Time Integration of Reacting Flows and Energy-Stable Hybrid Spatial Discretizations

Cory Mikida
Department of Aerospace Engineering
University of Illinois at Urbana-Champaign

January 22, 2021

Contents

1 Time Integration of Reacting Flows

1.1 Motivation & Background

In the simulation of reacting flows, the compressible Navier-Stokes equations are augmented to include equations governing the rate of change of the mass fraction of each species in the chemical mechanism at hand based on momentum, mass diffusion, and reaction kinetics. What results is a set of equations that can feature both a wide range of timescales and a number of comparatively stiff terms that can have an adverse effect on maximum allowable timestep size and, as a result, simulation performance and fidelity. In particular, the timescales of the chemical reactions are often multiple orders of magnitude lower than those of the fluid motion in practice. One approach to effectively time-marching these problems at the fluid-relevant timescale, then, is to introduce an operator-splitting technique ([38], [40], [24], [27], [37], [21]) in which the advection and diffusion terms in the right-hand-sides are advanced explicitly, with the chemical source term(s) being advanced implicitly and in a separate substep, typically by software packages such as CVODE [7]. However, this approach results in a scheme that is at best first- or second- order accurate in time globally.

To more accurately handle such situations, implicit-explicit (IMEX) time integration schemes have been developed in order to partition the governing equations and treat the stiff terms implicitly while treating the nonstiff terms explicitly. In 2003, Kennedy and Carpenter [19] introduced additive Runge-Kutta schemes for convection-diffusion-reaction equations, demonstrating a higher order of temporal accuracy than the operator splitting approach is capable of attaining. In 2011, Svärd and Mishra [42] also introduced more simplistic one-step IMEX schemes mimicking single stages in these additive-RK schemes, applying them to the reactive Euler equations and demonstrating first order with a need for numerical diffusion as the problem becomes less stiff.

Another approach to handling stiffness, and the resulting disparity in timescales between different simulation components, is the use of multi-rate time integration. Gear and Wells [15] were the initial authors of work on multi-step multi-rate methods, with similar contributions provided more recently by Constantinescu and Sandu [8] for hyperbolic conservation laws. Constantinescu and Sandu later extended this work to develop implicit multi-rate schemes [9], [10], but failed to introduce a scheme featuring both implicitly and explicitly advanced components within a multi-rate framework.

1.1.1 Goals, Bounds, and Impact

The primary goal of this section of the thesis is demonstration of a novel time integration scheme driving a canonical or especially demonstrative reacting flow problem (ideally one that points directly to real-world impact) with improved efficacy identified by either superior accuracy and stability or faster performance relative to existing methods. The integration project being undertaken aims to augment the capabilities of the multi-rate integrators developed in [32], adding the ability to treat certain right-hand side components implicitly as well as adapt either the macro-timestep or the step ratio of the scheme

based on local truncation error estimates. These additions are explicitly designed to handle the stiffness introduced to the flow problems to which these integrators were originally applied via the introduction of chemical source terms, as well as to efficiently integrate reacting flow simulations in which the stiffness varies. This latter phenomenon is especially common in ignition simulations, where the high-stiffness qualities of the kinetics during ignition contrast with surrounding simulation time in which the mass fractions are at or near chemical equilibrium.

Perhaps the next most prominent goal is a thorough investigation of the design space presented by the multi-rate framework we will use for time integration. The design choices inherent to this scheme include:

- Which right-hand side components in the governing equations should be treated as fast-evolving components, and which should be treated as slow-evolving components
- Evaluation order ("fastest-first" vs. "slowest-first")
- Re-extrapolation
- Inclusion of additional history beyond order requirements (shown in [32] to provide real-axis stability improvement)
- Which solution components to include in error estimation for adaptive timestep control
- Whether error control is accomplished through timestep control or step ratio control

Another critical component would be comparison of our scheme with CVODE as a proxy for the state of the art in time integration of chemically reacting flows, ideally achieving superiority in terms of observed temporal accuracy of the entire system, or in terms of performance (most accurately measurable by a reduction in the number of chemistry right-hand-side evaluations required to reach a given solution time).

Meanwhile, an aspect of the algorithms being implemented that has been identified early as beyond our bounds is the presence of multi-variable solves in a number of potential adaptive implicit integration scenarios, namely:

- Coupled implicit right-hand-sides
- Use of adaptive timestep controllers where the local error estimate of all solution components makes use of implicit and explicit state estimates.

With these goals and bounds in mind, the proposed thesis has the potential to have a significant impact on the simulation of reacting flows, of which there are countless real-world applications including (but not limited to) subsonic and supersonic combustion for propulsion (gas turbines, ramjets/scramjets, rockets), furnaces and residential heating, and solution of problems relating to atmospheric sciences.

1.2 Governing Equations

The problem at hand involves the simulation of chemically reacting flows, the governing equations of which amount to the Navier-Stokes equations with (in our form):

- A source term added to the energy equation in the form of heat flux
- The addition of a governing equation for the rate of change of the mass fractions of each species in the chemical mechanism simulated
- The addition of governing equations specifying the rate of change of temperature and pressure, derived from an assumption of constant internal energy and volume in a reaction substep (for the former) and the ideal gas law (for the latter).

These equations are given below, along with a brief explanation of the notation/nomenclature used.

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) = 0 \tag{1}$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j + P\delta_{ij} - \tau_{ij}) = 0 \tag{2}$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x_j}((\rho E + P)u_j + q_j - u_i\tau_{ij}) = 0 \tag{3}$$

$$\frac{\partial}{\partial t}(\rho\, Y_k) + \frac{\partial}{\partial x_j}(\rho\, Y_k u_j - \varphi_{ki}) = W_k\dot{\omega}_k \tag{4}$$

$$\frac{\partial T}{\partial t} + \frac{\sum_{k=1}^{N_{sp}} U_k(T)\frac{\partial \rho Y_k}{\partial t}\frac{1}{W_k}}{\sum_{k=1}^{N_{sp}}[C]_k\, C_{v,k}(T)} = 0 \tag{5}$$

$$\frac{\partial P}{\partial t} - \frac{R}{V}(T\frac{\partial n}{\partial t} + \frac{\partial T}{\partial t}n) = 0 \tag{6}$$

In these equations, ρ is the fluid density, u_i is the velocity in the i-th direction, and E is the total energy. δ_{ij} is the Kronecker delta, P is the pressure, T is the temperature, R is the universal gas constant, and Y_k is the mass fraction of species k. $[C]_k$ is the concentration of species k in the gas mixture, $C_{v,k}(T)$ is the specific heat at constant volume of species k at temperature T, and $U_k(T)$ is the internal energy of species k at temperature T. These quantities are defined using the NASA 9-coefficient polynomial parameterization [30], such that

$$\frac{C_{v,k}(T)}{R} = a_0 + a_1\, T + a_2\, T^2 + a_3\, T^3 + a_4\, T^4 - 1 \tag{7}$$

$$\frac{U_k(T)}{RT} = a_0 + \frac{a_1}{2}\, T + \frac{a_2}{3}\, T^2 + \frac{a_3}{4}\, T^3 + \frac{a_4}{5}\, T^4 + \frac{a_5}{T} - 1, \tag{8}$$

where the a coefficients are defined per-species for an arbitrary number of temperature regions.

Returning to the Navier-Stokes equations (1) - (6), N_{sp} is the number of species in the chemical mechanism, W_k is the molecular weight of species k, n is the number of moles of the gas mixture, V is the volume of the gas, and $\dot{\omega}_k$ is the net production rate of species

k. τ_{ij} is the viscous stress tensor, φ_{ki} are the diffusion fluxes, and q_j are the heat fluxes, defined by

$$q_j = -\frac{\partial(\lambda T)}{\partial x_j} + h_k \varphi_{kj}. \tag{9}$$

In this expression, λ is the thermal conductivity, and h_k is the enthalpy of species k. As for the diffusion fluxes, these are defined using a mixture-averaged approach - that is, φ_{ki} is given by

$$\varphi_{ki} = \varphi^*_{ki} + \varphi^c_{ki}, \tag{10}$$

where φ^*_{ki} is the mixture-averaged approximation, and φ^c_{ki} is a correction term to ensure mass conservation. The mixture-averaged approximation is defined by

$$\varphi^*_{ki} = -\rho D_{k,m} \frac{W_k}{W} \frac{\partial X_k}{\partial x_i} \tag{11}$$

where $D_{k,m}$ is the mixture-averaged diffusivity of species k, W is the mean molecular weight, and X_k is the mole fraction of species k. Finally, the correction term is given by

$$\varphi^c_{ki} = -Y_k \sum_{n=1}^{N_{sp}} \varphi^*_{ni} \tag{12}$$

In this set of governing equations (1) - (6), the mass fraction source term on the right side of equation (4) is typically observed to be stiff relative to the surrounding equations governing the motion of the gas mixture and the change in its physical properties. This presents a time integration problem in that this term alone typically dictates either an oppressively low timestep or the need for an implicit approach, often using tools such as CVODE. In practice, fluid solvers also often decouple an explicit treatment of the Navier-Stokes equations (1) - (3) from the implicit treatment of the chemical kinetics, resulting in a splitting approach that is at best first or second order in time.

