Evaluation of finite difference based asynchronous partial differential equations solver for reacting flows

Komal Kumari [a], Emmet Cleary [b], Swapnil Desai [c], Diego A. Donzis [a], Jacqueline H. Chen [c], Konduri Aditya [d],*

[a] Department of Aerospace Engineering, Texas A&M University, College Station, TX, United States
[b] California Institute of Technology, Pasadena, CA, United States
[c] Combustion Research Facility, Sandia National Laboratories, Livermore, CA, United States
[d] Department of Computational and Data Sciences, Indian Institute of Science, Bengaluru, KA, India

A B S T R A C T

Next-generation exascale machines with extreme levels of parallelism will provide massive computing resources for large scale numerical simulations of complex physical systems at unprecedented parameter ranges. However, novel numerical methods, scalable algorithms and re-design of current state-of-the art numerical solvers are required for scaling to these machines with minimal overheads. One such approach for partial differential equations based solvers involves computation of spatial derivatives with possibly delayed or asynchronous data using high-order asynchrony-tolerant (AT) schemes to facilitate mitigation of communication and synchronization bottlenecks without affecting the numerical accuracy. In the present study, an effective methodology of implementing temporal discretization using a multi-stage Runge-Kutta method with AT schemes is presented. Together these schemes are used to perform asynchronous simulations of canonical reacting flow problems, demonstrated in one-dimension including auto-ignition of a premixture, premixed flame propagation and non-premixed autoignition. Simulation results show that the AT schemes incur very small numerical errors in all key quantities of interest including stiff intermediate species despite delayed data at processing element (PE) boundaries. For simulations of supersonic flows, the degraded numerical accuracy of well-known shock-resolving WENO (weighted essentially non-oscillatory) schemes when used with relaxed synchronization is also discussed. To overcome this loss of accuracy, high-order AT-WENO schemes are derived and tested on linear and non-linear equations. Finally the novel AT-WENO schemes are demonstrated in the propagation of a detonation wave with delays at PE boundaries.

1. Introduction

The fundamental understanding of a multitude of complex scientific and engineering phenomena relies extensively on high fidelity numerical simulations of the governing equations. Of particular interest is the so-called Direct Numerical Simulations (DNS) [1] of reacting and non-reacting turbulent flows wherein all turbulence scales are fully resolved. In the

DNS approach the time-dependent governing Navier-Stokes, energy and species continuity equations are solved with high accuracy, and all dynamically relevant ranges of unsteady spatial and temporal scales are numerically resolved. This stringent resolution criterion imposes prohibitive computational cost and consequently limits the attainable parametric range of relevant DNS. Even for moderate Reynolds number or number of transported species, for example, DNS requires massive supercomputers with hundreds of thousands of processing elements (PEs) working concurrently.

In some of the most well resolved DNS of complex flows [2–13] high-order finite difference schemes have been used extensively to approximate the spatial derivatives. The parallel efficiency of these schemes is high since they use local stencils extending only to the nearest grid-points to approximate the derivatives. However, in a data parallel decomposition of the domain, where multiple PEs are working on different parts of the computational domain in parallel, the PEs also need to communicate across the processor boundaries of the nearest computational domain neighbors in each direction. These communications proceed in the form of halo exchanges of processor boundary information that is stored in buffer/ghost points. For current state-of-the-art numerical solvers, the PEs synchronize and wait for these halo exchanges to complete before derivatives at PE boundaries are computed. If this synchronization is not imposed, data from the latest communication may not be available at the buffer points. However, if old or delayed data is used at the boundaries with the standard finite difference schemes, the resulting solution is inconsistent and inaccurate [14,15]. Thus, standard schemes inherently necessitate synchronizations at PE boundaries and consequently incur severe penalties due to processor idling, especially when a large number of PEs are used. This is the communication and synchronization bottleneck that is expected to pose a major challenge in efficiently scaling to next-generation exascale machines [16].

An efficient way to mitigate this bottleneck is to relax the strict communication and synchronization requirements and perform simulations asynchronously. Essentially the derivatives are computed with the latest available data which may or may not be at the current time level and thus a delay is encountered at the PE boundaries. However, due to the degraded numerical accuracy of standard schemes with asynchrony, alternate approaches are needed for asynchronous computations such that numerical errors incurred are minimal. While asynchronous computations have been utilized successfully for iterative linear solvers [17–20], some of the early work in asynchronous simulations of partial differential equations (PDEs) is limited to simple canonical equations in one-dimension. In [21–23] the governing equation is modified *apriori* to offset the effect of asynchrony on the numerical solution. However, extension of this approach to higher order and dimension is very challenging. An alternate approach to realize asynchronous computing is to modify the numerical scheme [14, 15,24,25]. An example of this approach would be the asynchrony-tolerant (AT) schemes of arbitrarily high orders that were used for asynchronous simulations of one-dimensional linear and non-linear equations in [24]. The statistical error analysis of these AT schemes presented in [24] shows that the error depends upon simulation parameters, such as the number of PEs and delays at the boundaries. In a more recent study, the AT schemes were used to perform accurate asynchronous simulations of Burgers' turbulence [26] and compressible turbulence [27] that also exhibited superior scaling to their synchronous counterpart. Two distinct algorithms, synchronization avoiding (SAA) and communication avoiding (CAA), to efficiently introduce asynchrony at boundaries in a three-dimensional flow solver were also presented and verified in [27]. The ability of the asynchronous method to absorb system noise effects was demonstrated in [26]. The stability and spectral accuracy of AT schemes was analyzed in detail in [28]. To further advance the applicability of AT schemes, multi-physics simulations such as turbulent combustion, requiring massive computations, are a natural next choice.

As a first step to evaluate the numerical performance of AT schemes for computationally expensive and highly nonlinear turbulent combustion simulations, several canonical reacting one-dimensional flow problems are performed. In particular, the effect of data asynchrony is studied for autoignition, premixed flame propagation and non-premixed autoignition. Both single-step and detailed chemical mechanisms with stiff reactions are used to test the efficacy of the AT schemes. Moreover, for simulations of flows involving shocks and discontinuities AT-WENO schemes are derived for the first time. These AT-WENO schemes are then demonstrated for their numerical performance in simulations of both reacting and non-reacting flow.

The remainder of the paper is organized as follows. In section 2 a brief overview of asynchronous computing and AT schemes is presented along with time integration using multi-stage Runge-Kutta schemes. The AT-WENO schemes are derived in section 3 followed by order of accuracy tests. The governing equations are included in section 4 and the numerical test cases and results are presented in section 5. Finally, sections 6 and 7 comprise discussions and conclusions, respectively. The AT schemes used in the simulations and expressions for the AT-WENO schemes are included in the Appendix.

2. Asynchronous computing method

2.1. Concept

To illustrate the asynchronous computing method proposed in [14,15], consider the simple one-dimensional time-dependent heat equation,

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}, \tag{1}$$

where u(x, t) is the temperature and α is the diffusion coefficient. This equation can be approximated using a first-order Euler and second-order central difference schemes for spatial and temporal derivatives, respectively, as
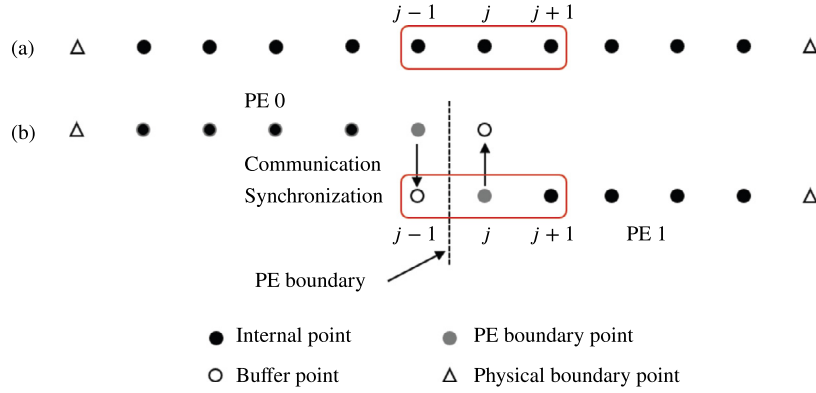
Fig. 1. A schematic of a one-dimensional discretized domain in (a) serial and in (b) parallel (bottom) with two processing elements. The red curve denotes the three-point stencil about point j and the dashed line denotes the processor boundary. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \alpha \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2} + \mathcal{O}(\Delta t, \Delta x^2). \tag{2}$$

Here $u_j^{n+1} = u(x_j, t_n)$ is the temperature at a grid point j and a time level n. This finite difference equation, for a given set of initial and boundary conditions, can be used to evolve the temperature field in a discretized one-dimensional domain (see Fig. 1(a)). Implementation of this numerical method in a serial code is straightforward, as function (u) values at the grid points and at necessary time levels would be available in the processing element's memory. On the other hand, in a parallel computing model with a distributed memory setting, the one-dimensional domain is decomposed into $\mathcal{P}$ sub-domains that are mapped to $\mathcal{P}$ processes/processing elements (PEs), as illustrated in Fig. 1(b) for $\mathcal{P} = 2$. Now, each PE's memory would contain a subset of function values that belong to the mapped sub-domain. The grid points in each sub-domain can be further divided into two sets: first, the interior points at which the function updates are computed with data available within the PE's memory, and second, the PE boundary points at which the function updates depend on the data from neighboring PEs. This is facilitated by, as mentioned earlier, the so-called halo exchanges which communicate the necessary function values from a neighboring PE into buffer or ghost points. Note that the function update using Eq. (2) at PE boundary points cannot proceed until the halo exchanges are complete. This is often ensured by imposing explicit synchronization that results in significant overheads.

In the asynchronous computing method [15], the halo exchanges are initiated, but no explicit synchronization is imposed. Therefore the function value at buffer points may or may not be at the latest time level n, which modifies the update equation in Eq. (2) for PE boundary points (assuming that the points are at the left boundary) to

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \alpha \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^{n-\tilde{k}}}{\Delta x^2} + \mathcal{O}(\Delta t/\Delta x^2, \Delta t, \Delta x^2), \tag{3}$$

where $\tilde{k}$ represents the time level delay resulting from the relaxed synchronization. If the communication at the latest time level is complete, then $\tilde{k} = 0$ and the computations at the PE boundary points are synchronous (like the interior points). Otherwise, $\tilde{k} > 0$ and the update equation would involve a delayed function value, resulting in *asynchronous computations*. Note that depending on the computation and communication time scales, delays at different PE boundaries can be different and can be several time levels older. Let $\mathcal{L}$ be the maximum allowable delay between the two PEs at a PE boundary. Then, the delay $\tilde{k} \in \{0, 1, 2, \ldots, \mathcal{L} - 1\}$. As the arrival of messages at PE boundaries can be modeled as a random process [29], the corresponding delay values, $\tilde{k}$, take on random values. Let p_k represent the probability of occurrence of a k time level delay, i.e. $\tilde{k} = k$, in a simulation. The sum of probabilities is $\sum_{k=0}^{\mathcal{L}-1} p_k = 1$.

It is evident that the numerical properties of standard schemes used asynchronously would not just depend on numerical and physical parameters such as grid resolution (Δx), time step (Δt) and diffusion coefficient (α), but would also depend on simulation parameters such as the number of PEs $(\mathcal{P})$, stencil size, delay probabilities (p_k) and maximum allowable delay $(\mathcal{L})$. A statistical framework was developed in [15] to assess the numerical properties of finite differences implemented with relaxed communication synchronizations. It was shown that asynchronous computations at the PE boundaries would not only affect the stability and consistency of the overall numerical method, but also significantly degrade the accuracy. For the schemes considered in Eq. (3), the truncation error consists of terms that are functions of $\Delta t/\Delta x^2$, Δt and Δx^2 to leading order. From stability analysis, which gives the relation $\Delta t \sim \Delta x^2$ for a constant CFL number, the leading order term is identified as $\mathcal{O}(\Delta t/\Delta x^2)$, which is zeroth-order accurate. Following the analysis described in [15], one can obtain the overall accuracy in the domain to be $\mathcal{O}(\Delta x)$ or first-order accurate. It can also be shown that the error would scale linearly with the number of PEs P and mean delay $\bar{\tilde{k}}$. As standard schemes would result in poor accuracy under relaxed data synchronization, new asynchrony-tolerant schemes were derived subsequently [24], which are summarized next.

