,\nu}\) and \(A_{\nu}\), and column vectors \(\mathbf{F}_{i,\nu}^{(n+1)}\) and \(\mathbf{F}_{\nu}^{(n+1)}\) be defined as below:

\[
\begin{align*}
    A_{i,\nu} &= \begin{bmatrix} L_{i,\nu}^{\left(n + \frac{1}{\eta_i}\right)} & \mathbb{R}_i & L_{i,\nu}^{\left(n + \frac{2}{\eta_i}\right)} & \mathbb{R}_i & \cdots & \mathbb{R}_i & L_{i,\nu}^{(n+1)} \\ -\mathbb{R}_i & L_{i,\nu}^{\left(n + \frac{1}{\eta_i}\right)} & -\mathbb{R}_i & \mathbb{R}_i & \cdots & \mathbb{R}_i & -\mathbb{R}_i \end{bmatrix}_{2\eta_i N_i}, \\
    A_{\nu} &= \begin{bmatrix} A_{1,\nu} \\ A_{2,\nu} \\ \vdots \\ A_{\nu} \end{bmatrix}
\end{align*}
\]  \(7.8\)

\[
\begin{align*}
    \mathbf{F}_{i,\nu}^{(n+1)} &= \begin{bmatrix} \mathbf{F}_{i,\nu}^{\left(n + \frac{1}{\eta_i}\right)} + \mathbb{R}_i \mathbf{X}_{i}^{(n)} + C_i^T \lambda^{(n)} \\ \mathbf{F}_{i,\nu}^{\left(n + \frac{2}{\eta_i}\right)} + C_i^T \lambda^{(n)} \\ \vdots \\ \mathbf{F}_{i,\nu}^{(n+1)} + C_i^T \lambda^{(n)} \end{bmatrix}, \\
    \mathbf{F}_{\nu}^{(n+1)} &= \begin{bmatrix} \mathbf{F}_{1,\nu}^{(n+1)} \\ \mathbf{F}_{2,\nu}^{(n+1)} \\ \vdots \\ \mathbf{F}_{\nu}^{(n+1)} \end{bmatrix}
\end{align*}
\]  \(7.9\)
Enforcing the algebraic constraints can be done using the following matrices:

\[
\mathbb{B}_i = \begin{bmatrix}
O_i & O_i & \cdots & O_i & C_i \\
\end{bmatrix}_{2\eta_i N_i}
\]

for \(d\)-continuity method \( (7.10) \)

\[
\mathbb{B}_i = \begin{bmatrix}
O_i & O_i & \cdots & C_i & -\frac{\alpha}{\Delta t} C_i \\
\end{bmatrix}_{2\eta_i N_i}
\]

for Baumgarte stabilization method \( (7.11) \)

Using the notation above, one can then construct the following augmented matrix:

\[
\mathbb{B} = \begin{bmatrix} \mathbb{B}_1 & \mathbb{B}_2 & \cdots & \mathbb{B}_S \end{bmatrix}
\]

(7.12)

The algebraic constraint (for both \(d\)-continuity and Baumgarte stabilization) can be compactly written as follows:

\[
\mathbb{B} X^{(n+1)} = 0
\]

(7.13)

We define the matrix \(\mathbb{C}\) as follows:

\[
\mathbb{C} = \begin{bmatrix}
-\frac{1}{\eta_1} C_1 & -\frac{2}{\eta_1} C_1 & \cdots & -\frac{1}{\eta_S} C_S & -\frac{2}{\eta_S} C_S & \cdots & -\frac{1}{\eta_S} C_S \\
\end{bmatrix}
\]

\(\sum_{i=1}^{S} 2\eta_i N_i\)

(7.14)

Finally, time marching can be performed by solving the following equation:

\[
\begin{bmatrix}
\tilde{A}_{\nu} & \mathbb{C}^T \\
\mathbb{B} & \mathbb{0}
\end{bmatrix}
\begin{bmatrix}
X^{(n+1)}_{\nu+1} \\
\lambda^{(n+1)}_{\nu+1} - \lambda^{(n)}
\end{bmatrix}
= \begin{bmatrix}
\mathbb{B}^T X^{(n+1)} \\
\mathbb{0}
\end{bmatrix}
\]

(7.15)

which gives the values of the nodal concentrations and the corresponding rates within a Newton-Raphson iteration for all subdomains and at all subdomain time-levels within a system time-step.

ACKNOWLEDGMENTS

The authors acknowledge the support of the National Science Foundation under Grant no. CMMI 1068181. The opinions expressed in this paper are those of the authors and do not necessarily reflect that of the sponsors.

References


Figure 1. A pictorial description of computational domain and its decomposition into subdomains, subdomain interface, and interface forces (i.e., Lagrange multipliers).
Figure 2. A pictorial description of time levels ($t_n$), system time-step ($\Delta t$), subdomain time-step ($\Delta t_i$), and subcycling. By subcycling in the $i$-th subdomain we mean that $\Delta t_i < \Delta t$.