The topic of this section of the thesis, then, is to answer the following research question: **can we derive performance and/or temporal accuracy benefits from an application of multi-rate Adams integrators (implicit and/or explicit) to this set of governing equations?**

1.3 Numerical Methods

1.3.1 Adams Methods

For clarity, here we provide a brief derivation of a standard Adams-Bashforth (AB) integrator, as described in [3], starting from a model IVP given by

$$\frac{\mathrm{d}y}{\mathrm{d}t} = F(t, y), \quad y(0) = y_0. \tag{13}$$

Applying a method of lines (MOL) approach to spatially discretize and solve the reactive Navier-Stokes equations (1) - (6) ultimately provides this form. We approximate the time dependency of the right-hand side function with a polynomial with coefficients $\boldsymbol{\alpha}$ (formed by interpolating past values of $F(t, y)$), extrapolate with that polynomial approximation, and integrate the extrapolant. To obtain the coefficients $\boldsymbol{\alpha}$ used to form this history-interpolating polynomial, we construct a linear system using a Vandermonde matrix:

$$V^T \cdot \boldsymbol{\alpha} = \int_0^{\Delta t} \tau^i d\tau = \frac{(\Delta t)^i}{i+1}, \quad i = 1, 2..., n, \quad V = \begin{bmatrix} 1 & t_1 & \dots & t_1^{n-1} \\ 1 & t_2 & \dots & t_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_n & \dots & t_n^{n-1} \end{bmatrix}, \tag{14}$$

where $\int_0^{\Delta t} \tau^i d\tau$ is a vector evaluating the integral of the interpolation polynomial, and V is the Vandermonde matrix with monomial basis and nodes $t_1, t_2, \dots t_n$, corresponding to past time values. In (14), n is equal to the order of the integrator, and t_i are the time history values, with $0 \leq t_1 < t_2 \dots < t_n$. The coefficients $\boldsymbol{\alpha}$ are used to extrapolate to the next state via

$$\begin{aligned} y(t_{i+1}) = y(t_i) + &\alpha_1 F(t_{i-n}, y_{i-n}) + \alpha_2 F(t_{i-n-1}, y_{i-n-1}) \\ &+ \dots + \alpha_n F(t_i, y_i). \end{aligned} \tag{15}$$

Clearly, the length of the past history needed to calculate a step (and, thus, the memory required) influences the order of accuracy attained. In [32], it is also shown that increasing the amount of tracked and interpolated history values beyond the explicit order requirement can also provide certain stability benefits.

An alternative time integration method is required for the first few time steps (the exact number of which is dependent on the number of history values needed) in order to establish right-hand side history and "bootstrap" the method. We use a third-order Runge-Kutta (RK3) integrator [17] to bootstrap the third-order AB methods, whereas a fourth-order Runge-Kutta (RK4) integrator [23] is used to bootstrap the fourth-order AB methods.

1.3.2 Multi-rate Adams Methods

We now describe a multi-rate generalization of the scheme, making use of the algorithm introduced in [15]. We consider the following model system with “fast” and “slow” solution components:

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{pmatrix} f(t) \\ s(t) \end{pmatrix} = \begin{pmatrix} a_f(f, s) \\ a_s(f, s) \end{pmatrix}. \tag{16}$$

With this in mind, we can set a slow (larger) time step H for a_s such that we maintain stability in the integration of the slow component. We also set a fast time step h for a_f such that H is an integer multiple of h, and define the ratio between the two, $\mathrm{SR} = H/h$, as the step ratio of the multi-rate scheme. While the results presented here make use of only two

separate state components, each with its own right-hand side function and independent rate, the theory is readily extensible to any number of rates.

Within this two-component scheme, a few design choices are available:

- The order in which we evaluate and advance the solution components. Namely, two primary options are advancing the fast-evolving solution component through all of its micro-timesteps h and waiting to perform the single macro-timestep H required for the slow component until the end (a "fastest-first" scheme, per the nomenclature of [15]), or pursuing an algorithm in which the slow component is instead advanced first.
- For slowest-first evaluation schemes, the choice of whether or not to re-extrapolate the slow state after additional state and right-hand side information is gathered at the micro-timestep level.

Empirical observations on the effects of these choices are made in [20]. It is useful to step through a brief example of a multi-rate Adams-Bashforth integrator, using a system with a fast component requiring twice as many timesteps as the slow component to remain well-resolved (SR = 2). We lay out the steps of a third-order fastest-first multi-rate Adams-Bashforth scheme with no re-extrapolation, assuming that a_s evolves at the slow rate (macro-timestep $H = 2h$) and a_f evolves at the fast rate (micro-timestep h). $\hat{a}$ denotes extrapolants of the right-hand side functions as polynomial functions of both time t and the set of history values $\vec{a}_{\text{hist}}$: $\hat{a} = P(t, \vec{a}_{\text{hist}})$. These polynomials approximating the evolution of a_f and a_s in time are what we will integrate to march a_f and a_s, and will be updated to replace older history values with new right-hand side evaluations during the course of integration through a macro-timestep H. We assume availability of right-hand side histories to start the AB method.

Step 1: Form the polynomial extrapolants we will integrate, per the AB methods described in the previous subsection:

$$\hat{a}_f(t) = P\Big(t, [a_f(f(t_{i-2}), s(t_{i-2})), a_f(f(t_{i-1}), s(t_{i-1})), a_f(f(t_i), s(t_i))]\Big)$$
$$\hat{a}_s(t) = P\Big(t, [a_s(f(t_{i-4}), s(t_{i-4})), a_s(f(t_{i-2}), s(t_{i-2})), a_s(f(t_i), s(t_i))]\Big)$$

The right-hand side history values of a_f (used to form $\hat{a}_f$) have been obtained at time points $t_{i-2} = t - 2h$, $t_{i-1} = t - h$, and current time $t_i = t$, whereas the right-hand side history values of a_s (used to form $\hat{a}_s$) have been obtained at time points $t_{i-4} = t - 2H$, $t_{i-2} = t - H$, and $t_i = t$.

Step 2: March both f and s to time t_{i+1} by integrating the polynomial extrapolants $\hat{a}_s(t)$ and $\hat{a}_f(t)$ formed in Step 1:

$$f(t_{i+1}) = f(t_i) + \int_{t_i}^{t_{i+1}} \hat{a}_f(\tau) d\tau,$$
$$s(t_{i+1}) = s(t_i) + \int_{t_i}^{t_{i+1}} \hat{a}_s(\tau) d\tau.$$

This results in a set of intermediate values $f(t_{i+1})$ and $s(t_{i+1})$.

Step 3: Evaluate the fast right-hand side $a_f(f(t_{i+1}), s(t_{i+1}))$.

Step 4: Update the set of right-hand side history values for a_f to include these new values, and construct a new extrapolant $\hat{a}_f$:

$$\hat{a}_f(t) = P\left(t, [a_f(f(t_{i-1}), s(t_{i-1})), a_f(f(t_i), s(t_i)), a_f(f(t_{i+1}), s(t_{i+1}))]\right).$$

Step 5: March s to time t_{i+2} by integrating the extrapolant formed in Step 1:

$$s(t_{i+2}) = s(t_i) + \int_{t_i}^{t_{i+2}} \hat{a}_s(\tau) d\tau.$$

Step 6: March f to time t_{i+2} by integrating the extrapolant formed in Step 3:

$$f(t_{i+2}) = f(t_{i+1}) + \int_{t_{i+1}}^{t_{i+2}} \hat{a}_f(\tau) d\tau.$$

Step 7: Evaluate $a_f(f(t_{i+2}), s(t_{i+2}))$ and $a_s(f(t_{i+2}), s(t_{i+2}))$ and update extrapolants:

$$\hat{a}_f(t) = P\Big(t, [a_f(f(t_i), s(t_i)), a_f(f(t_{i+1}), s(t_{i+1})), a_f(f(t_{i+2}), s(t_{i+2}))]\Big),$$
$$\hat{a}_s(t) = P\Big(t, [a_s(f(t_{i-2}), s(t_{i-2})), a_s(f(t_i), s(t_i)), a_s(f(t_{i+2}), s(t_{i+2}))]\Big).$$

The scheme evaluates the fast-evolving right-hand side a_f twice per macro-timestep H, whereas the slowly-evolving right-hand side a_s is only evaluated once. For the results shown later, this is the scheme we will use, albeit generalized to different step ratios $\mathrm{SR} = H/h$.