2.2. Asynchrony-tolerant (AT) spatial schemes

The basic idea of asynchrony-tolerant (AT) schemes is to use an extended stencil in space and/or time to recover loss of accuracy due to the use of delayed data. For example, the spatial derivative of $u_j^n := u(j\Delta x, n\Delta t)$ at location j and time level n can be computed using

$$\frac{\partial u}{\partial x}\bigg|_j^n = \frac{1}{\Delta x} \sum_{\ell=0}^{\mathcal{L}-1} \sum_{m=-M_1(\ell)}^{M_2(\ell)} a_m^\ell u_{j+m}^{n-\ell}, \tag{4}$$

where Δx is the grid size, $M_1(\ell)$ and $M_2(\ell)$ are the extents of stencil to the left and right of u_j^n and $\mathcal{L}$ is the maximum allowable delay level. For a given set of parameters $\{\mathcal{L}, M_1(\ell), M_2(\ell)\}$ the unknown weights a_m^ℓ are obtained by solving a linear system of equations that is formed by imposing necessary accuracy constraints on terms of the Taylor series expansion of $u_{j+m}^{n-\ell}$ about u_j^n. If $\mathcal{L} = 1$ and $M_1(0) = M_2(0) = M$, this system of equations yields coefficients for the standard central difference scheme. For $\mathcal{L} > 1$, AT schemes can be derived if the stencil parameters $\{M_1(\ell), M_2(\ell)\}$ are appropriately chosen. An example of a second-order AT scheme with $\Delta t \sim \Delta x^2$ at the left processor boundary, *i.e.* delayed data is used only at the left stencil points, is

$$\frac{\partial u}{\partial x}\bigg|_j^n = \frac{\tilde{k} u_{j-1}^{n-\tilde{k}-1} - (\tilde{k}+1)u_{j-1}^{n-\tilde{k}} + u_{j+1}}{\Delta x} \tag{5}$$

where $\tilde{k}$ is the delay [24]. We note that the coefficients in Eq. (5) depend upon delays, unlike the fixed coefficients in standard schemes. Furthermore, in the absence of delays *i.e.* when $\tilde{k} = 0$, Eq. (5) reduces to the standard central difference scheme for the first derivative. The general methodology to derive different families of AT schemes of arbitrary order of accuracy is described in [24]. Here the AT schemes from [24] are directly used and are listed in the Appendix without details of the derivation.

2.3. Temporal scheme: Runge-Kutta

For a fully-discrete system, the global order of accuracy depends on both the spatial and temporal discretizations. In order to compute this global order, a stability relation of the form $\Delta t \sim \Delta x^r$ is used to express the leading order truncation error term of the time discretization scheme also in terms of grid size (Δx). For example, for a fourth-order spatial scheme ($\mathcal{O}(\Delta x^4)$), if a second-order temporal discretization scheme ($\mathcal{O}(\Delta t^2)$) is used then the global spatial order is two if $\Delta t \sim \Delta x$ and four if $\Delta t \sim \Delta x^2$. One of the most widely used schemes for high-order temporal discretization is the multi-stage Runge-Kutta (RK) method. An S-stage explicit RK scheme for an equation of the form $du/dt = f(u, t)$ over a time step Δt is given by

$$u^{n+1} = u^n + \Delta t \sum_{s=0}^{S} b_s K_s, \tag{6}$$

where the stages K_s are computed using

$$
\begin{aligned}
K_1 &= f(u^n, t^n) \\
K_s &= f\left(u^n + \Delta t \sum_{i=1}^{s-1}(a_{si}k_i), t^n + c_s\Delta t\right).
\end{aligned} \tag{7}
$$

Here a_{si}, b_s, c_s are the coefficients of the RK scheme. For an RK scheme, in advancing from time level n to $n+1$, intermediate stages K_s are computed. Each of these stages depends upon the previous stages as can be seen from Eq. (7). Thus, for simulations performed in parallel, RK schemes require PEs to communicate and synchronize after every stage in order to update the corresponding values at buffer points. However, if simulations are performed with asynchronous communications using AT schemes, then a fractional delay will be encountered at the intermediate stages and different AT schemes will be required for each stage. An alternate method is proposed to effectively use multi-stage RK schemes with AT that circumvents the need to communicate at every stage and avoids fractional delays.

In the presence of delays, each stage K_s is computed using AT schemes at PE boundary points. As mentioned before, the current stage K_s explicitly depends upon the previous stages, $K_{s-1}, K_{s-2}, \ldots, K_1$. Thus, to ensure accuracy all the stages of RK scheme are computed locally at both PE boundary and buffer points of each PE. Furthermore, at the old time levels required by the AT scheme, these stages are also computed at the internal points close to the PE boundary. In order to perform the additional computations at buffer points, a larger message of size (= No. of stages in RK $\times$ spatial stencil) has to be communicated across processors at every time step. However, the PEs no longer communicate at every stage of the RK. Thus, there exists a trade-off between communication which is expensive and computation which is relatively cheap.

To illustrate that the RK schemes preserve the order of accuracy when combined with high-order AT schemes, the one-dimensional diffusion equation in Eq. (1) is reconsidered. Denoting the right-hand-side of Eq. (1) by $f(u,t) = \partial^2 u/\partial x^2$ (excluding the diffusion coefficient, α), the temporal discretization at the j-th spatial point using a two-stage second-order RK scheme (RK2) is given by

$$u_j^{n+1} = u_j^n + \alpha \Delta t \left(\frac{K_{j,1}}{2} + \frac{K_{j,2}}{2} \right) \tag{8}$$

where $K_{j,1} = f(u_j^n, t^n)$, $K_{j,2} = f\left((u_j^n)^*, t^n + \Delta t \right)$ and $(u_j^n)^* = u_j^n + \Delta t(f(u_j^n, t^n)) = u_j^n + \Delta t \alpha K_{j,1}$. When a standard fourth-order central difference scheme is used for numerical approximation of $f(u_j^n, t^n)$, the following $K_{j,1}$ and $K_{j,2}$ are obtained,

$$K_{j,1} = \frac{-u_{j-2}^n + 16u_{j-1}^n - 30u_j^n + 16u_{j+1}^n - u_{j+2}^n}{12\Delta x^2}$$

$$K_{j,2} = -\frac{\alpha \Delta t (K_{j-2,1}) + u_{j-2}^n}{12\Delta x^2} + \frac{4\left(\alpha \Delta t (K_{j-1,1}) + u_{j-1}^n\right)}{3\Delta x^2} - \frac{5\left(\alpha \Delta t (K_{j,1}) + u_j^n\right)}{2\Delta x^2} \tag{9}$$

$$+ \frac{4\left(\alpha \Delta t (K_{j+1,1}) + u_{j+1}^n\right)}{3\Delta x^2} - \frac{\alpha \Delta t (K_{j+2,1}) + u_{j+2}^n}{12\Delta x^2}.$$

The Taylor series expansion of Eq. (8) on substituting Eq. (9) then yields a truncation error of the form

$$\text{TE}_j^n|_{\text{sync}} = \frac{1}{6}\left(-f^{(0,3)}(x,t)\right)\Delta t^2 - \frac{1}{90}\alpha f^{(6,0)}(x,t)\Delta x^4 + \mathcal{O}(\Delta x^6, \Delta t^3, \Delta x^4 \Delta t) \tag{10}$$

or

$$\text{TE}_j^n|_{\text{sync}} = \left(-\frac{r_\alpha^2 u^{(0,3)}(x,t)}{6\alpha^2} - \frac{1}{90}\alpha u^{(6,0)}(x,t)\right)\Delta x^4 + \mathcal{O}(\Delta x^6), \tag{11}$$

where $r_\alpha = \alpha \Delta t/\Delta x^2$ is the diffusive CFL number and the subscript "sync" denotes synchronous. Note that from Eq. (11) the global spatial order of accuracy is four.

When a fourth-order AT scheme is used at the boundary points for computation of spatial derivatives, we have

$$K_{j,1} = \left(\tilde{k}^2 + \tilde{k}\right)\frac{\left(-u_{j-2}^{n-\tilde{k}-2} + 16u_{j-1}^{n-\tilde{k}-2} - 30u_j^n + 16u_{j+1}^n - u_{j+2}^n\right)}{24\Delta x^2}$$

$$- \left(\tilde{k}^2 + 2\tilde{k}\right)\frac{\left(-u_{j-2}^{n-\tilde{k}-1} + 16u_{j-1}^{n-\tilde{k}-1} - 30u_j^n + 16u_{j+1}^n - u_{j+2}^n\right)}{12\Delta x^2} \tag{12}$$

$$+ \left(\tilde{k}^2 + 3\tilde{k} + 2\right)\frac{\left(-u_{j-2}^{n-\tilde{k}} + 16u_{j-1}^{n-\tilde{k}} - 30u_j^n + 16u_{j+1}^n - u_{j+2}^n\right)}{24\Delta x^2}$$

which is used to compute $K_{j,2} = f(u_j^n + \Delta t \alpha K_{j,1}, t^n + \Delta t)$ with $f(.,.)$ being evaluated using the fourth-order AT-scheme as well. Once again, the Taylor series expansion is used to obtain the truncation error,

$$\text{TE}_j^n|_{\text{AT},\tilde{k}} = -\frac{5}{24}\alpha \tilde{k}\left(\tilde{k}^2 + 3\tilde{k} + 2\right)u^{(0,3)}(x,t)\frac{\Delta t^3}{\Delta x^2} - \frac{1}{6}u^{(0,3)}(x,t)\Delta t^2 - \frac{1}{90}\alpha u^{(6,0)}(x,t)\Delta x^4 + \mathcal{O}(\tilde{k}^3\frac{\Delta t^3}{\Delta x}). \tag{13}$$

The above expression simplifies to give a global fourth-order truncation error term when r_α is substituted,

$$\text{TE}_j^n|_{\text{AT},\tilde{k}} = -\left(\frac{75\tilde{k}\left(\tilde{k}^2 + 3\tilde{k} + 2\right)r_\alpha^3 u^{(0,3)}(x,t) + 60r_\alpha^2 u^{(0,3)}(x,t) + 4\alpha^3 u^{(6,0)}(x,t)}{360\alpha^2}\right)\Delta x^4 + \mathcal{O}(\Delta x^6). \tag{14}$$

When the delay is zero i.e. $\tilde{k} = 0$, the truncation error in Eq. (14) reduces to $\text{TE}_j^n|_{\text{sync}}$ (Eq. (11)) which is the synchronous truncation error at the internal points. Therefore, RK schemes can be used with the AT-schemes without affecting the global order of accuracy. During a simulation the stencil data at older time levels for the buffer as well as for internal points are available, and hence standard synchronous schemes can be used to compute the sub-stages $K_{j,s}$ at these points for efficient implementation of RK with AT. Numerical simulations of linear equations were performed using this approach and the global order of accuracy was preserved. Here the order of accuracy tests is shown for RK schemes with AT-WENO schemes that are derived next (see Table 1).

3. Asynchrony-tolerant weighted essentially non-oscillatory schemes (AT-WENO)

Despite their advantages, central finite difference schemes are not always suitable for performing DNS of, for example, supersonic flows. In the presence of sharp gradients or discontinuities in density, temperature and composition due to detonations, shocks or high pressure flames, the central difference schemes are prone to numerical oscillations and instabilities. A well established numerical technique to accurately simulate flows with piece-wise smooth solutions between discontinuities is the so called weighted essentially non-oscillatory schemes (WENO) [30,31]. These schemes aim to achieve high-order accuracy at the smooth regions of the flow and resolve the discontinuities with minimal oscillations by automatically selecting the locally smoothest stencil. WENO schemes are considerably more expensive than regular central difference schemes due to the need to evaluate the same flux functions from multiple candidate stencils [32]. However, the numerical method is suitable for implementation in finite difference codes using stencil operations very similar to the central difference schemes. Also, the usual advantages of DNS using finite difference schemes including parallelism and scalability apply to the WENO schemes as well. Apart from DNS [33,12], the advantages of WENO schemes have also been demonstrated in implicit large eddy simulations (iLES) [34–36].