Figure 3. Split degree-of-freedom problem: A pictorial description.
Figure 4. Split degree-of-freedom problem: We have employed the multi-time-step coupling method based on $d$-continuity method. The values of concentrations, Lagrange multipliers, and drifts are compared with their respective exact solutions. Subdomain 1 is integrated using the implicit Euler method, and subdomain 2 is integrated using the midpoint rule. The initial concentration is taken to be 1. Different subcycling ratios have been employed. It can be seen that the numerical results obtained using the proposed coupling method based on $d$-continuity method show very good compatibility with the exact solution.
Figure 5. Split degree-of-freedom problem: The values of concentrations, Lagrange multipliers, and drifts are compared with their respective exact solutions. In this problem Baumgarte stabilization is used. Midpoint rule is employed in subdomain 1, and explicit Euler scheme is employed in subdomain 2. The initial concentrations are 1. The subdomain time-steps are \( \Delta t_1 = 0.1 \) and \( \Delta t_2 = 0.02 \) in all cases. As it can be observed, the accuracy can be improved by decreasing the system time-step and the Baumgarte stabilization parameter \( \alpha \). Note that there is no significant drift in the numerical solutions.
Figure 6. One-dimensional problem: The computational domain is divided into three subdomains of lengths $L_1 = 0.1$, $L_2 = 0.8$, and $L_3 = 0.1$. Two-node linear finite elements are used in all the subdomains. The source is unity in the entire domain (i.e., $f(x,t) = 1$). It should be noted that there will be boundary layers for the chosen parameters. In order to adequately capture these boundary layers, we shall employ very fine meshes in subdomains one and three.

Figure 7. One-dimensional problem: This figure compares the numerical solution obtained using the proposed $d$-continuity method is compared with the exact solution. Each subdomain is meshed using 100 two-node finite elements. Two different choices are made for subdomain time-steps, as indicated in the figure. In both the cases, the system time-step is taken as $\Delta t = 0.25$, and $\vartheta_1 = \vartheta_3 = 1/2$ and $\vartheta_2 = 1$. It should be noted that the time-stepping schemes chosen are implicit in all the subdomains, as it is not possible to have explicit/implicit coupling under the $d$-continuity coupling method. The stability result given in Theorem 2 requires that $\vartheta_i \geq 1/2 \forall i$. 
Figure 8. One-dimensional problem: The numerical solution using the proposed coupling method with Baumgarte stabilization is shown in this figure. As it was shown in theorem 4, when conditionally stable trapezoidal schemes are used, multi-time-stepping can expand the acceptable values of $\alpha$ without compromising the stability of the coupling method. In figures (c) and (d), explicit Euler method is used in subdomains 1 and 3 ($\vartheta_1 = \vartheta_3 = 0$). Implicit Euler method is employed in the second subdomain. Note that the system time-step is considerably larger than the subdomain time-steps $\Delta t_1$ and $\Delta t_3$. Under the proposed coupling method, choosing system time-step larger than the critical time-step of subdomains, does not cause instability. However, reducing the system time-step, or opting for a larger Baumgarte stabilization parameter $\alpha$, improves the overall accuracy.
Figure 9. Two-dimensional transient Hemker problem: The dimensions of the computational domain are taken as $L_x = 14$ and $L_y = 8$. A circular hole is centered at the origin, and has a radius of unity. Concentration is unity on the circumference of the circle, and zero along the left side of the domain. No-flux boundary condition is enforced on the rest of the boundary. The prescribed initial condition is zero.
Figure 10. Two-dimensional transient Hemker problem: This figure shows the computational mesh, and the decomposition of the domain into subdomains. The computational domain is meshed with 11512 triangular finite elements using GMSH [Geuzaine and Remacle, 2009], and is partitioned into three subdomains. The first subdomain is indicated in blue color, the second subdomain in green color, and the third subdomain is in red color. (See the online version of the paper for a color picture.)
Figure 11. Two-dimensional transient Hemker problem: The value of concentrations is shown on the domain of interest at $t = 5$. In this particular example, Galerkin weak formulation is employed. The advection velocity field is $\mathbf{v} = (1, 0)$, and $\varepsilon = 0.01$. In figure (a), $d$-continuity method is employed to enforce continuity at the subdomain interface. The computational domain is partitioned into three subdomains. The system time-step is $\Delta t = 0.1$, and subdomain time-steps are: $\Delta t_1 = 0.001$, $\Delta t_2 = 0.01$, and $\Delta t_3 = 0.1$. The time-integrators are: $\vartheta_1 = 1/2$, $\vartheta_2 = 1$, and $\vartheta_3 = 1$. Figure (b) shows the results when Baumgarte stabilization is employed to enforce continuity at the interface. The system time-step is $\Delta t = 0.2$, and the subdomain time-steps are: $\Delta t_1 = 0.01$, $\Delta t_2 = 0.05$, and $\Delta t_3 = 0.02$. Midpoint rule ($\vartheta_1 = 1/2$) is used in subdomain one, implicit Euler method is used in subdomain two ($\vartheta_2 = 1$), and explicit Euler method in subdomain three ($\vartheta_3 = 0$). The stabilization parameter is taken as $\alpha = 1$. Spurious oscillations due to semi-discrete Galerkin method can be seen in near the circle. The minimum value of concentrations seen in these examples is -0.439, which is significant compared to the maximum, which is unity.
Figure 12. Two-dimensional transient Hemker problem: Concentrations at $t = 5$ are shown. GLS formulation is used in subdomain 1, SUPG formulation is used in subdomain 2, and the standard Galerkin formulation is used in subdomain 3. The system time-step is taken as 0.2 in both cases. In figure (a), subdomain 1 is integrated using midpoint rule ($\vartheta_1 = 1/2$) with a time-step of $\Delta t_1 = 10^{-3}$. Subdomains 2 and 3 are integrated using implicit Euler method ($\vartheta_2 = \vartheta_3 = 1$), with time-steps of $\Delta t_2 = 5 \times 10^{-3}$, and $\Delta t_3 = 0.2$. Figure (b) shows the results using Baumgarte stabilization. The Baumgarte stabilization parameter is $\alpha = 1$, subdomain 1 is integrated using implicit Euler method ($\vartheta_1 = 1$) and a time-step of $\Delta t_1 = 10^{-3}$. Subdomain 2 is integrated using midpoint rule ($\vartheta_2 = 1/2$), and $\Delta t_2 = 5 \times 10^{-3}$. Explicit Euler method is used in subdomain 3 ($\vartheta_3 = 0$) and $\Delta t_3 = 2 \times 10^{-2}$. The minimum value for concentrations is -0.062 in both cases.
Figure 13. Fast bimolecular reaction problem without advection: The initial condition for the concentrations of all reactants is taken to be zero. The computational domain is meshed using 5442 four-node quadrilateral elements, and is divided into four subdomains using METIS [Karypis and Kumar, 1999]. Subdomain one is indicated in blue color, subdomain two is indicated in green region, subdomain three is in yellow color, and subdomain four is in red color. (See the online version of the paper for a color picture.)
Figure 14. Fast bimolecular reaction problem without advection: This figure shows the concentrations of the invariants $F$ and $G$ at $t = 0.01$ and $t = 0.1$. 
Figure 15. Fast bimolecular reaction problem without advection: Concentrations of the reactants and the product are shown at $t = 0.01$ and $t = 0.1$. 