To fully realize the potential multi-rate formulation for reacting Navier-Stokes problems (for which the matrix of right-hand sides is not as simple as in [32], where both right-hand sides and state components are realized as per-overset-grid quantities), we need to introduce additive terms to (16), which we express as terms coupling the fast and slow states (a_{fs} and a_{sf}) and self-influencing terms (a_{ff} and a_{ss}):

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} f(t) \\ s(t) \end{pmatrix} = \begin{pmatrix} a_{ff}(f,s) + a_{fs}(f,s) \\ a_{sf}(f,s) + a_{ss}(f,s) \end{pmatrix}. \tag{17}$$

In the reacting flow formulation with which we are concerned (1) - (6), we define the "fast" component of our Navier-Stokes solution as a vector of the mass fractions of the species as well as the temperature and pressure, $[\rho Y_k, T, P]^T$. The "slow" component is then defined as the set of typical conserved variables in a non-reacting simulation, $[\rho, (\rho\vec{u}), (\rho E)]^T$. As

a result, one possible definition of the additive terms (17) can be given as follows:

$$a_{ff}(f,s) = W_k\dot{\omega}_k \tag{18}$$

$$a_{fs}(f,s) = -\frac{\partial}{\partial x_j}(\rho Y_k u_j - \varphi_{ki}) \tag{19}$$

$$a_{sf}(f,s) = \begin{pmatrix} 0 \\ -\frac{\partial}{\partial x_j}(P\delta_{ij}) \\ -\frac{\partial}{\partial x_j}(Pu_j) \end{pmatrix} \tag{20}$$

$$a_{ss}(f,s) = \begin{pmatrix} -\frac{\partial}{\partial x_j}(\rho u_j) \\ -\frac{\partial}{\partial x_j}(\rho u_i u_j - \tau_{ij}) \\ -\frac{\partial}{\partial x_j}((\rho E)u_j + q_j - u_i\tau_{ij}) \end{pmatrix} \tag{21}$$

As our treatment of this set of equations with multi-rate integration develops, so too will our approach to defining the additive terms of (17).

1.3.3 Implicit Adams Methods and Extension to Multi-rate

The concepts in the above sections, both single- and multi-rate, are easily extended to include implicit integration methods. We define an implicit Adams (Adams-Moulton [33]) method in the same manner as above, starting with the same simplified IVP (13) and expressing the scheme as

$$\begin{aligned} y(t_{i+1}) = y(t_i) &+ \beta_1 F(t_{i-n+1}, y_{i-n+1}) + \beta_2 F(t_{i-n}, y_{i-n}) \\ &+ \cdots + \beta_n F(t_{i+1}, y_{i+1}). \end{aligned} \tag{22}$$

Note the implicit nature of this time advancement scheme ($F(t_{i+1}, y_{i+1})$ is directly involved in the calculation of y_{i+1}). As for the coefficients $\boldsymbol{\beta}$, they are constructed in a manner similar to that of the Adams-Bashforth coefficients $\boldsymbol{\alpha}$, employing a Vandermonde matrix to create and solve a linear system

$$V^T \cdot \boldsymbol{\beta} = \int_0^{\Delta t} \tau^i d\tau = \frac{(\Delta t)^i}{i+1}, \quad i = 1, 2..., n, \quad V = \begin{bmatrix} 1 & t_1 & \dots & t_1^{n-1} \\ 1 & t_2 & \dots & t_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_n & \dots & t_n^{n-1} \end{bmatrix}, \tag{23}$$

with the Vandermonde matrix now incorporating nodes shifted Δt forwards in time relative to its Adams-Bashforth counterpart.

Extension of this implicit method into the multi-rate framework is straightforward, given that the driving principle of the Adams-Moulton method (namely, maintenance of right-hand side history values through which to interpolate) is the same as that of Adams-Bashforth, and amounts to allowing the histories of each right-hand side term to maintain their own timestep intervals.

In the case that implicit time integration is used, it will be used to drive $a_{ff}(f,s)$. In order to perform the implicit solve required to advance this right-hand side via Adams-Moulton,

a Jacobian matrix is required. While other implementations such as CVODE's adaptive BDF algorithm for stiff problems make use of finite-difference generated Jacobians, we will instead use PyJac-V2 [11] to obtain the analytical Jacobian for a given chemical mechanism via code generation, which will then be used by a simple Newton algorithm to perform the implicit solve. More details on the construction of this Jacobian are given by Curtis et. al, but we document the basic form, in addition to an alternate differential-algebraic formulation, below.

The chemical kinetics Jacobian can be written as the matrix

$$J = \frac{\partial f_i}{\partial \Phi_j} \tag{24}$$

where f is the set of right-hand sides directly pertaining to the kinetics and reactive source term:

$$f = \left(\frac{dT}{dt}, \frac{dP}{dt}, \frac{dn_1}{dt}, \frac{dn_2}{dt}, \dots \frac{dn_{N_{sp}-1}}{dt} \right). \tag{25}$$

While the temperature and pressure right-hand sides are given by (5) and (6), note that the molar source terms pertain only to the net production rates of the species:

$$\frac{dn_i}{dt} = V\dot{\omega}_i. \tag{26}$$

Note also (based on the structure of PyJac-V2) that the state vector for the Jacobian is defined using the number of moles of each species in the chemical mechanism rather than the mass fractions:

$$\Phi = \left(T, P, n_1, n_2, \dots n_{N_{sp}-1} \right). \tag{27}$$

The number of moles in of the last species in the mixture is calculated via the mass fractions summing to 1.

With these definitions in mind, we can expand (24) and express it in true matrix form:

$$J = \begin{bmatrix} \frac{d\dot{T}}{dT} & \frac{d\dot{T}}{dP} & \frac{d\dot{T}}{dn_1} & \cdots & \frac{d\dot{T}}{dn_{N_{sp}-1}} \\ \frac{d\dot{P}}{dT} & \frac{d\dot{P}}{dP} & \frac{d\dot{P}}{dn_1} & \cdots & \frac{d\dot{P}}{dn_{N_{sp}-1}} \\ \frac{d\dot{n}_1}{dT} & \frac{d\dot{n}_1}{dP} & \frac{d\dot{n}_1}{dn_1} & \cdots & \frac{d\dot{n}_1}{dn_{N_{sp}-1}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{d\dot{n}_{N_{sp}-1}}{dT} & \frac{d\dot{n}_{N_{sp}-1}}{dP} & \frac{d\dot{n}_{N_{sp}-1}}{dn_1} & \cdots & \frac{d\dot{n}_{N_{sp}-1}}{dn_{N_{sp}-1}} \end{bmatrix} \tag{28}$$

Providing this matrix to the implicit solver should serve to reduce the number of chemical right-hand side calls required to time-march the system relative to methods that employ a finite-difference generated Jacobian, which will directly improve performance.

A critical consideration in the formulation of both the governing equations and the chemical Jacobian will be maintaining a solution that lies on the ideal gas constraint manifold (i.e. a

physical solution) - this has been identified in early work as a potential sticking point when fully differential approaches to solving the governing equations that time-march the rates-of-change of temperature and pressure (as opposed to differential-algebraic approaches that directly satisfy the ideal gas and constant internal energy constraints) are used.

To avoid drift off of the ideal gas law manifold, a chemical kinetic Jacobian modified to include the algebraic constraints explicitly can be used. This approach, detailed in Hung [18], modifies the molar source terms to incorporate this constraint while removing the temperature and pressure source terms (which we then replace with a temperature Newton solve based on the constant internal energy constraint via Cantera [16]), as well as a direct application of the ideal gas law to obtain the pressure. This modified Jacobian can be expressed as

$$J^* = \begin{bmatrix} \frac{d\dot{n}_1^*}{dn_1} & \cdots & \frac{d\dot{n}_1^*}{dn_{N_{sp}-1}} \\ \vdots & \ddots & \vdots \\ \frac{d\dot{n}_{N_{sp}-1}^*}{dn_1} & \cdots & \frac{d\dot{n}_{N_{sp}-1}^*}{dn_{N_{sp}-1}} \end{bmatrix} \tag{29}$$

where

$$\frac{d\dot{n}_i^*}{dn_j} = \frac{d\dot{n}_i}{dn_j} - \frac{U_j}{C_v}\frac{d\dot{n}_i}{dT}. \tag{30}$$

In this expression, U_j is the contribution of the jth species to the internal energy.

Which of these Jacobians is ideal for our purposes is a consideration worth monitoring as our implicit integration techniques develop. In particular, a third relatively unexplored approach would use the fully differential method with pressure and temperature source terms, while also using attenuation to force the solution towards the constraint manifold. This attenuation approach is described in some detail (albeit for smaller model problems) by Ascher and Petzold [2].