In this section, a brief overview of the standard WENO schemes is provided followed by a discussion on the effect of asynchrony on the standard WENO schemes (AS-WENO) and derivation of the asynchrony-tolerant WENO (AT-WENO) schemes for accurate numerical simulations when delays are observed at PE boundaries. Consider the one-dimensional (x-direction) version of the governing equations:

$$\frac{\partial Q}{\partial t} + \frac{\partial C}{\partial x} + \frac{\partial D}{\partial x} = S, \tag{15}$$

where Q is the solution vector, C is the vector comprising the convective flux terms, D is the viscous and molecular diffusion flux vector, and S is the vector of source terms. The exact form of terms in Q, C and D is defined in section 4. In the WENO framework, the terms contained in the D vector do not require any special treatment and are evaluated using central difference schemes (standard or AT). However, the convective flux terms in C have to be computed appropriately to ensure both stability and accuracy in regions near and at discontinuities. Specifically, the derivative of the convective flux $\partial C/\partial x$ at a point j is approximated using the flux at the edges, i.e., $\widehat{C}_{j+\frac{1}{2}}$ of a cell $\mathcal{I}_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$. The flux at the edges is computed using the WENO approximation procedure that can be carried out using interpolation or reconstruction, and accordingly, requires point values or cell averages of the fluxes. Note that the hat notation ($\widehat{}$) is used to denote the variables at cell edges. The derivative is then computed using the relation

$$\left.\frac{\partial C}{\partial x}\right|_j = \frac{\widehat{C}_{j+\frac{1}{2}} - \widehat{C}_{j-\frac{1}{2}}}{\Delta x} \tag{16}$$

for a uniform grid and the order depends upon the order of the numerical flux approximation $\widehat{C}_{j\pm\frac{1}{2}}$. When using finite-differences, the WENO approximation directly yields the fluxes at the edges in terms of the fluxes at the grid points. For stability, appropriate up-winding of fluxes is also required and is achieved through splitting the flux, for example using the local Lax-Friedrichs flux splitting methodology,

$$C = C^+ + C^-, \quad \text{where}$$
$$C^\pm = \frac{1}{2}(C \pm \lambda_{max} Q), \tag{17}$$

at every grid-point. The quantity λ_{max} in Eq. (17) is the maximum local wave propagation speed which can be computed as

$$\lambda_{max}|_j = \max\{(|u|)_{j-1}, (|u \pm c|)_{j-1}, (|u|)_j, (|u \pm c|)_j\}, \tag{18}$$

where u and c are the local flow velocity and the speed of sound, respectively. The derivative is then simply evaluated using

$$\left.\frac{\partial C}{\partial x}\right|_j = \frac{\widehat{C}^+_{j+\frac{1}{2}} - \widehat{C}^+_{j-\frac{1}{2}}}{\Delta x} + \frac{\widehat{C}^-_{j+\frac{1}{2}} - \widehat{C}^-_{j-\frac{1}{2}}}{\Delta x} \tag{19}$$

with an appropriate upwind stencil for approximating both positive and negative fluxes [12]. For computation of derivative in Eq. (19) using the WENO procedure, the first step is to approximate the fluxes at the edges ($j \pm 1/2$) through an interpolation or a reconstruction procedure. For simplicity, the approximation of these fluxes using an interpolation technique is discussed here in detail. However, the approximation through reconstruction that treats grid-point values as cell averages can also be performed following the steps detailed here. In order to do so, the primitive function defined in [30] is required.

For illustration, consider a specific WENO scheme that uses three sub-stencils each of which comprises three grid-points denoted by $S^{(i)}$, $i = \{0, 1, 2\}$, in Fig. 2 to approximate the value of a quantity $\hat{u}$ at cell edges, say $j + 1/2$, and time-level n. Taken together the larger stencil obtained by combining these $S^{(i)}$ contains five points. Since the approximation
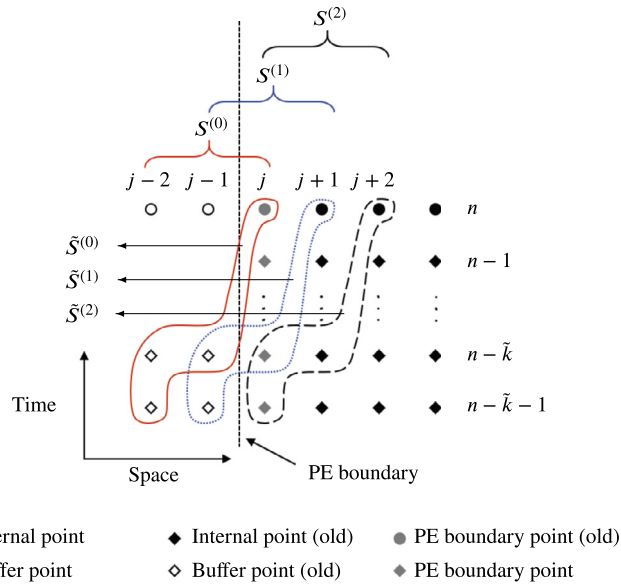
Fig. 2. Candidate stencils for asynchrony-tolerant WENO (AT-WENO) scheme for approximation at point $j+1/2$ and delays at left processor boundary: $\tilde{S}^{(0)}$, $\tilde{S}^{(1)}$ and $\tilde{S}^{(2)}$ in red, blue and black curves, respectively. $S^{(0)}$, $S^{(1)}$ and $S^{(2)}$ correspond to three-point stencils for standard WENO.

order depends upon the number of points, a fifth-order accurate approximation can be obtained with the full stencil for a smooth function. This is the classical three-fifth order WENO scheme, where the three smaller candidate stencils $S^{(0)} = \{j-2, j-1, j\}$, $S^{(1)} = \{j-1, j, j+1\}$, and $S^{(2)} = \{j, j+1, j+2\}$ give a degree two polynomial interpolation of $\hat{u}^n_{j+\frac{1}{2}}$ at time level n. The synchronous third-order interpolant in each of the three stencils can be computed using Taylor series expansion or Lagrange interpolation [30,31],

$$
\hat{u}^{n,(0)}_{j+\frac{1}{2}} = \frac{3}{8}u^n_{j-2} - \frac{5}{4}u^n_{j-1} + \frac{15}{8}u^n_j + \mathcal{O}(\Delta x^3)
$$
$$
\hat{u}^{n,(1)}_{j+\frac{1}{2}} = -\frac{1}{8}u^n_{j-1} + \frac{3}{4}u^n_j + \frac{3}{8}u^n_{j+1} + \mathcal{O}(\Delta x^3) \tag{20}
$$
$$
\hat{u}^{n,(2)}_{j+\frac{1}{2}} = \frac{3}{8}u^n_j + \frac{3}{4}u^n_{j+1} - \frac{1}{8}u^n_{j+2} + \mathcal{O}(\Delta x^3).
$$

The final approximation is taken as a convex combination of the above three third-order approximations, yielding a higher order interpolant in the larger stencil. Ideally, if $u(x,t)$ is smooth in the large stencil $S = \bigcup_{i=0}^{2} S^{(i)} = \{j-2, j-1, j, j+1, j+2\}$, a fifth-order approximation can be achieved using the following expression:

$$
\hat{u}^n_{j+\frac{1}{2}} = \sum_{i=0}^{2} \omega_i \hat{u}^{n,(i)}_{j+\frac{1}{2}}, \tag{21}
$$

where ω_i's are the non-linear weights

$$
\omega_i = \frac{\alpha_i}{\sum_{i=0}^{2} \alpha_i}, \quad \alpha_i = \frac{\gamma_i}{(\epsilon + \beta^{(i)})^2}. \tag{22}
$$

The non-linear weights in Eq. (22) satisfy $w_i \geq 0$ and $\sum_{i=0}^{2} \omega_i = 1$ and are related to the ideal or linear weights γ_i through the smoothness indicator $\beta^{(i)}$ [37,31,38]. A simple Taylor series expansion yields the ideal weights $\gamma = \{\frac{1}{16}, \frac{5}{8}, \frac{5}{16}\}$, such that

$$
\hat{u}^n_{j+\frac{1}{2}} = \frac{3}{128}u^n_{j-2} - \frac{5}{32}u^n_{j-1} + \frac{45}{64}u^n_j + \frac{15}{32}u^n_{j+1} - \frac{5}{128}u^n_{j+2} + \mathcal{O}(\Delta x^5). \tag{23}
$$

We now focus on the effect of data asynchrony on each of the individual approximations. Consider a point j at the left processor boundary with a delay $\tilde{k}$ encountered at each of the buffer points. The order of accuracy of the interpolants in the first two stencils is affected by this delay,

$$\hat{u}^{n,(0)}_{j+\frac{1}{2}} = \frac{3}{8}u^{n-\tilde{k}}_{j-2} - \frac{5}{4}u^{n-\tilde{k}}_{j-1} + \frac{15}{8}u^n_j + \mathcal{O}(\Delta t, \Delta x \Delta t, \Delta x^3)$$

$$\hat{u}^{n,(1)}_{j+\frac{1}{2}} = -\frac{1}{8}u^{n-\tilde{k}}_{j-1} + \frac{3}{4}u^n_j + \frac{3}{8}u^n_{j+1} + \mathcal{O}(\Delta t, \Delta x \Delta t, \Delta x^3).$$

(24)

More precisely, there are terms in the truncation error that depend upon Δt. On relating $\Delta t \sim \Delta x^r$, the interpolants with delays reduce to first order when $r = 1$ (convective CFL) and to second order when $r = 2$ (diffusive CFL). The convex combination of the three interpolants with ideal weights γ_i also has a leading order truncation term that scales as $\mathcal{O}(\Delta t)$, thereby degrading the order even in the full stencil, *i.e.*

$$\hat{u}^n_{j+\frac{1}{2}} = \frac{3}{128}u^{n-\tilde{k}}_{j-2} - \frac{5}{32}u^{n-\tilde{k}}_{j-1} + \frac{45}{64}u^n_j + \frac{15}{32}u^n_{j-1} - \frac{5}{128}u^n_{j+2} + \mathcal{O}(\Delta t, \Delta x \Delta t, \Delta t^2, \Delta x^2 \Delta t).$$

(25)

Clearly, the standard interpolants cannot be used in the presence of delays. A similar degradation in order was observed when standard finite difference schemes are used with asynchrony [15] for computation of spatial derivatives. Following the general methodology to overcome this loss of accuracy in [24], the asynchrony-tolerant WENO schemes (AT-WENO) can be derived. For this, the AT-interpolant for each of the substencils in the presence of delays is computed. When multiple time-levels are used, the interpolation is two-dimensional i.e. in space and time, and thus, requires a greater number of points to achieve a given order of accuracy. Since the Taylor series expansion is used to obtain the coefficients, a relation of the form $\Delta t \sim \Delta x^r$ is also needed to identify the overall low-order terms (in space) that should be eliminated for achieving the desired order of accuracy. A schematic representation to identify low-order terms which need to be subjected to constraints is illustrated in Figures 2 and 3 in [24]. For example, for a third-order accurate approximation, when $\Delta t \sim \Delta x$, six equations corresponding to terms of the order $\{1, \Delta t, \Delta x, \Delta t^2, \Delta t \Delta x, \Delta x^2\}$ are needed, and therefore, each candidate stencil comprises six points. Similarly, if $r = 2$ only four equations $\{1, \Delta t, \Delta x, \Delta x^2\}$ are required, and therefore, a four-point stencil is sufficient to achieve a third-order approximation. Since the stencil now depends upon the delays observed at the processor boundaries, the stencil itself is random because of the random nature of the delays. One can represent this stochastic stencil as $\tilde{S}$. For the first candidate stencil (as shown in Fig. 2), $\tilde{S}^{(0)} = \{u^{-\tilde{k}+n-1}_{j-2}, u^{n-\tilde{k}}_{j-2}, u^{n-\tilde{k}}_{j-1}, u^n_j\}$ with delay $\tilde{k}$ and $r = 2$, the AT interpolant is

$$\tilde{u}^{n,(0)}_{j+\frac{1}{2}} = \frac{7}{8}\tilde{k}u^{-\tilde{k}+n-1}_{j-2} + \frac{1}{8}\left(3 - 7\tilde{k}\right)u^{n-\tilde{k}}_{j-2} - \frac{5}{4}u^{n-\tilde{k}}_{j-1} + \frac{15}{8}u^n_j + \text{TE}^{(0)}.$$

(26)

The leading order truncation error in Eq. (26) term depends on the delay $\tilde{k}$ and is equal to

$$\text{TE}^{(0)} = \frac{-5}{4}\tilde{k}\dot{u}'(x,t)\Delta t \Delta x + \frac{5}{16}u'''(x,t)\Delta x^3 + \cdots \sim \mathcal{O}(\Delta x^3),$$

(27)

where $\dot{u} = \partial u/\partial t$ and $u' = \partial u/\partial x$ and $\Delta t \sim \Delta x^2$ gives the overall spatial order of accuracy to be $\mathcal{O}(\Delta x^3)$. When $\tilde{k} = 0$, Eq. (26) reduces to the standard third-order synchronous interpolant in Eq. (20). For the second stencil $\tilde{S}^{(1)} = \{u^{-\tilde{k}+n-1}_{j-1}, u^{-\tilde{k}}_{j-1}, u^n_j, u^n_{j+1}\}$, with delays considered only at the buffer point (u_{j-1}), the third-order interpolant is

$$\tilde{u}^{n,(1)}_{j+\frac{1}{2}} = \frac{1}{8}\tilde{k}u^{-\tilde{k}+n-1}_{j-1} + \frac{1}{8}\left(-\tilde{k} - 1\right)u^{n-\tilde{k}}_{j-1} + \frac{3}{4}u^n_j + \frac{3}{8}u^n_{j+1} + \text{TE}^{(1)},$$

(28)

where the leading order truncation error term

$$\text{TE}^{(1)} = -\frac{1}{16}u'''(x,t)\Delta x^3 + \cdots \sim \mathcal{O}(\Delta x^3)$$

(29)

is independent of delay $\tilde{k}$, unlike Eq. (27). Lastly, for stencil $\tilde{S}^{(2)} = \{u^n_j, u^n_{j+1}, u^n_{j+2}\} = S^{(2)}$, since all the points are within the PE, the standard interpolant *i.e.* $\tilde{u}^{n,(2)}_{j+\frac{1}{2}} = \hat{u}^{n,(2)}_{j+\frac{1}{2}}$ can readily be used. The full stencil is comprised of seven points $\tilde{S} = \bigcup_{i=0}^{2} \tilde{S}^{(i)} = \{u^{-\tilde{k}+n-1}_{j-2}, u^{n-\tilde{k}}_{j-2}, u^{-\tilde{k}+n-1}_{j-1}, u^{n-\tilde{k}}_{j-1}, u^n_j, u^n_{j+1}, u^n_{j+2}\}$. With this choice of stencil, at most, a fourth-order accurate approximation can be obtained for the full stencil since the degrees of freedom are sufficient only to eliminate seven low-order terms. Extension to higher orders would require extending the stencil in both space and time. This is explained in great detail in [24] for AT finite difference schemes.