(a) Concentration of reactant $A$ at $t = 0.01$

(b) Concentration of reactant $A$ at $t = 0.1$

(c) Concentration of reactant $B$ at $t = 0.01$

(d) Concentration of reactant $B$ at $t = 0.1$

(e) Concentration of product $C$ at $t = 0.01$

(f) Concentration of product $C$ at $t = 0.1$
Figure 16. Fast bimolecular reaction problem without advection: The $\infty$-norm of drift in concentrations of the chemical species $C$ is shown. There are 10047 interface constraint equations in this problem. As evident from this figure, the drift along the subdomain interface can be decreased by either decreasing the system time-step or by increasing the Baumgarte stabilization parameter $\alpha$. It can be seen that, the overall advances in the value of drifts follows the suggested equations (4.31). Note that reducing system time-step has resulted in much smaller drift compared to the values shown in figure (a). However, presence of subcycling will cause deviations from the trend suggested by equation (4.31), which assumes no subcycling and no mixed time-integrators.
(a) A pictorial description of the problem.

(b) Decomposition of the computational domain into subdomains.

**Figure 17.** Fast bimolecular reaction with advection: Chemical species $A$ and $B$ pumped into the reaction chamber from the left side and produce the product $C$ as a result of the chemical reaction. The computational domain is divided into four subdomains, which are indicated using different colors. Subdomains 1, 2, 3 and 4 are, respectively, indicated in blue, green, yellow and red colors. The computational domain is meshed using 4148 three-node triangular elements. The decomposition of the computational domain is done using the METIS software package [Karypis and Kumar, 1999]. Note that each subdomain consists of many non-contiguous parts, and is highly unstructured. (See the online version of the paper for a color picture.)
Figure 18. Fast bimolecular reaction with advection: This figure shows the concentration of the product $C$ at various instances of time obtained using the proposed $d$-continuity multi-time-step coupling method. The system time-step is taken to be $\Delta t = 0.1$, and the subdomain time-steps are $\Delta t_1 = 0.01$, $\Delta t_2 = 0.05$, $\Delta t_3 = 0.01$ and $\Delta t_4 = 0.05$. Implicit Euler method is employed in subdomains 1 and 3, and the midpoint rule is employed in subdomains 2 and 4. As one can see from the figure, there is no drift along the subdomain interface, and the proposed coupling method performed well.