1.3.4 Timestep Control Algorithm

Early implicit-explicit runs of small-scale reacting flow problems quickly identified a need to adapt the timestep of the integrator based on relative and/or absolute local error demands. We therefore employ a timestep control algorithm which, in its current form, is that of ODE45 [12], wherein a timestep multiplier r is calculated based on a local error estimate constructed from state estimates of two (differing) explicit orders:

$$r = \frac{\|s_{q+1} - s_q\|_2}{\text{ATOL} + \text{RTOL} \cdot \max(\|s_q\|_2, \|s_{q+1}\|_2)} \tag{31}$$

Here, s_q is the state estimate obtained using an order-q scheme, and ATOL and RTOL are absolute and relative error tolerances specified by the user. Once r is calculated, the timestep is *decreased* if $r >= 1$ via

$$\Delta t = 0.9\Delta t(r)^{-1/q} \tag{32}$$

and *increased* if $r < 1$ via

$$\Delta t = 0.9\Delta t(r)^{-1/(q+1)} \tag{33}$$

Another form under consideration is that of CVODE, which makes use of a weighted root-mean-square norm to more adequately handle states with components that feature a large variation in magnitude. The weights are calculated as

$$w_i = \frac{1}{\text{ATOL} + \text{RTOL}|s_{q,i}|} \tag{34}$$

with the weighted root-mean-square norm then being given by

$$\|E\|_{wrms} = \left(\sum_{i=1}^{N} \frac{1}{N} (w_i(s_{q+1,i} - s_{q,i}))^2 \right)^{1/2}. \tag{35}$$

Based on whether $\|E\|_{wrms}$ is less than or greater than 1, the timestep is modified in the same manner as above.

Regardless of the error norm used, this added capability alone presents a number of questions in terms of how best to accomplish this error control. In particular, step ratio adjustment *in situ*, rather than timestep adjustment, to meet error needs provides an implementation challenge in terms of code generation of these integrators, whereas the choice of which solution component(s) are involved in error estimation/timestep control also presents a design decision that could have notable effect on numerical performance.

1.4 Validation

1.4.1 Demonstrative Problem: Reacting Mixing Layer

The efficacy of the novel time integrators construced via the above methodology will be assessed using a reacting mixing layer problem with a hyperbolic tangent velocity profile for the baseflow (as used by [31] and [4]), with the San Diego 9-species chemical mechanism [45] employed for the chemical kinetics. By perturbing this baseflow with the mode having the highest growth rate to instability (reproduced based on the inviscid analyses of [31] and [4]), we can establish a fast-evolving solution component (the chemical kinetics/mass fractions of the species) and a slow-evolving solution component (the fluid), both possessing accessible means of rate modification (for the chemistry, we can accomplish this by scaling the pre-exponential factors in the Arrhenius reaction rates, and for the fluid, we can scale the magnitude of the initial crossflow velocity).

In doing so, what should result is a problem with ample testbed capabilities for our new multi-rate integrators, and also a problem firmly rooted in physical utility in that it provides the simplest model for scramjet/ramjet combustion processes.

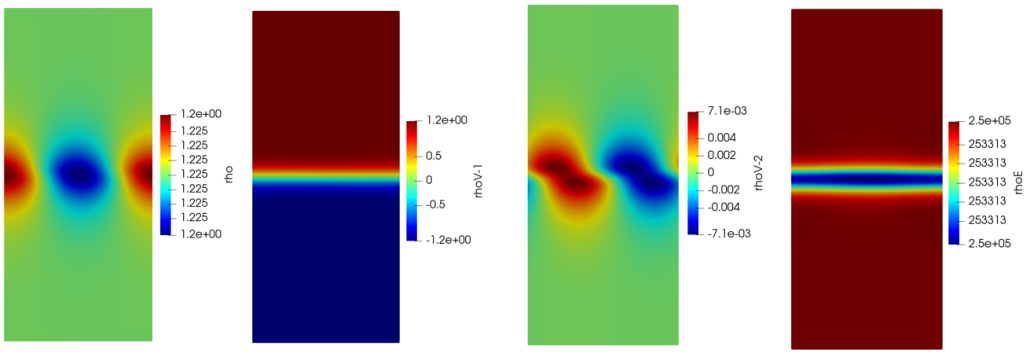

Figure 1: Perturbed initial condition of the hyperbolic tangent example - sea-level air.

1.5 Outlook

1.5.1 Current Status

At present, implicit-explicit time integration of reacting flows using Runge-Kutta based methods in conjunction with code generation of source terms and Jacobians for the chemical mechanisms is implemented and undergoing testing with the fluid solver application. Implicit Adams methods (with single-rate Adams-Moulton being the baseline) for the purposes of chemistry integration are also being implemented, along with an application of the error-informed timestep control algorithm to both single-rate and multi-rate Adams methods. Construction of implicit-explicit multi-rate Adams methods with error-informed adaptivity are well underway, with testing of these new methods on a small-scale Cantera [16] two-reactor system in progress.

As for the reacting crossflow validation problem, an initial setup employing cold flow (no autoignition) with the perturbed baseflow applied to sea level air is complete, with validation via the growth rate of the unstable mode ongoing. Also implemented in the reacting flow solver is a one-dimensional laminar free flame problem, which may also be used as a first-pass validation (via comparison to the Cantera-estimated steady-state flame speed) for new integration methods for reacting flows as they become available.

1.5.2 Risk Mitigation

An important question to ask amidst all of these promises is: **what if this doesn't work out?** Assuming demonstration of improved performance or accuracy of the *multi-rate* integrators over the state-of-the-art is for reasons unforseen impossible, a "fallback" takeaway would be demonstration of improvement over a low-order splitting approach via Leap implementation of an existing IMEX Runge-Kutta based scheme that treats the chemistry implicitly via code-generated analytical Jacobians. Given the use of a coupled implicit-explicit scheme (for which high order has already been proven in other circumstances), as well as a chemical Jacobian that is more accurately obtained than via finite differences (the method of CVODE), this outcome should be attainable at minimum.

As a step towards proving this, we have implemented a small-scale one-dimensional free-flame test, wherein we use a code-generated implementation of the fourth-order additive Runge-Kutta scheme of Kennedy and Carpenter to drive a simulation of a premixed, freely propagating hydrogen flame with an unburned gas temperature of 300 K at a pressure of one atmosphere. Comparing the estimated laminar flame speed to that of a steady-state analysis performed by Cantera, we see good agreement.

2 Energy-Stable Hybrid Spatial Discretizations

2.1 Motivation & Background

When simulating problems in computational fluid dynamics involving complex geometries, numerical difficulties are often encountered that require employing one of a number of different approaches to mitigate loss of accuracy and/or stability due to poor spatial resolution. While finite-difference approaches to spatial discretization are popular due to their simplicity and efficiency when working with smooth solutions, their geometric flexibility is typically limited in spite of the well-documented ability to use coordinate transforms to create intricate curvilinear meshes, as well as the ability to use overset meshes to produce higher resolution in particularly troublesome sections of the spatial domain.

In particular, the SBP methodology pioneered by Strand [39], which mimics integration-by-parts in the discrete sense and is provably stable via the energy method when coupled with weak enforcement of boundary conditions (simultaneous approximation term; see [41]), is an attractive one when it comes to finite differences, and its geometric flexibility has recently been further improved by a large body of work in creating accurate and stable interfaces between nonconforming blocks. In 2010, Mattsson and Carpenter [28] introduced interpolation operators that maintained the stability and high-order accuracy of the underlying spatial scheme on each block, but the resulting scheme is based on a fixed refinement ratio and the requirement that the blocks conform at their corners (the latter of these characteristics was later removed by Nissen et. al in 2015 [34]). The use of SBP-SAT finite difference techniques in conjunction with nonconforming grid interfaces has been further extended to use with the second-order wave equation [44], 3D elastic wave simulations [14], the Schrödinger equations [35], and the advection-diffusion equation [25].

Meanwhile, another attractive approach to modeling complex geometries more efficiently with a localized spatial discretization change is to create a hybrid spatial discretization that requires an interface between a high-order finite difference method (used away from the complex boundary) and an unstructured mesh with improved geometric flexibility (used near the complex boundary). Nordström and Gong proposed such a scheme in 2006 [36], coupling a high-order finite difference method and a second-order finite volume scheme. More recently, in 2016, Kozdon and Wilcox [22] developed an interface method between SBP finite-difference blocks that was based on projection into an intermediate "glue" grid characterized by piecewise polynomials, with the method also extending to the coupling between SBP finite-difference methods and discontinuous Galerkin methods. In this work, the interface method was proven to be stable via the energy method, but the projection

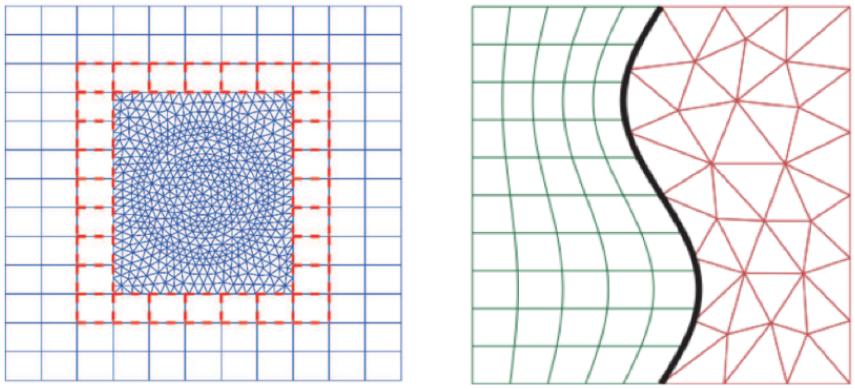

Figure 2: Two examples demonstrating use of nonconforming interfaces. Left: Zhu, Chen, Zhong, and Liu [46]. Right: Kozdon and Wilcox [22]

operators were not fully determined and were based in part on an optimization procedure focused on driving the product of the projection matrices closer to an identity operation. Furthermore, the method was not extended to three dimensions, and the method is reduced to first order accuracy if the continuous coordinate transforms on either side of the interface are dissimilar.