On evaluation of the ideal weights for this larger stochastic stencil $\tilde{S}$ *i.e.*

$$\tilde{u}^n_{j+\frac{1}{2}} = \sum_{i=0}^{2} \gamma_i \tilde{u}^{n,(i)}_{j+\frac{1}{2}},$$

(30)

yields

$$\gamma = \left\{ 0, \frac{1}{2}, \frac{1}{2} \right\} \tag{31}$$

which, in turn, gives a fourth-order truncation error term in Eq. (30). These ideal weights eliminate the first stencil $\tilde{S}^{(0)}$ and reduce the full stencil to a four-point synchronous stencil in the absence of delays. If the second stencil is modified $\tilde{S}^{(1)} = \{u_{j-1}^{-\tilde{k}+n-1}, u_{j-1}^{n-\tilde{k}}, u_j^{n-\tilde{k}}, u_{j+1}^n \}$, the ideal weights are no longer positive, $\gamma = \left\{ -\frac{3}{2}, -\frac{5}{2}, 5 \right\}$. Further changing the third stencil to $\tilde{S}^{(2)} = \{u_j^{-\tilde{k}+n-1}, u_j^{n-\tilde{k}}, u_{j+1}^{n-\tilde{k}}, u_{j+2}^n \}$ such that all three stencils now have similar stencil structure and thus use four points as shown in Fig. 2, the ideal weights obtained are

$$\gamma = \left\{ \frac{3}{8}, \frac{5}{4}, -\frac{5}{8} \right\}. \tag{32}$$

This exercise shows that irrespective of the choice of asynchronous stencil with four points, at least one of the ideal or linear weights is non-positive for integer values of $\tilde{k}$. This non-convex nature of ideal weights leads to instabilities and oscillations [39]. The procedure to deal with the non-positive weights has been described in [39] and it involves splitting all the weights into positive and negative parts,

$$\tilde{\gamma}_i^+ = \frac{1}{2} \left(\gamma_i + \theta |\gamma_i| \right)$$
$$\tilde{\gamma}_i^- = \tilde{\gamma}_i^+ - \gamma_i \tag{33}$$

for $i = \{0, 1, 2\}$. The split ideal weights are then scaled by the parameters

$$\sigma^{\pm} = \sum_{i=1}^{3} \tilde{\gamma}_i^{\pm}; \quad \gamma_i^{\pm} = \tilde{\gamma}_i^+ / \sigma^{\pm}, \tag{34}$$

which are used to write the split polynomial interpolations

$$\left(\hat{u}_{j+\frac{1}{2}}^n \right)^{\pm} = \sum_{i=0}^{2} \gamma_i^{\pm} \hat{u}_{j+\frac{1}{2}}^{n,(i)}, \tag{35}$$

which is equivalent to replacing the ideal positive weight γ_i in Eq. (22) by the corresponding scaled split weight $\gamma_i^{\pm}$ [39]. However, this procedure does involve a series of additional computations to compute the derivative of the flux accurately.

For this particular choice of stencil with delayed data at the left boundaries, the reconstruction polynomial can also be derived using the primitive function [30]. For example, for a stencil of the form $\tilde{S}^{(i)} = \{u_{j-1+i}^{-\tilde{k}+n-1}, u_{j+i}^{n-\tilde{k}}, u_{j+1+i}^{n-\tilde{k}}, u_{j+1+i}^n \}$ where $i = \{0, 1, 2\}$, one obtains the following AT approximations,

$$\tilde{u}_{j+\frac{1}{2}}^{n,(i)} = \frac{1}{6} \left(3i^2 - 12i + 5 \right) \tilde{k} u_{i+j-2}^{-\tilde{k}+n-1} + \frac{1}{6} \left(-3i^2 \tilde{k} + 12i\tilde{k} - 5\tilde{k} + 3i^2 - 6i + 2 \right) u_{i+j-2}^{n-\tilde{k}}$$
$$+ \frac{1}{6} \left(-6i^2 + 18i - 7 \right) u_{i+j-1}^{n-\tilde{k}} + \frac{1}{6} \left(3i^2 - 12i + 11 \right) u_{i+j}^n. \tag{36}$$

The reconstruction polynomials in Eq. (36) reduce to the traditional candidate reconstruction polynomials in [38,30] when $\tilde{k} = 0$ for three-fifth order WENO scheme with ideal weights $\{1/10, 6/10, 3/10\}$.

As is evident from Eq. (22), the non-linear weights ω_i that are integral to the WENO formulation depend upon the smoothness indicators β_i defined in [30] as

$$\beta^{(i)} = \sum_{\ell=1}^{k} \Delta x^{2\ell-1} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \left(\frac{d^{\ell}}{dx^{\ell}} p^{(i)}(x) \right)^2 dx \tag{37}$$

where $p^{(i)}(x)$ is the polynomial of degree k (=2, in the example considered here) for stencil $S^{(i)}$. For a one-dimensional polynomial interpolation Eq. (37) can be easily evaluated to give a smoothness indicator for each stencil. However, for a simpler extension to an asynchronous stencil, Simpson's 3/8 rule is used to compute $\beta^{(i)}$,

$$\beta^{(i)} = \frac{\Delta x}{6} \left(\left(\sum_{\ell=1}^{k} \Delta x^{2\ell-1} \left(\frac{d^{\ell}}{dx^{\ell}} p^{(j)}(x) \Big|_{x=x_{j-\frac{1}{2}}} \right) \right)^2 + 4 \sum_{\ell=1}^{k} \Delta x^{2\ell-1} \left(\frac{d^{\ell}}{dx^{\ell}} p^{(j)}(x) \Big|_{x=x_{j-\frac{1}{6}}} \right)^2 \right.$$
$$\left. + 4 \sum_{\ell=1}^{k} \Delta x^{2\ell-1} \left(\frac{d^{\ell}}{dx^{\ell}} p^{(j)}(x) \Big|_{x=x_{j+\frac{1}{6}}} \right)^2 + \sum_{\ell=1}^{k} \Delta x^{2\ell-1} \left(\frac{d^{\ell}}{dx^{\ell}} p^{(j)}(x) \Big|_{x=x_{j+\frac{1}{2}}} \right)^2 \right), \tag{38}$$

such that the derivatives of the interpolant are computed numerically at each of $\{x_{j-\frac{1}{2}}, x_{j-\frac{1}{6}}, x_{j+\frac{1}{6}}, x_{j+\frac{1}{2}}\}$ using all the points in the stencil $S^{(i)}$. This yields the smoothness indicator expressed in the form

$$\beta^{(0)} = \frac{1}{3}\left(4\left(u^n\right)^2_{j-2} + \left(11u^n_j - 19u^n_{j-1}\right)u^n_{j-2} + 25\left(u^n\right)^2_{j-1} + 10\left(u^n\right)^2_j - 31u^n_{j-1}u^n_j\right)$$

$$\beta^{(1)} = \frac{1}{3}\left(4\left(u^n\right)^2_{j-1} + \left(5u^n_{j+1} - 13u^n_j\right)u^n_{j-1} + 13\left(u^n\right)^2_j + 4\left(u^n\right)^2_{j+1} - 13u^n_j u^n_{j+1}\right) \qquad (39)$$

$$\beta^{(2)} = \frac{1}{3}\left(10\left(u^n\right)^2_j + \left(11u^n_{j+2} - 31u^n_{j+1}\right)u^n_j + 25\left(u^n\right)^2_{j+1} + 4\left(u^n\right)^2_{j+2} - 19u^n_{j+1}u^n_{j+2}\right)$$

which is consistent with $\beta^{(j)}$ in the literature. Extending this to the asynchronous stencil, the smoothness indicator of the following form is obtained when a four-point stencil is used for all three candidate stencils,

$$\tilde{\beta}^{(0)} = \frac{1}{3}\left(\left(10\tilde{k}^2 - 11\tilde{k} + 4\right)\left(u^{n-\tilde{k}}_{j-2}\right)^2 + \left(-20\tilde{k}^2 u^{n-\tilde{k}-1}_{j-2} + \tilde{k}\left(11u^{n-\tilde{k}-1}_{j-2} + 31u^{n-\tilde{k}}_{j-1} - 20u^n_j\right) - 19u^{n-\tilde{k}}_{j-1}\right.$$

$$\left. + 11u^n_j\right)u^{n-\tilde{k}}_{j-2} + 10\tilde{k}^2\left(u^{n-\tilde{k}-1}_{j-2}\right)^2 + 25\left(u^{n-\tilde{k}}_{j-1}\right)^2 - 31u^n_j u^{n-\tilde{k}}_{j-1} + \tilde{k}u^{n-\tilde{k}-1}_{j-2}\left(20u^n_j - 31u^{n-\tilde{k}}_{j-1}\right) + 10\left(u^n_j\right)^2\right) \qquad (40)$$

$$\tilde{\beta}^{(1)} = \frac{1}{3}\left(\left(4\tilde{k}^2 - 5\tilde{k} + 4\right)\left(u^{n-\tilde{k}}_{j-1}\right)^2 + \left(-8\tilde{k}^2 u^{n-\tilde{k}-1}_{j-1} + \tilde{k}\left(5u^{n-\tilde{k}-1}_{j-1} + 13u^{n-\tilde{k}}_j - 8u^n_{j+1}\right) - 13u^{n-\tilde{k}}_j\right.$$

$$+ 5u^n_{j+1}\right)u^{n-\tilde{k}}_{j-1} + 4\tilde{k}^2\left(u^{n-\tilde{k}-1}_{j-1}\right)^2 + 13\left(u^{n-\tilde{k}}_j\right)^2 - 13u^n_{j+1}u^{n-\tilde{k}}_j \qquad (41)$$

$$\left. + \tilde{k}u^{n-\tilde{k}-1}_{j-1}\left(8u^n_{j+1} - 13u^{n-\tilde{k}}_j\right) + 4\left(u^n_{j+1}\right)^2\right)$$

$$\tilde{\beta}^{(2)} = \frac{1}{3}\left(\left(4\tilde{k}^2 - 11\tilde{k} + 10\right)\left(u^{n-\tilde{k}}_j\right)^2 + \left(-8\tilde{k}^2 u^{n-\tilde{k}-1}_j + \tilde{k}\left(11u^{n-\tilde{k}-1}_j + 19u^{n-\tilde{k}}_{j+1} - 8u^n_{j+2}\right) - 31u^{n-\tilde{k}}_{j+1} + 11u^n_{j+2}\right)\right.$$

$$\left. u^{n-\tilde{k}}_j + 4\tilde{k}^2\left(u^{n-\tilde{k}-1}_j\right)^2 + 25\left(u^{n-\tilde{k}}_{j+1}\right)^2 - 19u^n_{j+2}u^{n-\tilde{k}}_{j+1} + \tilde{k}u^{n-\tilde{k}-1}_j\left(8u^n_{j+2} - 19u^{n-\tilde{k}}_{j+1}\right) + 4\left(u^n_{j+2}\right)^2\right). \qquad (42)$$

Each of the $\tilde{\beta}^i$ listed in Eq. (40), Eq. (41), Eq. (42) reduce to the corresponding $\beta^{(j)}$ expressions in Eq. (39) when $\tilde{k} = 0$. With the approximations and smoothness indicators in the candidate stencil known along with the ideal weights, the derivative of the flux can then be computed using the standard WENO procedure.