Since, Friedrich et. al [13] extended the method(s) of Kozdon and Wilcox to make the projection process degree-preserving (a characteristic that had been previously shown to be problematic by Lundquist and Nordström [26]) as well as energy-stable and conservative, but their approach still results in projection operators that are not fully determined, relying upon an optimization procedure to produce the operators that give their results. In 2019, Almquist et. al [1] developed new order-preserving interpolation operators that accomplished the same goal as Friedrich et. al without increasing the quadrature order, and which extended to problems with second-order spatial derivatives, but their results again were confined to problems with a fixed 2:1 refinement ratio. Furthermore, unlike Kozdon and Wilcox, both the methods of Almquist and Friedrich fail to extend their method to hybrid structured-unstructured discretizations.

2.1.1 Goals, Bounds, and Impact

This section of the proposal targets a lack of fully-determined high-order accurate and provably stable interface conditions between structured and unstructured meshes for computational fluid dynamic simulations. In particular, our aim is to create a provably stable, accurate, and fully-determined interface between summation-by-parts finite difference discretizations (structured) and discontinuous Galerkin methods (unstructured) that doesn't rely on optimization to establish its projection operators. Achieving this would provide a more robust option for creating localized areas of unstructured meshing (with superior flexibility for discretizing complex boundaries/geometries) in existing simulations, providing an alternative to comparatively less flexible overset curvilinear meshes or multiblock

conforming/nonconforming finite difference schemes.

2.2 Governing Equations

The initial problem to which our new hybrid method will be targeted is the acoustic wave equation in two dimensions, in first-order form:

$$\rho \frac{\partial v_i}{\partial t} + \frac{\partial p}{\partial x_i} = 0 \ \ (i = 1, 2), \tag{36}$$

$$\frac{\partial p}{\partial t} + \lambda \left(\frac{\partial v_1}{\partial x_1} + \frac{\partial v_2}{\partial x_2} \right) = 0 \tag{37}$$

As in [22], our aim will be to prove the semi-discrete hybrid discretization stable via the energy method by proving that

$$\frac{d\mathcal{E}}{dt} \leq 0 \tag{38}$$

where the total energy of the semi-discrete system $\mathcal{E}$ is obtained by summing the energy E of each discretization block:

$$\mathcal{E} = \sum_{blocks} E \tag{39}$$

The definitions of both the energies and the energy dissipation rates for each block (each having its own spatial discretization), are given in subsequent sections.

2.3 Numerical Methods

2.3.1 Summation-by-Parts Operators

For the structured discretization that makes up one component of the proposed hybrid, we make use of several finite difference operators that possess the summation-by-parts (SBP) property. Taking two matrices H, Q, we here state that these two matrices are SBP matrices of order p provided

- $H^{-1}Qv$ is an order h^p approximation to $\partial/\partial x$, where h is the spatial step size in one dimension.
- H is a symmetric positive-definite matrix.
- $Q + Q^T = \text{diag}(-1, 0, 0, ..., 0, 1)$.

These conditions together ensure that the discrete version of the integration by parts property holds; that is,

$$\langle H^{-1}Qx, y \rangle_H = x_N y_N - x_1 y_1 - \langle x, H^{-1}Qy \rangle_H .$$

The resulting operators can be either explicit (in this case, H is purely diagonal) or implicit. This theory, originally presented in [39], can also be extended to higher dimensions using Kronecker products. For example, in two dimensions, the matrices $P^{-1}G_x$ and $P^{-1}G_y$ define the x- and y- derivatives on a two-dimensional grid:

$$\begin{aligned} P &= H_x \otimes H_y \\ G_x &= Q_x \otimes H_y \\ G_y &= H_x \otimes Q_y \end{aligned}$$

Note that this formulation assumes that we have H_x, Q_x, a pair of $n_x \times n_x$ SBP matrices of approximation order p, and H_y, Q_y, a pair of $n_y \times n_y$ SBP matrices of approximation order q. Finally, we also note that these SBP operators do not guarantee strict stability for an initial boundary value problem. We must also apply the boundary conditions using a formulation that permits an energy estimate. More information on these boundary conditions, called simultaneous approximation term (SAT) boundary conditions, can be found at [41], [43], and [5].

More information on SBP operators of various order, including the coefficients themselves, can be found in [39], [6], [29]. For this work, we will employ diagonal-norm SBP operators for the finite difference discretizations.

The SBP-SAT spatial discretization of the governing equations in the previous section ((36) - (37)), in two dimensions and including the effect of coordinate transform from the physical space to a reference space $[-1,1] \times [-1,1]$, is given by

$$\rho \boldsymbol{J} \frac{d\boldsymbol{v}_i}{dt} + \boldsymbol{D}_1 \boldsymbol{J} \frac{\boldsymbol{\partial \xi_1}}{\boldsymbol{\partial x_i}} \boldsymbol{p} + \boldsymbol{D}_2 \boldsymbol{J} \frac{\boldsymbol{\partial \xi_2}}{\boldsymbol{\partial x_i}} \boldsymbol{p} = -\boldsymbol{H}^{-1} \boldsymbol{\mathcal{F}}_{v_i}, \quad i = 1, 2, \tag{40}$$

$$\boldsymbol{J} \frac{\partial \boldsymbol{p}}{\partial t} + \lambda \left(\boldsymbol{J} \frac{\boldsymbol{\partial \xi_1}}{\boldsymbol{\partial x_1}} \boldsymbol{D}_1 v_1 + \boldsymbol{J} \frac{\boldsymbol{\partial \xi_2}}{\boldsymbol{\partial x_1}} \boldsymbol{D}_2 v_1 + \boldsymbol{J} \frac{\boldsymbol{\partial \xi_1}}{\boldsymbol{\partial x_2}} \boldsymbol{D}_1 v_2 + \boldsymbol{J} \frac{\boldsymbol{\partial \xi_2}}{\boldsymbol{\partial x_2}} \boldsymbol{D}_2 v_2 \right) = -\lambda \boldsymbol{H}^{-1} \boldsymbol{\mathcal{F}}_p. \tag{41}$$

Here, we have defined the matrices

$$\boldsymbol{H} = \boldsymbol{H}_{N_1} \otimes \boldsymbol{H}_{N_2}, \qquad \boldsymbol{D}_1 = \boldsymbol{D}_{N_1} \otimes \boldsymbol{I}_{N_2}, \qquad \boldsymbol{D}_2 = \boldsymbol{I}_{N_1} \otimes \boldsymbol{D}_{N_2}, \tag{42}$$

where $\boldsymbol{D}_i = \boldsymbol{H}_i^{-1} \boldsymbol{Q}_i$ defines an SBP approximation of $\frac{\partial}{\partial x_i}$. We define corresponding solution vectors as, for example,

$$\boldsymbol{p} = \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0N_2} & p_{10} & \cdots & p_{N_1 N_2} \end{bmatrix}^T \tag{43}$$

for a domain of size $(N_1 + 1) \times (N_2 + 1)$. Note also that in these equations, x_i refer to coordinates in the physical domain, whereas ξ_i refer to coordinates in the reference domain. The matrix $\boldsymbol{J}$ is a diagonal matrix

$$\boldsymbol{J} = \operatorname{diag} \begin{bmatrix} J_{00} & J_{01} & \ldots & J_{0N_2} & J_{10} & \ldots & J_{N_1 N_2} \end{bmatrix}, \tag{44}$$

where the elements J_{kl} correspond to Jacobian determinants at grid point (k, l):

$$J = \frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial x_2}{\partial \xi_1} \frac{\partial x_1}{\partial \xi_2}. \tag{45}$$

The bolded coordinate transform matrices $\frac{\boldsymbol{\partial \xi_i}}{\boldsymbol{\partial x_j}}$ and $\frac{\boldsymbol{\partial x_i}}{\boldsymbol{\partial \xi_j}}$ are defined analogously.