For the choice of stencil discussed above, if the delay is zero then instead of achieving fifth-order approximation in the larger stencil, only fourth-order accuracy is obtained. This is explained as follows. The ideal weights $\{\gamma_0, \gamma_1, \gamma_2\}$ in Eq. (30) satisfy $\sum_{i=0}^2 \gamma_i = 1$ and thus, two additional constraints need to be imposed to find a unique solution when seeking a combination of smaller stencils. Upon upgrading from a third-order approximation in the smaller stencil $\tilde{S}^{(j)}$ to a fourth-order approximation in the full stencil $\tilde{S}$, only two lower order truncation error terms of the form $\{\Delta t \Delta x, \Delta x^3\}$ can be eliminated. Thus, a unique solution for γ_i is obtained. However, such an approximation will never recover fifth-order accuracy in the absence of delays since the Δx^4 truncation error term is non-zero. Consequently, when there is no delay, the large stencil $\tilde{S}$ will have at most four points and the approximation is only fourth-order accurate. For example, the ideal weights Eq. (32) yield non-zero coefficients only for points $\{j-2, j-1, j, j+2\}$ in the full stencil when $\tilde{k} = 0$. Moreover, this stencil is spatially discontinuous.

The reduction in the number of points in the larger stencil when the delay is zero and a corresponding lower order accurate approximation can be overcome by selectively eliminating additional truncation error terms. For an asynchronous fifth-order approximation, terms of the form $\{\Delta t^2, \Delta t \Delta x^2, \Delta x^4\}$ in addition to $\{\Delta t \Delta x, \Delta x^3\}$ also need to be eliminated. Since this yields a system with six constraints and only three unknowns, a unique solution cannot be determined. However, it is still possible to achieve a fifth-order approximation in the absence of delays. Addition of one more point in the smaller asynchronous stencil and using this degree of freedom to eliminate the $\Delta t \Delta x$ truncation term does not affect the order of the resulting interpolant. However, when the ideal weights in the large stencil are computed, one of the constraints can now be imposed on the Δx^4 term. This exercise then yields convex ideal weights that are exactly equal to the ones obtained when all three small stencils are synchronous. While the asynchronous approximation in the full stencil is still of order four, the synchronous approximation recovers fifth-order accuracy when the delay goes to zero. Moreover, a convex combination
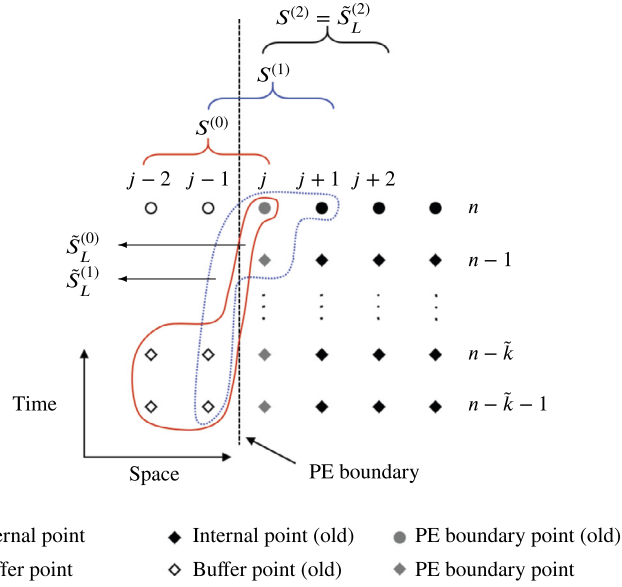
Fig. 3. Updated candidate stencil for asynchrony-tolerant WENO (AT-WENO) scheme for approximation at point $j + 1/2$ and delays at left processor boundary: $\tilde{S}_L^{(0)}$ and $\tilde{S}_L^{(1)}$ in red and blue curves, respectively. $S^{(0)}, S^{(1)}$ and $S^{(2)}$ correspond to three-point stencils for standard WENO. Here $\tilde{S}_L^{(2)}$ is same as $S^{(2)}$.

of the candidate stencil with none of the weights being negative is achieved. Thus, the additional computation involved in the treatment on negative weights is no-longer required. This new asynchronous stencil at the left boundary is given as

$$
\begin{aligned}
\tilde{S}_L^{(0)} &= \{u_{j-2}^{n-\tilde{k}-1}, u_{j-2}^{n-\tilde{k}}, u_{j-1}^{n-\tilde{k}-1}, u_{j-1}^{n-\tilde{k}}, u_j^n\}, \\
\tilde{S}_L^{(1)} &= \{u_{j-1}^{n-\tilde{k}-1}, u_{j-1}^{n-\tilde{k}}, u_j^{n-\tilde{k}}, u_j^n, u_{j+1}^n\} \\
\tilde{S}_L^{(2)} &= S^{(2)} = \{u_j^n, u_{j-1}^n, u_{j+2}^n\}.
\end{aligned}
\tag{43}
$$

and is shown in Fig. 3. For this stencil the third-order interpolation of the form

$$
\begin{aligned}
\tilde{u}_{j+\frac{1}{2}}^{n,(0)} &= -\frac{3}{8}\tilde{k}u_{j-2}^{-\tilde{k}+n-1} + \frac{5}{4}\tilde{k}u_{j-1}^{-\tilde{k}+n-1} + \frac{1}{8}\left(3\tilde{k}+3\right)u_{j-2}^{n-\tilde{k}} + \frac{1}{4}\left(-5\tilde{k}-5\right)u_{j-1}^{n-\tilde{k}} + \frac{15}{8}u_j^n \\
\tilde{u}_{j+\frac{1}{2}}^{n,(1)} &= \frac{1}{8}\tilde{k}u_{j-1}^{-\tilde{k}+n-1} + \frac{1}{8}\left(-\tilde{k}-1\right)u_{j-1}^{n-\tilde{k}} + \frac{3}{4}u_j^n + \frac{3}{8}u_{j+1}^n \\
\tilde{u}_{j+\frac{1}{2}}^{n,(2)} &= \frac{3}{8}8u_j^n + \frac{3}{4}u_{j+1}^n - \frac{1}{8}u_{j+2}^n
\end{aligned}
\tag{44}
$$

is obtained when a constraint is imposed on the truncation error term $\Delta t \Delta x$ for the first two stencils, $\tilde{S}^{(0)}$ and $\tilde{S}^{(1)}$. The ideal weights are then computed to be

$$
\gamma = \left\{\frac{1}{16}, \frac{5}{8}, \frac{5}{16}\right\}.
\tag{45}
$$

While the interpolation procedure for approximating the fluxes at the edges is explained in detail, approximation using reconstruction, which is equivalent to the WENO approximation for the first derivative, is the main relevant WENO procedure when designing finite volume or finite difference schemes to solve hyperbolic conservation laws [40]. Similar constraints in terms of the order-of-accuracy and elimination of specific terms can also be used to derive a reconstruction function with an asynchronous stencil using a primitive variable. For the choice of stencil listed in Eq. (43), by treating point values as cell-averages the third-order reconstruction approximation at the left boundary assumes the following functional form

$$
\begin{aligned}
\tilde{u}_{j+\frac{1}{2}}^{n,(0)} &= -\frac{1}{3}\tilde{k}u_{j-2}^{-\tilde{k}+n-1} + \frac{7}{6}\tilde{k}u_{j-1}^{-\tilde{k}+n-1} + \frac{1}{6}\left(2\tilde{k}+2\right)u_{j-2}^{n-\tilde{k}} + \frac{1}{6}\left(-7\tilde{k}-7\right)u_{j-1}^{n-\tilde{k}} + \frac{11}{6}u_j^n \\
\tilde{u}_{j+\frac{1}{2}}^{n,(1)} &= \frac{1}{6}\tilde{k}u_{j-1}^{-\tilde{k}+n-1} + \frac{1}{6}\left(-\tilde{k}-1\right)u_{j-1}^{n-\tilde{k}} + \frac{5}{6}u_j^n + \frac{1}{3}u_{j+1}^n \\
\tilde{u}_{j+\frac{1}{2}}^{n,(2)} &= \frac{1}{3}u_j^n + \frac{5}{6}u_{j+1}^n - \frac{1}{6}u_{j+2}^n
\end{aligned}
\tag{46}
$$

such that the coefficients reduce to their corresponding synchronous values when $\tilde{k} = 0$. The ideal or linear weights in this case can then be computed to be

$$\gamma = \left\{ \frac{1}{10}, \frac{3}{5}, \frac{3}{10} \right\} \tag{47}$$

which provides the fifth-order reconstruction polynomial in the larger stencil when the delay is zero, and is fourth-order, otherwise. Finally, the smoothness indicator for stencil Eq. (43) is computed using Simpson's 3/8 rule,

$$
\begin{aligned}
\tilde{\beta}^{(0)} = \frac{1}{3} \bigg(& 4 \left(u_{j-2}^{n-\tilde{k}} \right)^2 + 25 \left(u_{j-1}^{n-\tilde{k}} \right)^2 + \tilde{k}^2 \Big(4 \left(u_{j-2}^{n-\tilde{k}-1} \right)^2 + 4 \left(u_{j-2}^{n-\tilde{k}} \right)^2 + 19 u_{j-2}^{n-\tilde{k}} \left(u_{j-1}^{n-\tilde{k}-1} - u_{j-1}^{n-\tilde{k}} \right) \\
& + 25 \left(u_{j-1}^{n-\tilde{k}-1} - u_{j-1}^{n-\tilde{k}} \right)^2 + u_{j-2}^{n-\tilde{k}-1} \left(19 \left(u_{j-1}^{n-\tilde{k}} - u_{j-1}^{n-\tilde{k}-1} \right) - 8 u_{j-2}^{n-\tilde{k}} \right) \Big) - 31 u_j^n u_{j-1}^{n-\tilde{k}} + 10 \left(u_j^n \right)^2 \\
& + u_{j-2}^{n-\tilde{k}} \left(11 u_j^n - 19 u_{j-1}^{n-\tilde{k}} \right) + \tilde{k} \Big(8 \left(u_{j-2}^{n-\tilde{k}} \right)^2 + \left(19 u_{j-1}^{n-\tilde{k}-1} - 38 u_{j-1}^{n-\tilde{k}} + 11 u_j^n \right) u_{j-2}^{n-\tilde{k}} \\
& + u_{j-2}^{n-\tilde{k}-1} \left(-8 u_{j-2}^{n-\tilde{k}} + 19 u_{j-1}^{n-\tilde{k}} - 11 u_j^n \right) - \left(u_{j-1}^{n-\tilde{k}-1} - u_{j-1}^{n-\tilde{k}} \right) \left(50 u_{j-1}^{n-\tilde{k}} - 31 u_j^n \right) \Big) \bigg)
\end{aligned}
\tag{48}
$$

$$
\begin{aligned}
\tilde{\beta}^{(1)} = \frac{1}{3} \bigg(& 4 \tilde{k}^2 \left(u_{j-1}^{n-\tilde{k}-1} - u_{j-1}^{n-\tilde{k}} \right)^2 + 4 \left(u_{j-1}^{n-\tilde{k}} \right)^2 + 13 \left(u_j^n \right)^2 + 4 \left(u_{j+1}^n \right)^2 + \left(5 u_{j+1}^n - 13 u_j^n \right) u_{j-1}^{n-\tilde{k}} \\
& - 13 u_j^n u_{j+1}^n - \tilde{k} \left(u_{j-1}^{n-\tilde{k}-1} - u_{j-1}^{n-\tilde{k}} \right) \left(8 u_{j-1}^{n-\tilde{k}} - 13 u_j^n + 5 u_{j+1}^n \right) \bigg)
\end{aligned}
\tag{49}
$$

$$\tilde{\beta}^{(2)} = \frac{1}{3} \left(10 \left(u_j^n \right)^2 - 31 u_{j+1}^n u_j^n + 25 \left(u_{j+1}^n \right)^2 + 4 \left(u_{j+2}^n \right)^2 + \left(11 u_j^n - 19 u_{j+1}^n \right) u_{j+2}^n \right) = \beta^{(2)}. \tag{50}$$

In terms of the leading order truncation error, these smoothness indicators have the same properties as their synchronous counterparts, and therefore, preserve the order characteristics of the non-linear weights in Eq. (21). Thus, the AT-WENO approximation in the simulations presented here is carried out using the reconstruction polynomials in Eq. (46), ideal weights in Eq. (47) and the smoothness indicators listed in Eq. (50). While only a specific example of AT-WENO is presented here, similar steps can be followed to derive high-order schemes as well by extending the stencil in both space and time. Furthermore, additional low-order truncation error terms can also be selectively eliminated in both candidate and full stencil such that the AT-WENO scheme in the full stencil reduces to the corresponding synchronous scheme in the absence of delays.

Before proceeding with the validation simulations in Section 5, the order of accuracy of the synchronous and the new AT-WENO schemes is tested on a linear advection equation,

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0 \tag{51}$$

in a periodic domain $x \in [-1, 1]$ for an initial condition $u(x, 0) = \sin^4(\pi x + 0.25)$ and a convective velocity of $c = 5$ for different spatial resolutions. Since the AT-WENO scheme is derived using the relation $\Delta t \sim \Delta x^2$, we retain this power-law for computing the time step in the simulations to assess the order of accuracy. The temporal scheme used is RK-3 implemented using the procedure listed in section 2.3. Delays with uniform probability are introduced at every grid-point. The L_1 and L_∞ norm of the error at $t = 1.0$ is tabulated in Table 1 for the synchronous case, standard WENO used asynchronously (AS-WENO) and the asynchrony-tolerant WENO (AT-WENO). As expected, the order degrades to two when the standard WENO sees delays at the boundaries and the error is orders of magnitude larger than the synchronous error. On the contrary, the AT-WENO schemes exhibit errors comparable to the synchronous case and the order of accuracy is close to four.

The analysis was repeated for an inviscid Burgers equation,

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u^2}{2} \right) = 0 \tag{52}$$

in a periodic domain with an initial condition, $u(x, 0) = 0.5 + \sin(\pi x)$. The order of accuracy both before and after the formation of the shock is computed. At $t = 0.15$, the solution is smooth throughout the domain and the order deterioration only occurs when the standard WENO is used asynchronously (AS-WENO). Both synchronous and AT-WENO have an order higher than the theoretical order when the grid is refined. For L_1 and L_∞ norm of error after shock formation, only the smooth region at a distance of 0.05 on both sides of the shock, i.e. $|x - x_{\text{shock}}| > 0.05$, is considered [37]. Once again, errors in both the L_1 and L_∞ norms and the order of accuracy for AT-WENO are consistent with that for standard WENO. Thus, AT-WENO exhibits expected numerical accuracy for both linear and non-linear equations. (See Table 2.)

Table 1

Order of accuracy for linear advection equation $u_t + cu_x = 0$ with initial condition $u(x, 0) = \sin^4(\pi x + 0.25)$ at $t = 1.0$ for synchronous, AS-WENO and AT-WENO.

Case	N	L_1 error	L_1 order	L_∞ error	L_∞ order
	16	1.95×10^{-1}	-	3.93×10^{-1}	-
	32	3.87×10^{-2}	2.3	9.09×10^{-2}	2.1
Synchronous	64	2.79×10^{-3}	3.8	5.57×10^{-3}	4.0
(Order 5)	128	2.65×10^{-4}	3.4	1.07×10^{-3}	2.4
	256	1.09×10^{-5}	4.6	7.00×10^{-5}	3.9
	512	3.01×10^{-7}	5.8	2.18×10^{-6}	5.0
	16	3.25×10^{-1}	-	6.10×10^{-1}	-
	32	1.42×10^{-1}	1.2	2.93×10^{-1}	1.0
AS-WENO	64	2.84×10^{-2}	2.3	6.38×10^{-2}	2.2
$p = [0.5, 0.5]$	128	5.70×10^{-3}	2.3	1.32×10^{-2}	2.3
(Order 5)	256	1.22×10^{-3}	2.2	2.90×10^{-3}	2.2
	512	2.82×10^{-4}	2.1	6.79×10^{-4}	2.1
	16	2.00×10^{-1}	-	3.86×10^{-1}	-
	32	4.15×10^{-2}	2.3	1.00×10^{-1}	1.9
AT-WENO	64	2.98×10^{-3}	3.8	6.70×10^{-3}	3.9
$p = [0.5, 0.5]$	128	2.90×10^{-4}	3.4	9.54×10^{-4}	2.9
(Order 4)	256	1.42×10^{-5}	4.4	7.04×10^{-5}	3.8
	512	5.79×10^{-7}	4.6	2.40×10^{-6}	4.9
	16	3.01×10^{-1}	-	5.09×10^{-1}	-
	32	6.18×10^{-2}	2.3	1.37×10^{-1}	1.9
AT-WENO	64	5.20×10^{-3}	3.8	1.13×10^{-2}	3.6
$p = [0.3, 0.3, 0.3]$	128	4.22×10^{-4}	3.6	1.12×10^{-3}	3.3
(Order 4)	256	2.40×10^{-5}	4.1	7.73×10^{-5}	3.9
	512	1.26×10^{-6}	4.3	2.86×10^{-6}	4.8

4. Governing equations

In the following sections the effect of asynchrony is evaluated for a set of canonical reacting flow configurations. The AT schemes described in Section 3 are implemented in a compressible reacting flow solver with periodic and open boundaries. The governing conservation equations and constitutive laws are presented in this section. The one-dimensional form of the conservation equations for mass, momentum, total energy and species are

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\frac{\partial (\rho u)}{\partial x} \\
\frac{\partial (\rho u)}{\partial t} &= -\frac{\partial (\rho u u)}{\partial x} + \frac{\partial \tau}{\partial x} - \frac{\partial P}{\partial x} \\
\frac{\partial (\rho e_0)}{\partial t} &= -\frac{\partial [u(\rho e_0 + P)]}{\partial x} + \frac{\partial (\tau u)}{\partial x} - \frac{\partial q}{\partial x} \\
\frac{\partial (\rho Y_i)}{\partial t} &= -\frac{\partial (\rho u Y_i)}{\partial x} - \frac{\partial (\rho Y_i V_i)}{\partial x} + W_i \dot{\omega}_i,
\end{aligned}
\tag{53}
$$

where Y_i is the mass fraction, W_i is the molecular weight, V_i is the species mass diffusion, $\dot{\omega}_i$ is the molar production rate of species i and e_0 is the specific total energy

$$
e_0 = \frac{u^2}{2} - \frac{P}{\rho} + h.
\tag{54}
$$

$h = \sum_{i=1}^{N_s} Y_i h_i = \sum_{i=1}^{N_s} Y_i \left(h_i^0 + \int_{T_0}^{T} c_{p,i} dT \right)$ is the total enthalpy expressed in terms of h_i^0 which is the enthalpy of formation of species i at temperature T_0 and the isobaric heat capacity $c_p = \sum_{i=1}^{N_s} Y_i c_{p,i}$. For an ideal gas mixture, $P = \rho R_u T / W$ and $c_p - c_v = R_u / W$, where $W = \left(\sum_{i=1}^{N_s} Y_i / W_i \right)^{-1}$ and R_u is the universal gas constant, are used to compute the pressure and specific heats. The viscous stress τ is

$$
\tau = \frac{4}{3} \mu \frac{\partial u}{\partial x},
\tag{55}
$$

and the heat flux and species diffusion velocities are given by

Table 2
Order of accuracy for inviscid Burgers' equation $u_t + (u^2/2)_x = 0$ with initial condition $u(x, 0) = 0.5 + \sin(\pi x)$ before and after the shock for synchronous, AS-WENO and AT-WENO.

$t = 0.15$: Before shock

Case	N	L_1 error	L_1 order	L_∞ error	L_∞ order
	32	3.03×10^{-4}	-	2.36×10^{-3}	-
Synchronous	64	1.14×10^{-5}	4.7	1.08×10^{-4}	4.4
(Order 5)	128	4.38×10^{-7}	4.7	3.71×10^{-6}	4.9
	256	1.41×10^{-8}	5.0	2.12×10^{-7}	4.1
	512	3.37×10^{-10}	5.4	4.53×10^{-9}	5.5
	32	7.99×10^{-4}	-	4.77×10^{-3}	-
AS-WENO	64	1.44×10^{-4}	2.5	7.21×10^{-4}	2.7
$p = [0.5, 0.5]$	128	3.39×10^{-5}	2.1	1.49×10^{-4}	2.3
(Order 5)	256	8.61×10^{-6}	2.0	3.76×10^{-5}	2.0
	512	2.05×10^{-6}	2.1	8.93×10^{-6}	2.1
	32	3.11×10^{-4}	-	2.48×10^{-3}	-
AT-WENO	64	1.21×10^{-5}	4.7	1.15×10^{-4}	4.4
$p = [0.5, 0.5]$	128	4.85×10^{-7}	4.6	4.14×10^{-6}	4.8
(Order 4)	256	1.82×10^{-8}	4.7	2.07×10^{-7}	4.3
	512	6.61×10^{-10}	4.8	4.82×10^{-9}	5.4
	32	3.23×10^{-4}	-	2.67×10^{-3}	-
AT-WENO	64	1.32×10^{-5}	4.6	1.23×10^{-4}	4.4
$p = [0.3, 0.3, 0.3]$	128	6.14×10^{-7}	4.4	4.88×10^{-6}	4.7
(Order 4)	256	2.74×10^{-8}	4.5	1.99×10^{-7}	4.6
	512	1.25×10^{-9}	4.5	7.42×10^{-9}	4.7

$t = 0.55$: After shock

Case	N	L_1 error	L_1 order	L_∞ error	L_∞ order
	32	6.26×10^{-3}	-	1.20×10^{-1}	-
	64	2.43×10^{-4}	4.7	1.14×10^{-2}	3.4
Synchronous	128	6.09×10^{-6}	5.3	3.00×10^{-4}	5.2
(Order 5)	256	1.83×10^{-7}	5.1	2.18×10^{-5}	3.8
	512	2.25×10^{-9}	6.3	3.79×10^{-7}	5.8
	32	6.49×10^{-3}	-	1.20×10^{-1}	-
AS-WENO	64	3.21×10^{-4}	4.3	1.10×10^{-2}	3.4
$p = [0.5, 0.5]$	128	3.17×10^{-5}	3.3	3.48×10^{-4}	5.0
(Order 5)	256	6.96×10^{-6}	2.2	1.75×10^{-5}	4.3
	512	1.70×10^{-6}	2.0	4.14×10^{-6}	2.1
	32	6.26×10^{-3}	-	1.20×10^{-1}	-
AT-WENO	64	2.44×10^{-4}	4.7	1.14×10^{-2}	3.4
$p = [0.5, 0.5]$	128	6.10×10^{-6}	5.3	2.99×10^{-4}	5.2
(Order 4)	256	1.85×10^{-7}	5.0	2.18×10^{-5}	3.8
	512	2.43×10^{-9}	6.2	3.78×10^{-7}	5.8
	32	6.26×10^{-3}	-	1.20×10^{-1}	-
AT-WENO	64	2.44×10^{-4}	4.7	1.14×10^{-2}	3.4
$p = [0.3, 0.3, 0.3]$	128	6.15×10^{-6}	5.3	2.99×10^{-4}	5.3
(Order 4)	256	1.89×10^{-7}	5.0	2.18×10^{-5}	3.8
	512	2.80×10^{-9}	6.1	3.76×10^{-7}	5.9

$$q = -\lambda \frac{\partial T}{\partial x} + \sum_{i=1}^{N_s} h_i J_i$$
$$V_i = -\frac{D_i^{\mathrm{mix}}}{X_i} \frac{\partial X_i}{\partial x}$$

(56)

where $J_i = \rho Y_i V_i$ is the species diffusive flux, D_i^{mix} is the mixture-averaged diffusion coefficient, and $X_i = Y_i W / W_i$ is the mole fraction. Barodiffusion and the Soret and Dufour effects are not considered. CHEMKIN [41] and TRANSPORT [42] software libraries were linked with the solver and used for evaluating reaction rates, thermodynamic and mixture-averaged transport properties. Eq. (53) can be written in a compact form as in Eq. (15) where

Table 3
Probability of simulated delays for different sets used in the numerical experiments presented in Section 5. This legend (color and/or symbol) is used in all the figures from Fig. 4 to Fig. 13.

	Probability $[p_0\ p_1\ p_2]$	Legend
Set-1	[0.8 0.1 0.1]	—o
Set-2	[0.6 0.3 0.1]	−.*
Set-3	[0.4 0.5 0.1]	− − □
Set-4	[1.0 0.0 0.0]	... +

Table 4
Summary of numerical simulations and their relevance.