The penalty terms in the above equations implement the boundary conditions via the SAT method, and are defined as follows:

$$\boldsymbol{\mathcal{F}}_{v_i} = \left(\boldsymbol{e}_W \otimes \boldsymbol{\mathcal{F}}_{v_i}^W\right) + \left(\boldsymbol{e}_E \otimes \boldsymbol{\mathcal{F}}_{v_i}^E\right) + \left(\boldsymbol{\mathcal{F}}_{v_i}^S \otimes \boldsymbol{e}_S\right) + \left(\boldsymbol{\mathcal{F}}_{v_i}^N \otimes \boldsymbol{e}_N\right), \tag{46}$$

$$\boldsymbol{\mathcal{F}}_{p} = \left(\boldsymbol{e}_W \otimes \boldsymbol{\mathcal{F}}_{p}^W\right) + \left(\boldsymbol{e}_E \otimes \boldsymbol{\mathcal{F}}_{p}^E\right) + \left(\boldsymbol{\mathcal{F}}_{p}^S \otimes \boldsymbol{e}_S\right) + \left(\boldsymbol{\mathcal{F}}_{p}^N \otimes \boldsymbol{e}_N\right). \tag{47}$$

In these expressions, the subscripts E, W, N, S refer to the edges of the domain, with the $\boldsymbol{e}$ vectors being equal to zero in every entry except for the first or last. The actual penalty vectors $\boldsymbol{\mathcal{F}}_{v_i}^W$ and $\boldsymbol{\mathcal{F}}_{p}^W$ are expressed as

$$\boldsymbol{\mathcal{F}}_{v_i}^W = \boldsymbol{H}_2 \boldsymbol{S}_{JW} \boldsymbol{n}_i^W \left(\boldsymbol{p}_W^* - \boldsymbol{p}_W\right), \qquad \boldsymbol{\mathcal{F}}_{p}^W = \boldsymbol{H}_2 \boldsymbol{S}_{JW} \lambda \left(\boldsymbol{v}_W^* - \boldsymbol{v}_W\right), \tag{48}$$

where the surface normals and Jacobians are given by

$$S_J = J\sqrt{\left(\frac{\partial \xi_i}{\partial x_2}\right)^2 + \left(\frac{\partial \xi_i}{\partial x_1}\right)^2}, \quad n_1 = \pm \frac{J}{S_J}\frac{\partial \xi_i}{\partial x_2}, \quad n_2 = \pm \frac{J}{S_J}\frac{\partial \xi_i}{\partial x_1}, \tag{49}$$

and the corresponding bolded matrices are constructed to be diagonal and contain values of these quantities at each grid point, as with $\boldsymbol{J}$ (see (44)). The starred vectors $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ are specified by the boundary or interface condition.

It can be shown (see [22]) that the energy dissipation rate of an SBP block in this case is given by

$$\frac{dE}{dt} = \sum_{K=\{W,E,S,N\}} \mathcal{D}_K, \tag{50}$$

$$\mathcal{D}_K = -\boldsymbol{v}_K^T \boldsymbol{H}_K \boldsymbol{S}_{JK} \boldsymbol{p}_K^* + \boldsymbol{v}_K^T \boldsymbol{H}_K \boldsymbol{S}_{JK} \boldsymbol{p}_K - \left(\boldsymbol{v}_K^*\right)^T \boldsymbol{H}_K \boldsymbol{S}_{JK} \boldsymbol{p}_K. \tag{51}$$

Naturally, this dissipation rate, in conjunction with a definition of $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ based on the numerical characteristics of the interface, will be critical in proving stability via the energy method.

2.3.2 Discontinuous Galerkin Method

The discontinuous Galerkin method we will target our interface method(s) to will be briefly summarized here. As in Kozdon and Wilcox, we use a curvilinear triangle-based DG method, and we denote a DG element Ω_e with triangular reference element $\tilde{\Omega}$. To obtain a version of the governing equations (36) - (37) usable by DG, we must first obtain the variational form by introducing test functions w_i and φ for the velocities and pressure respectively, multiplying them into their respective equations, and integrating over the

reference element:

$$\int_{\tilde{\Omega}} w_i \left[\rho J \frac{\partial v_i}{\partial t} + \frac{\partial}{\partial \xi_1}\left(J \frac{\partial \xi_1}{\partial x_i} p\right) + \frac{\partial}{\partial \xi_2}\left(J \frac{\partial \xi_2}{\partial x_i} p\right)\right] \, dA$$
$$= -\int_{\partial\tilde{\Omega}} w_i S_J n_i \left(p^* - p\right) ds, \quad i = 1, 2, \tag{52}$$
$$\int_{\tilde{\Omega}} \varphi \left[J \frac{\partial p}{\partial t} + \lambda \left(J \frac{\partial \xi_1}{\partial x_1} \frac{\partial v_1}{\partial \xi_1} + J \frac{\partial \xi_2}{\partial x_1} \frac{\partial v_1}{\partial \xi_2} + J \frac{\partial \xi_1}{\partial x_2} \frac{\partial v_2}{\partial \xi_1} + J \frac{\partial \xi_2}{\partial x_2} \frac{\partial v_2}{\partial \xi_2}\right)\right] \, dA$$
$$= -\int_{\partial\tilde{\Omega}} \varphi \lambda S_J \left(v^* - v\right) ds, \tag{53}$$

In these equations, the normals n_i and surface Jacobians S_J are the same as the ones defined by equations (49), and the starred quantities again refer to boundary and interface conditions implemented via penalization, in this instance better referred to as numerical fluxes. Applying integration by parts to (52) to move the spatial derivatives to the test functions gives us the skew-symmetric form of the variational equations, which can then be spatially discretized with the DG method (see Hesthaven and Warburton for more detail on this) to obtain:

$$\rho \boldsymbol{M}_J \frac{d\boldsymbol{v}_i}{dt} = \boldsymbol{D}_1^T \boldsymbol{M}_{1i} \boldsymbol{p} + \boldsymbol{D}_2^T \boldsymbol{M}_{2i} \boldsymbol{p} - \sum_{K=1}^{3} \boldsymbol{L}_K^T \boldsymbol{P}_{bc}^T \boldsymbol{n}_{iK} \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \boldsymbol{p}_K^*, \tag{54}$$

$$\boldsymbol{M}_J \frac{d\boldsymbol{p}}{dt} = -\lambda \left(\boldsymbol{M}_{11} \boldsymbol{D}_1 \boldsymbol{v}_1 + \boldsymbol{M}_{21} \boldsymbol{D}_2 \boldsymbol{v}_1 + \boldsymbol{M}_{12} \boldsymbol{D}_1 \boldsymbol{v}_2 + \boldsymbol{M}_{22} \boldsymbol{D}_2 \boldsymbol{v}_2\right)$$
$$- \sum_{K=1}^{3} \lambda \boldsymbol{L}_K^T \boldsymbol{P}_{bc}^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \left(\boldsymbol{v}_K^* - \boldsymbol{v}_K^-\right). \tag{55}$$

These equations provide a semi-discretization on each element. The vector $\boldsymbol{v}_K^-$ is the normal component of velocity along edge K of the element evaluated at the cubature points:

$$\boldsymbol{v}_K^- = \boldsymbol{n}_{1K} \boldsymbol{P}_{bc} \boldsymbol{L}_K \boldsymbol{v}_1 + \boldsymbol{n}_{2K} \boldsymbol{P}_{bc} \boldsymbol{L}_K \boldsymbol{v}_2. \tag{56}$$

Edge-projected pressures are similarly defined as

$$\boldsymbol{p}_K^- = \boldsymbol{P}_{bc} \boldsymbol{L}_K \boldsymbol{p}. \tag{57}$$

In these expressions, $\boldsymbol{L}_K$ and $\boldsymbol{L}_K^T$ are operators which take volume terms to edge K of the element and edge K terms to the volume respectively. D_1 and D_2 are the reference element differentiation matrices for the two reference coordinate directions. The projection matrices P_c and P_{bc} project from the volume and edge approximations to the volume and edge cubature points, respectively. The matrices Ω_c and Ω_{bc} are diagonal matrices of the integration weights for the volume and an edge, respectively, defined at the cubature locations. A critical condition assumed by [22] for stability of their method is that Ω_c and Ω_{bc} are positive definite. Finally, the element mass matrices in the discretization are defined as

$$\boldsymbol{M}_J = \boldsymbol{P}_c^T \boldsymbol{\Omega}_c \boldsymbol{J} P_c, \quad \boldsymbol{M}_{ij} = \boldsymbol{P}_c^T \boldsymbol{\Omega}_c \boldsymbol{J} \frac{\boldsymbol{\partial \xi_i}}{\boldsymbol{\partial x_j}} \boldsymbol{P}_c. \tag{58}$$

Here the diagonal matrices $\boldsymbol{J}$ and $\frac{\partial \boldsymbol{\xi_i}}{\partial \boldsymbol{x_j}}$ are defined identically to (44) and defined at the cubature points. The diagonal matrices $\boldsymbol{S}_{JK}$ and $\boldsymbol{n}_{iK}$ are defined in the same way at the cubature points.

As with SBP, it can be shown that a single DG element has the energy dissipation rate

$$\frac{dE}{dt} = \sum_{K=1}^{3} \mathcal{D}_K, \tag{59}$$

$$\mathcal{D}_K = -\left(\boldsymbol{v}_K^-\right)^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \boldsymbol{p}_K^* - \left(\boldsymbol{p}_K^-\right)^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \left(\boldsymbol{v}_K^* - \boldsymbol{v}_K^-\right). \tag{60}$$

2.3.3 Interface Method

The key contribution of this work will be construction of an interface between the structured (finite difference SBP) and the unstructured (DG) spatial discretizations that is stable and accurate, and does not rely upon optimization or an unequal number of constraints and unknowns for construction of the solution projections. Currently, our plan is to match the approach of [22], with the interface being defined by operators $\boldsymbol{P}_{a2b}$ and $\boldsymbol{P}_{b2a}$ that project the solution along the edge of discretization a into discretization b, and vice versa.

As detailed in their paper, the operators of Kozdon and Wilcox rely primarily upon the principle of projecting solutions from either discretization first into an intermediate "glue" space of piecewise continuous polynomials before projecting the solution onto the other nonconforming discretization. The critical condition that allows proof of stability is referred to as the *compatibility condition*, and amounts to a simple relation between the grid-to-glue and glue-to-grid projection operators, the mass matrix of the piecewise polynomial description used in the glue layer, and a norm operator $\boldsymbol{H}$:

$$\boldsymbol{P}_{f2g}\boldsymbol{M} = \boldsymbol{P}_{g2f}^T \boldsymbol{H} \tag{61}$$

In the case of SBP finite difference discretizations, the norm $\boldsymbol{H}$ is equivalent to the SBP-norm, which for the purposes of this work we will take to be a diagonal matrix. In the case of discontinuous Galerkin, this $\boldsymbol{H}$ operator is the diagonal matrix of integration weights at the cubature points at a single DG element edge, $\boldsymbol{\Omega}_{bc}$.

This compatibility condition, used in conjunction with the per-block semi-discrete expressions for energy (51) and (60), is what allows for energy stability to be proven, and as a result, this condition is used as an explicit constraint when constructing the projection operators $\boldsymbol{P}_{f2g}$ and $\boldsymbol{P}_{g2f}$ themselves. The construction, which is laid out in further detail in [22], employs a Legendre polynomial basis on the glue grid, and defines interior and boundary accuracy conditions when projecting from grid to glue (and vice versa) that match those of corresponding SBP diagonal-norm operators of the same order.

These conditions, in conjunction with symmetry conditions and a set of unknowns as defined in the figure below, result in a *non-square* system featuring more unknowns than constraints, in particular near the boundary stencils. Given the underdetermined nature of the system, the authors then use an optimization procedure aimed at minimizing the

distance between the nearest eigenvalues of $\boldsymbol{B} = \boldsymbol{P}_{f2g}\boldsymbol{P}_{g2f}$ to produce the final projection operators for a given order.

With all of this borne in mind, our primary goal for the interface method, then, is to obtain projection operators that maintain the provable energy stability of Kozdon and Wilcox, which are also

- obtained from a construction procedure that is demonstrably *unisolvent*, with an equal number of constraints and unknowns resulting in a square system of equations
- therefore not based on optimization
- able to provide "round-trip" fidelity of projection, meaning that projection from the grid to the glue, and then back to the grid, results in a solution identical to the starting point.

The strategy for pursuance of these goals begins with consideration of the *interior* projection operator only, given the simplicity of the existing interior stencil relative to the boundary (which features many more unknowns). Noting also that the compatibility condition disallows us from obtaining exact "round-trip" projection, other conditions may be considered based on further manipulation or restructuring of the edge dissipation relations (51) and (60).

2.4 Validation

The test problem for validation will match that of [22], modeling the two-dimensional acoustic wave equation in first-order form with the following initial condition:

$$p(x_1, x_2, 0) = \cos(k_1 x_1)\cos(k_1 x_2) + \sin(k_2 x_1)\sin(k_2 x_2), \tag{62}$$

$$v_i(x_1, x_2, 0) = 0, \qquad i = 1, 2, \tag{63}$$

where $k_1 = \pi/2$ and $k_2 = \pi$. All exterior boundary conditions are zero pressure (free-surface) conditions. The exact solution of this problem is

$$p(x_1, x_2, t) = \cos(\omega_1 t)\cos(k_1 x_1)\cos(k_1 x_2) + \cos(\omega_2 t)\sin(k_2 x_1)\sin(k_2 x_2), \tag{64}$$

$$v_1(x_1, x_2, t) = \frac{k_1}{\omega_1}\sin(\omega_1 t)\sin(k_1 x_1)\cos(k_1 x_2) - \frac{k_2}{\omega_2}\sin(\omega_2 t)\cos(k_2 x_1)\sin(k_2 x_2), \tag{65}$$

$$v_2(x_1, x_2, t) = \frac{k_1}{\omega_1}\sin(\omega_1 t)\cos(k_1 x_1)\sin(k_1 x_2) - \frac{k_2}{\omega_2}\sin(\omega_2 t)\sin(k_2 x_1)\cos(k_2 x_2), \tag{66}$$

where $\omega_j = k_j\sqrt{2}$ for $j = 1, 2$.

As for the spatial discretization, we will employ a square spatial domain with the left half using diagonal-norm SBP operators for finite differencing, and the right half using a discontinuous Galerkin method. The interior order of the SBP method, as well as the order of the projection across the interface, is specified to match the order of the discontinous Galerkin method.

Our success metrics for this test problem include:

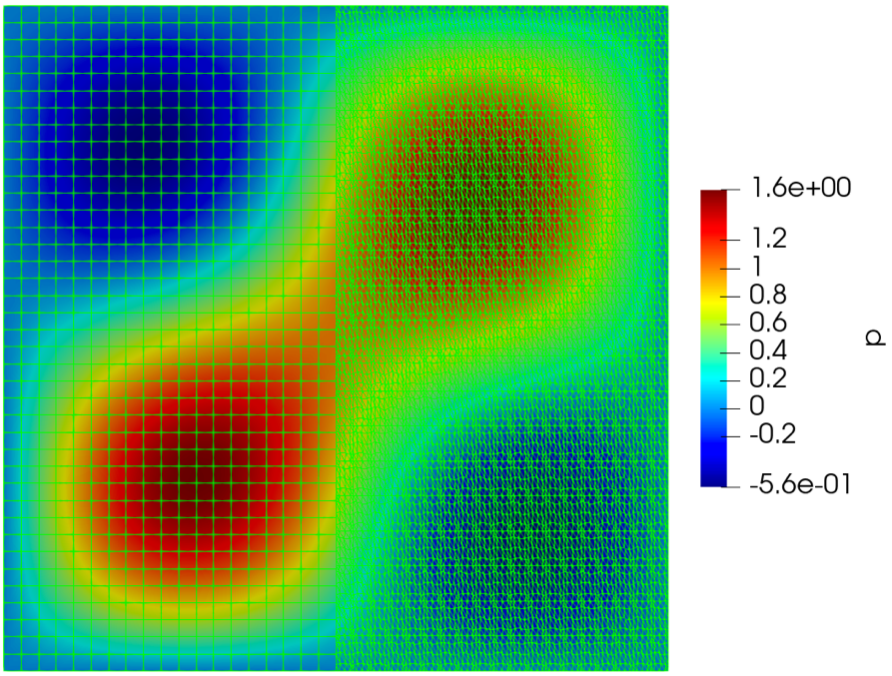

Figure 3: Pressure initial condition for test problem, also showing split-domain discretization.

- Demonstrating accurate recreation of the total energy of the system, as observed in the numerical simulation of the problem, via a total energy relation derived from the spatial and interfacial schema.
- Demonstrating that this energy does not increase (condition for semi-discrete stability).
- Demonstrating the prescribed order of accuracy using L2 norms.

Upon fulfillment of these goals, extension to three dimensions will be considered.

2.5 Outlook

2.5.1 Current Status

2.5.2 Risk Mitigation

Should our goal of a new and unisolvent stable interface method not come to fruition, a "fallback" goal of sorts is identified in that Kozdon and Wilcox's method has yet to be extended to three dimensions. This would present more of a technical challenge than an intellectual one, but a study of how the method responds to certain geometric scenarios untestable in two dimensions (namely three-dimensional corners) has potential to yield further information about how the optimized operators perform numerically, as well as

shed light on optimization targets to yield better results in higher dimensions.