Case number	Case name	Relevant processes to be resolved
5.1	Acoustic wave propagation (non-reacting)	Pressure perturbation travelling at speed of sound
5.2	Auto-ignition of H_2 (periodic domain)	Spontaneous ignition dominated by reaction term
5.3	Auto-ignition of C_2H_4 (temperature fluctuations at inflow)	Unsteadiness, oscillatory ignition front
5.4	Premixed flame propagation	Reactive-diffusive balance in reaction zone
5.5	Non-premixed ignition	Diffusion controlled reaction front
5.6	Propagation of a detonation wave	Jump due to shock front followed by a reaction zone

$$Q = \begin{pmatrix} \rho \\ \rho u \\ \rho e_0 \\ \rho Y_i \end{pmatrix}, \quad C = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ u(\rho e_0 + P) \\ \rho u Y_i \end{pmatrix}, \quad D = \begin{pmatrix} 0 \\ -\tau \\ -\tau u + q \\ \rho Y_i V_i \end{pmatrix}, \text{ and } \quad S = \begin{pmatrix} 0 \\ 0 \\ 0 \\ W_i \dot{\omega}_i \end{pmatrix}. \tag{57}$$

The derivatives of product terms in Eq. (53) are expanded using the chain rule. For example, $\partial(\rho u Y_i)/\partial x := (\rho u Y_i)_x = \rho u (Y_i)_x + \rho u_x Y_i + \rho_x u Y_i$. This essentially allows computation of derivatives of such terms using AT schemes without having to retain every product term at multiple time-levels. Apart from the expansion using the chain rule, the one-dimensional asynchronous solver used in the present study is largely based on S3D [3] which is widely used to perform DNS of turbulent combustion.

5. Numerical results

In this section, five different flow configurations are selected to assess the effect of asynchrony on canonical combustion problems. For the first case, two types of asynchronous simulations are considered,

1. Standard schemes used asynchronously (AS-SFD or AS, AS-WENO as applicable)
2. AT schemes for asynchronous computation (AT, AT-WENO as applicable).

For the remainder of the cases, only the asynchronous simulation performed using AT schemes is compared with the synchronous simulation. The domain is decomposed into P processors and delays are introduced using a random number generator at each processor boundary similar to [24]. The maximum allowed delay levels are three with the probability of non-zero delay gradually increasing from Set-1 to Set-3. The different probability sets considered in numerical simulations are tabulated in Table 3 where Set-4 represents a synchronous simulation. The probability for Set-2 and Set-3 is similar to the probability of delays observed on TACC supercomputers [27]. A summary of all the numerical experiments performed and their relevance is listed in Table 4.

5.1. Non-reacting case: acoustic wave propagation

For this non-reacting case the propagation of an acoustic wave in air is considered. The effect of the source of the acoustic wave, for example from an ignition kernel, is decoupled from its propagation, and the only focus here is on whether the asynchrony-tolerant framework can accurately capture waves traversing with the speed of sound. Furthermore, if an error in gradients due to delayed data at the boundaries manifest themselves in the form of dilatational modes then one should observe these effects in the propagation of acoustic waves through larger errors. In the current problem the acoustic wave is considered to be a perturbation in the pressure field that is described with the following initial condition [43],
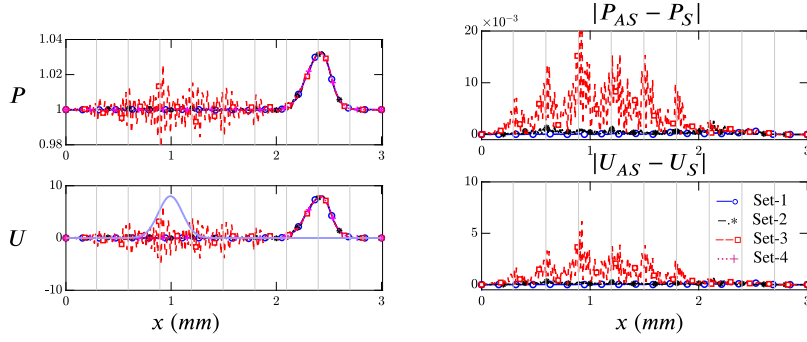
Fig. 4. AS-SFD: instantaneous profiles and errors in pressure (atm) and velocity (ms^{-1}) at $t = 4 \times 10^{-6}s$. The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3. The light blue line indicates the initial condition and gray vertical lines represent processor boundaries. AS: Asynchronous with standard finite-difference schemes, S: synchronous.
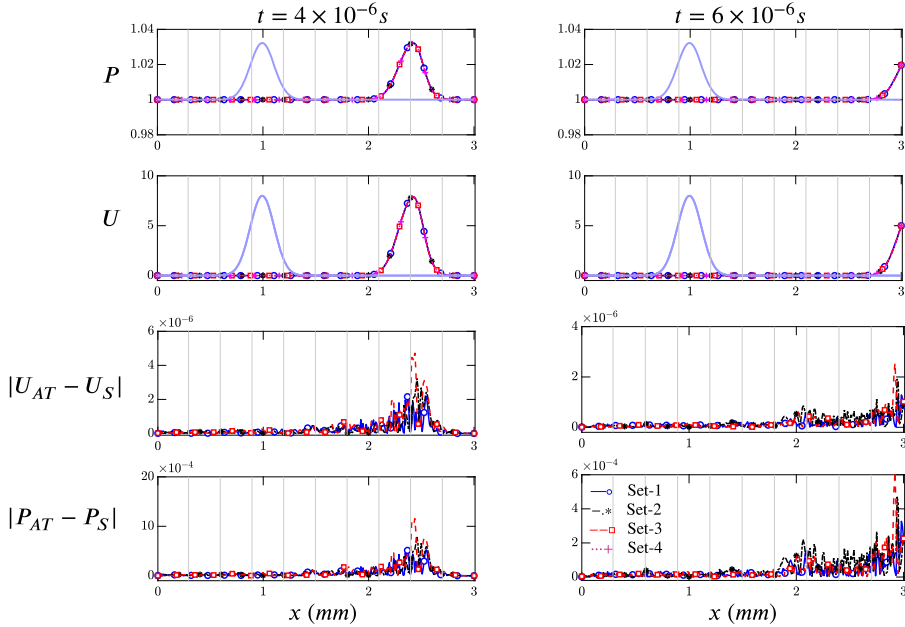

Fig. 5. Asynchrony-tolerant (AT): instantaneous profiles and errors in pressure (atm) and velocity (ms^{-1}) at $t = 4 \times 10^{-6}s$ (left) and $t = 6 \times 10^{-6}s$ (right). The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3. The light blue line indicates the initial condition and gray vertical lines represent processor boundaries. AT: asynchronous with asynchrony-tolerant, S: synchronous.

$$u(x, 0) = u_0 + \mathcal{A} \exp \left[-\mathcal{B} \left(\frac{x - x_0}{L} \right) \right]$$

$$P(x, 0) = P_0 + \rho_0 c_0 (u - u_0)$$

$$\rho(x, 0) = \rho_0 + \frac{\rho_0 (u - u_0)}{c_0}$$

$$T = P/\rho R$$

(58)

where u_0, ρ_0, P_0 prescribe the uniform mean values, and ideal gas law is used to compute the temperature field. Here $\mathcal{A}$ and $\mathcal{B}$ determine the magnitude and stiffness of the acoustic fluctuation and x_0 is the location of the fluctuation peak. This initial field is shown as a light blue line in Fig. 4 and Fig. 5. Non-reflecting inflow/outflow boundary conditions [44,45] are used and the initial fluctuation is allowed to traverse across at least one processing element boundary where it encounters delays. While at the internal points standard fourth-order central difference schemes are used, at the physical boundary points the derivatives are computed using second-order finite difference schemes. For computation of derivatives at the processor boundaries, fourth-order AT schemes are used (see Table A.13 in Appendix A).

When the standard central-difference schemes are used asynchronously (AS-SFD), there are visible fluctuations in the pressure field even at early times. This is evident from Fig. 4 where both instantaneous pressure and velocity fields at time

Table 5

L_1 and L_∞ norms of error in velocity and pressure in acoustic wave propagation simulations.

|		$t = 4 \times 10^{-6} s$		$t = 6 \times 10^{-6} s$	
AS-SFD					
	Case	$U\ (ms^{-1})$	$P\ (atm)$	$U\ (ms^{-1})$	$P\ (atm)$
	1	3.33×10^{-2}	1.36×10^{-4}	3.30×10^{-2}	1.34×10^{-4}
L_1 error	2	9.87×10^{-2}	4.00×10^{-4}	3.78×10^{-1}	1.52×10^{-3}
	3	8.01×10^{-1}	3.26×10^{-3}	2.28×10^{1}	9.31×10^{-2}
	1	2.27×10^{-1}	9.43×10^{-4}	2.72×10^{-1}	1.08×10^{-3}
L_∞ error	2	6.53×10^{-1}	2.69×10^{-3}	2.03	8.08×10^{-3}
	3	6.16	2.57×10^{-2}	1.45×10^{2}	7.56×10^{-1}
Asynchrony-tolerant (AT)					
	Case	$U\ (ms^{-1})$	$P\ (atm)$	$U\ (ms^{-1})$	$P\ (atm)$
	1	5.88×10^{-5}	2.43×10^{-7}	3.38×10^{-5}	1.40×10^{-7}
L_1 error	2	6.70×10^{-5}	2.75×10^{-7}	4.82×10^{-5}	1.95×10^{-7}
	3	8.83×10^{-5}	3.64×10^{-7}	3.81×10^{-5}	1.57×10^{-7}
	1	5.25×10^{-4}	2.18×10^{-6}	3.27×10^{-4}	1.29×10^{-6}
L_∞ error	2	7.84×10^{-4}	3.23×10^{-6}	4.70×10^{-4}	1.92×10^{-6}
	2	1.16×10^{-3}	4.73×10^{-6}	6.09×10^{-4}	2.56×10^{-6}
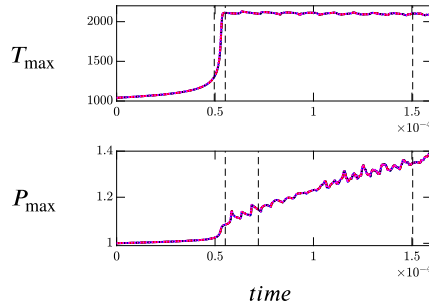
Fig. 6. Temporal evolution of the peak temperature (K) and pressure (atm) during auto-ignition of a premixture of hydrogen/air in a periodic domain. The different lines (without symbols): blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3. The dashed line indicates the time at which the instantaneous scalar and velocity profiles and errors are listed in Table 6 and Table 7.

$t = 4 \times 10^{-6} s$ show large numerical errors and clear deviation from the corresponding synchronous field. For larger delay values corresponding to Set-3 in Table 3, the numerical perturbations become significant at much shorter times and render the simulation unstable eventually. These errors are amplified even after the acoustic wave leaves the domain. However, when AT schemes are used for the propagation of an acoustic wave with delays at processor boundaries, the solution remains in close agreement with its synchronous counterpart and the errors are several orders of magnitude smaller (see Fig. 5). Similar behavior is also observed at later times when the wave is close to leaving the right boundary. Moreover, the errors in both pressure and velocity are not localized to near processing element boundary points where AT schemes are used to compute derivatives.

The L_1 and L_∞ norms of the errors computed with respect to the synchronous simulation are defined as

$$Z_{L_1} = \langle Z_{\text{synchronous}} - Z_{\text{AS or AT}} \rangle$$
$$Z_{L_\infty} = \max \left(Z_{\text{synchronous}} - Z_{\text{AS or AT}} \right),$$

(59)

where Z is any quantity, for example temperature, pressure or mass fractions, and $\langle . \rangle$ is the spatial average. These errors are listed in Table 5 for both AS-SFD and AT simulations at two different times. The AT schemes exhibit significantly less numerical error at both times, and the error does not grow monotonically with time like it does for the AS-SFD.