Furthermore, an additional unexplored topic that could prove to be a worthwhile "fallback" goal - and one that potentially ties the two topics of this proposal together - would be exploring the implications of the use of Kozdon and Wilcox's projection operators on the timestep limitations of a given physical problem, and (if a timestep disparity between the interior of a given subdomain and its interface with another subdomain is identified) exposing this challenge to multi-rate Adams integration.

References

[1] Martin Almquist, Siyang Wang, and Jonatan Werpers. Order-preserving interpolation for summation-by-parts operators at nonconforming grid interfaces. *SIAM Journal on Scientific Computing*, 41(2):A1201–A1227, 2019.

[2] Uri M Ascher and Linda R Petzold. *Computer methods for ordinary differential equations and differential-algebraic equations*, volume 61. Siam, 1998.

[3] Francis Bashforth and John Couch Adams. *An attempt to test the theories of capillary action*. University Press, 1883.

[4] William Blumen. Shear layer instability of an inviscid compressible fluid. *Journal of Fluid Mechanics*, 40(4):769–781, 1970.

[5] Daniel J Bodony. Accuracy of the simultaneous-approximation-term boundary condition for time-dependent problems. *Journal of Scientific Computing*, 43(1):118–133, 2010.

[6] Mark H Carpenter, David Gottlieb, and Saul Abarbanel. Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes. Technical report, NASA, 1993.

[7] Scott D Cohen, Alan C Hindmarsh, and Paul F Dubois. Cvode, a stiff/nonstiff ode solver in c. *Computers in physics*, 10(2):138–143, 1996.

[8] Emil M Constantinescu and Adrian Sandu. Multirate timestepping methods for hyperbolic conservation laws. *Journal of Scientific Computing*, 33(3):239–278, 2007.

[9] Emil M Constantinescu and Adrian Sandu. Extrapolated implicit-explicit time stepping. *SIAM Journal on Scientific Computing*, 31(6):4452–4477, 2010.

[10] Emil M Constantinescu and Adrian Sandu. Extrapolated multirate methods for differential equations with multiple time scales. *Journal of Scientific Computing*, 56(1):28–44, 2013.

[11] Nicholas J Curtis, Kyle E Niemeyer, and Chih-Jen Sung. Using simd and simt vectorization to evaluate sparse chemical kinetic jacobian matrices and thermochemical source terms. *Combustion and Flame*, 198:186–204, 2018.

[12] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae.

Journal of computational and applied mathematics, 6(1):19–26, 1980.

[13] Lucas Friedrich, David C Del Rey Fernández, Andrew R Winters, Gregor J Gassner, David W Zingg, and Jason Hicken. Conservative and stable degree preserving sbp operators for non-conforming meshes. *Journal of Scientific Computing*, 75(2):657–686, 2018.

[14] Longfei Gao, Omar Ghattas, and David Keyes. Energy-conserving 3d elastic wave simulation with finite difference discretization on staggered grids with nonconforming interfaces. *arXiv preprint arXiv:2012.13863*, 2020.

[15] Charles William Gear and DR Wells. Multirate linear multistep methods. *BIT Numerical Mathematics*, 24(4):484–502, 1984.

[16] David G. Goodwin, Raymond L. Speth, Harry K. Moffat, and Bryan W. Weber. Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes. `https://www.cantera.org`, 2018. Version 2.4.0.

[17] Karl Heun. Neue methoden zur approximativen integration der differentialgleichungen einer unabhängigen veränderlichen. *Z. Math. Phys*, 45:23–38, 1900.

[18] Patrick Hin Fun Hung. *Algorithms for reaction mechanism reduction and numerical simulation of detonations initiated by projectiles*. PhD thesis, California Institute of Technology, 2003.

[19] Christopher A Kennedy and Mark H Carpenter. Additive runge–kutta schemes for convection–diffusion–reaction equations. *Applied numerical mathematics*, 44(1-2):139–181, 2003.

[20] Andreas Klöckner. *High-performance high-order simulation of wave and plasma phenomena*. PhD thesis, Brown University, 2010.

[21] Omar M Knio, Habib N Najm, and Peter S Wyckoff. A semi-implicit numerical scheme for reacting flow: Ii. stiff, operator-split formulation. *Journal of Computational Physics*, 154(2):428–467, 1999.

[22] Jeremy E Kozdon and Lucas C Wilcox. Stable coupling of nonconforming, high-order finite difference methods. *SIAM Journal on Scientific Computing*, 38(2):A923–A952, 2016.

[23] Wilhelm Kutta. Beitrag zur näherungweisen integration totaler differentialgleichungen. 1901.

[24] Simon Lapointe, Sudeepta Mondal, and Russell A Whitesides. Data-driven selection of stiff chemistry ode solver in operator-splitting schemes. *Combustion and Flame*, 220:133–143, 2020.

[25] Tomas Lundquist, Arnaud Malan, and Jan Nordström. A hybrid framework for coupling arbitrary summation-by-parts schemes on general meshes. *Journal of Computational Physics*, 362:49–68, 2018.

[26] Tomas Lundquist and Jan Nordström. On the suboptimal accuracy of summation-by-parts schemes with non-conforming block interfaces, 2016.

[27] Jonathan F MacArt and Michael E Mueller. Semi-implicit iterative methods for low mach number turbulent reacting flows: Operator splitting versus approximate factorization. *Journal of Computational Physics*, 326:569–595, 2016.

[28] Ken Mattsson and Mark H Carpenter. Stable and accurate interpolation operators for high-order multiblock finite difference methods. *SIAM Journal on Scientific Computing*, 32(4):2298–2320, 2010.

[29] Ken Mattsson, Magnus Svärd, and Jan Nordström. Stable and accurate artificial dissipation. *Journal of Scientific Computing*, 21(1):57–79, 2004.

[30] Bonnie J McBride. *NASA Glenn coefficients for calculating thermodynamic properties of individual species*. National Aeronautics and Space Administration, John H. Glenn Research Center ..., 2002.

[31] Alfons Michalke. On the inviscid instability of the hyperbolictangent velocity profile. *Journal of Fluid Mechanics*, 19(4):543–556, 1964.

[32] Cory Mikida, Andreas Klöckner, and Daniel Bodony. Multi-rate time integration on overset meshes. *Journal of Computational Physics*, 396:325–346, 2019.

[33] Forest Ray Moulton. *New methods in exterior ballistics*. Chicago, 1926.

[34] Anna Nissen, Katharina Kormann, Magnus Grandin, and Kristoffer Virta. Stable difference methods for block-oriented adaptive grids. *Journal of Scientific Computing*, 65(2):486–511, 2015.

[35] Anna Nissen, Gunilla Kreiss, and Margot Gerritsen. Stability at nonconforming grid interfaces for a high order discretization of the schrödinger equation. *Journal of Scientific Computing*, 53(3):528–551, 2012.

[36] Jan Nordström and Jing Gong. A stable hybrid method for hyperbolic problems. *Journal of Computational Physics*, 212(2):436–453, 2006.

[37] Zhuyin Ren and Stephen B Pope. Second-order splitting schemes for a class of reactive systems. *Journal of Computational Physics*, 227(17):8165–8176, 2008.

[38] Bruno Sportisse, Guy Bencteux, and Pierre Plion. Method of lines versus operator splitting for reaction–diffusiuon systems with fast chemistry. *Environmental Modelling & Software*, 15(6-7):673–679, 2000.

[39] Bo Strand. Summation by parts for finite difference approximations for d/dx. *Journal of Computational Physics*, 110(1):47–67, 1994.

[40] Gilbert Strang. On the construction and comparison of difference schemes. *SIAM journal on numerical analysis*, 5(3):506–517, 1968.

[41] Magnus Svärd, Mark H Carpenter, and Jan Nordström. A stable high-order finite difference scheme for the compressible Navier–Stokes equations, far-field boundary conditions. *Journal of Computational Physics*, 225(1):1020–1038, 2007.

[42] Magnus Svärd and Siddhartha Mishra. Implicit–explicit schemes for flow equations with stiff source terms. *Journal of computational and applied mathematics*,

235(6):1564–1577, 2011.

[43] Magnus Svärd and Jan Nordström. A stable high-order finite difference scheme for the compressible Navier–Stokes equations: no-slip wall boundary conditions. *Journal of Computational Physics*, 227(10):4805–4824, 2008.

[44] Siyang Wang, Kristoffer Virta, and Gunilla Kreiss. High order finite difference methods for the wave equation with non-conforming grid interfaces. *Journal of Scientific Computing*, 68(3):1002–1028, 2016.

[45] San Diego Mechanism web page. Chemical-kinetic mechanisms for combustion applications.

[46] Bao Zhu, Jiefu Chen, Wanxie Zhong, and Qing Huo Liu. A hybrid fetd-fdtd method with nonconforming meshes. *Communications in Computational Physics*, 9(3):828–842, 2011.