5.2. Premixed auto-ignition of H_2 using one-step chemistry

A canonical problem of practical relevance to compression ignition in internal combustion engines is auto-ignition of a premixed fuel-air mixture at constant volume. Here auto-ignition of a fuel lean H_2/air mixture at an equivalence ratio of 0.4895 at an initial temperature of $400K$ is considered. A hot spot represented by a Gaussian temperature spike with a peak value of $1040K$, which is above the ignition limit, is introduced at the center of a one-dimensional closed domain and the
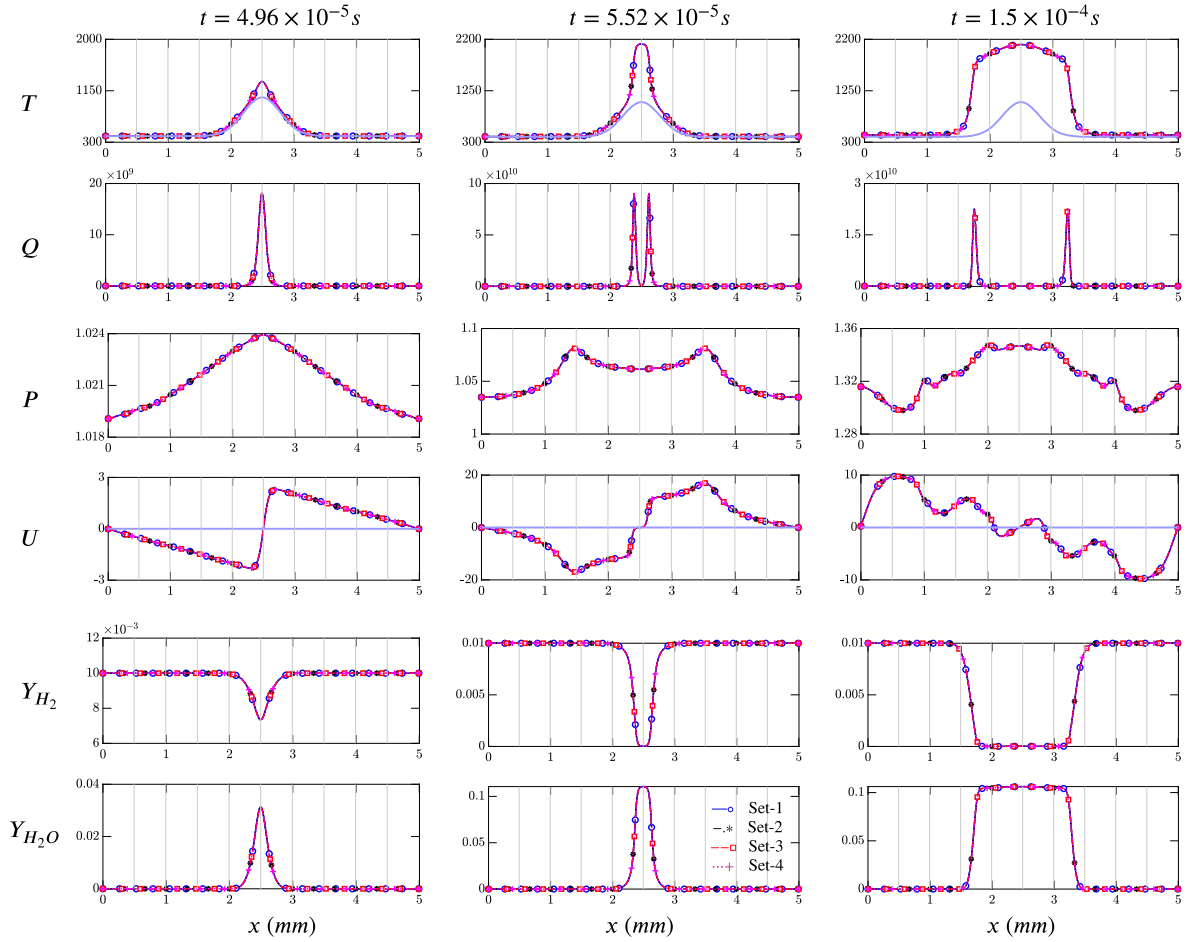
Fig. 7. Asynchrony-tolerant (AT): auto-ignition of a premixed H_2/air mixture: instantaneous profiles of temperature (K), heat-release rate ($Jm^{-3}s^{-3}$), pressure (atm), velocity (ms^{-1}), and mass fractions of Y_{H_2} and Y_{H_2O}. The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3. The light blue line indicates the initial condition and the gray vertical lines represent processor boundaries.

mixture eventually auto-ignites after an induction period. This thermal explosion leads to a rapid increase in temperature to nearly 2000K. Following the ignition of the hot spot, reaction fronts emanate from the spontaneous ignition kernel, one propagating to the left, while the other to the right. The gas expansion from the heat release during auto-ignition results in an induced velocity from an initially quiescent flow field. The time elapsed from the initial hot-spot to a time at which the temporal gradient of temperature or heat release rate is maximum is the so-called ignition delay time (τ_{ign}). The time evolution of maximum temperature and pressure is shown in Fig. 6. It is clear that while the temperature increases slowly initially, there is an exponential increase in the temperature at around $t = 5 \times 10^{-5}s$. The temporal evolution of the peak temperature and pressure is captured accurately by the asynchrony-tolerant schemes. Following ignition, the two fronts propagate towards the left and right boundary and are allowed to traverse across at least one processing element boundary where delays are explicitly encountered. For simplicity we consider a periodic domain that allows us to compute spectral characteristics of both thermodynamic and hydrodynamic quantities. Note that the periodic boundary conditions considered here provide a closed volume with compression heating from the exothermic reactions that lead to a gradual rise in pressure with time. This is evident from Fig. 6 where the peak pressure increases from 1 atm to 1.4 atm after ignition. The oscillatory nature observed in the peak pressure, and to a lesser extent in the peak temperature, is due to the propagation of compression waves in the closed domain that originated from the initial localized spontaneous ignition kernels.

To investigate the effect of delays at processor boundaries on important quantities of interest both before and after spontaneous ignition, the spatial evolution of temperature, velocity, pressure, heat release, and mass fractions of hydrogen and water are shown in Fig. 7. The time instants considered are indicated with dashed-black lines in Fig. 6. Qualitatively, no difference exists between the instantaneous values of these quantities for synchronous simulation and asynchronous simulations with AT schemes. The L_1 and L_∞ norm of error computed with respect to the synchronous simulation is listed in Table 6 and Table 7, respectively, for three time instants. Also listed are the errors when the standard schemes are used
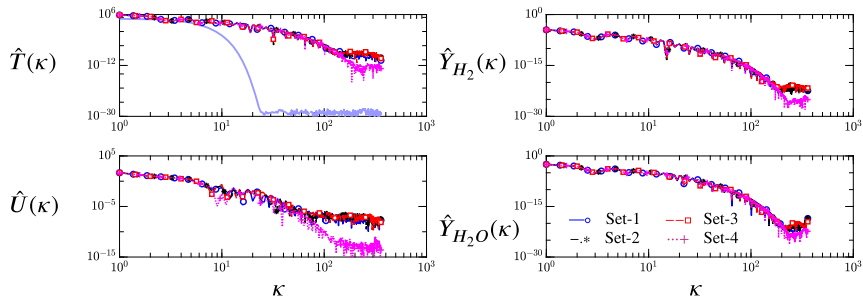
Fig. 8. AS-SFD: spectra of temperature ($\hat{T}(\kappa)$), velocity ($\hat{U}(\kappa)$), and mass fractions $\hat{Y}_{H_2}(\kappa)$ and $\hat{Y}_{H_2O}(\kappa)$ for auto-ignition of a premixed H_2/air mixture. The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3.

Fig. 9. Asynchrony-tolerant (AT): spectra of temperature ($\hat{T}(\kappa)$), velocity ($\hat{U}(\kappa)$), and mass fractions $\hat{Y}_{H_2}(\kappa)$ and $\hat{Y}_{H_2O}(\kappa)$ for auto-ignition of premixed H_2. The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3.
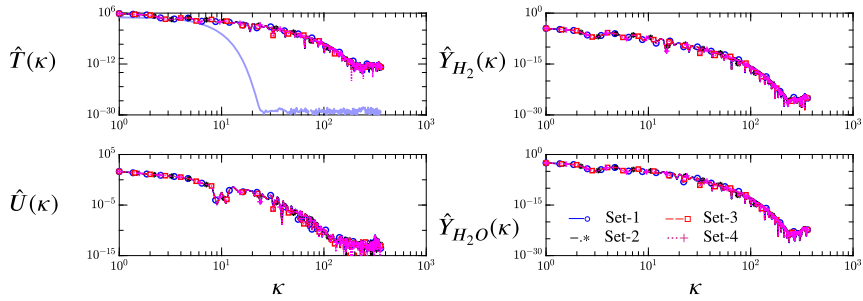
Table 6
L_1 norm of error in temperature, velocity, and mass fraction of H_2 and H_2O for auto-ignition of premixed hydrogen.

Time	Case	U (ms^{-1})	P (atm)	T (K)	H_2	H_2O
AS-SFD						
	1	2.4294×10^{-1}	8.1737×10^{-4}	3.2713×10^{-1}	1.2700×10^{-6}	2.5152×10^{-5}
1.5e-4	2	3.9893×10^{-1}	1.3420×10^{-3}	5.3992×10^{-1}	2.1059×10^{-6}	4.1778×10^{-5}
	3	5.6643×10^{-1}	1.9061×10^{-3}	7.6244×10^{-1}	2.9054×10^{-6}	5.8324×10^{-5}
Asynchrony-tolerant (AT)						
	1	1.07×10^{-7}	3.16×10^{-10}	8.50×10^{-8}	1.89×10^{-13}	2.29×10^{-12}
4.96e-5	2	1.23×10^{-7}	3.68×10^{-10}	8.19×10^{-8}	2.96×10^{-13}	2.31×10^{-12}
	3	1.33×10^{-7}	4.21×10^{-10}	1.11×10^{-7}	4.22×10^{-13}	3.66×10^{-12}
	1	6.80×10^{-7}	2.18×10^{-9}	7.14×10^{-7}	2.49×10^{-12}	2.53×10^{-11}
7.2e-5	2	6.46×10^{-7}	2.20×10^{-9}	9.74×10^{-7}	3.77×10^{-12}	4.89×10^{-11}
	3	1.07×10^{-6}	3.52×10^{-9}	1.09×10^{-6}	2.79×10^{-12}	3.67×10^{-11}
	1	1.17×10^{-4}	2.70×10^{-7}	7.67×10^{-4}	4.56×10^{-9}	4.87×10^{-8}
1.5e-4	2	5.67×10^{-5}	1.49×10^{-7}	2.52×10^{-4}	1.54×10^{-9}	1.61×10^{-8}
	3	1.10×10^{-4}	2.70×10^{-7}	7.17×10^{-4}	4.33×10^{-9}	4.62×10^{-8}

asynchronously. Note that in this case the errors are several orders of magnitude larger than AT schemes, irrespective of the quantity. Moreover, for the former case, the maximum error in temperature is approximately 9K which can trigger reactions and result in nonphysical ignition in otherwise quiescent flow due to the strong temperature dependence of reaction rates. Such numerical errors are not observed for AT schemes even when the gradients, for example, in temperature or species mass fractions exist at the processing element boundaries.

The spectra of temperature, velocity and mass fraction of reactants and products, are shown in Fig. 8 and Fig. 9. The AT schemes exhibit excellent agreement with the synchronous simulations at all wavenumbers (κ). Moreover, non-physical accumulation of energy at high wavenumbers due to numerical errors does not exist which is in fact observed in Fig. 8 when the standard schemes are used with asynchrony. Hence, AT schemes demonstrate an excellent resolving efficiency in both physical and spectral space despite delayed data being used for the computation of derivatives at processing element boundaries.

Table 7

L_∞ norm of error in temperature, velocity, and mass fraction of H_2 and H_2O for auto-ignition of premixed hydrogen.

Time	Case	U (ms^{-1})	P (atm)	T (K)	H_2	H_2O
AS-SFD						
1.5e-4	1	8.73×10^{-1}	2.72×10^{-3}	3.50	1.80×10^{-5}	3.19×10^{-4}
	2	1.51	4.46×10^{-3}	5.78	2.96×10^{-5}	5.25×10^{-4}
	3	2.14	6.23×10^{-3}	8.79	4.41×10^{-5}	7.88×10^{-4}
Asynchrony-tolerant (AT)						
2e-5	1	1.30×10^{-6}	2.44×10^{-9}	1.48×10^{-6}	4.61×10^{-12}	6.60×10^{-11}
	2	1.64×10^{-6}	2.30×10^{-9}	2.07×10^{-6}	1.07×10^{-11}	9.12×10^{-11}
	3	1.83×10^{-6}	2.80×10^{-9}	1.18×10^{-6}	1.19×10^{-11}	9.79×10^{-11}
5.52e-5	1	7.55×10^{-6}	2.64×10^{-8}	9.52×10^{-6}	7.27×10^{-11}	8.87×10^{-10}
	2	5.70×10^{-6}	1.95×10^{-8}	1.88×10^{-5}	1.36×10^{-10}	1.90×10^{-9}
	3	9.45×10^{-6}	3.26×10^{-8}	1.19×10^{-5}	9.30×10^{-11}	1.10×10^{-9}
1.5e-4	1	1.08×10^{-3}	2.44×10^{-6}	1.14×10^{-2}	5.51×10^{-8}	9.58×10^{-7}
	2	4.08×10^{-4}	7.97×10^{-7}	3.60×10^{-3}	1.77×10^{-8}	3.05×10^{-7}
	3	8.06×10^{-4}	1.75×10^{-6}	1.07×10^{-2}	5.12×10^{-8}	8.93×10^{-7}

Table 8

Asynchrony-tolerant (AT): L_1 norm of error in temperature, velocity, and mass fraction of OH and CH_2O for auto-ignition of C_2H_4/air flame with temperature fluctuations at the inflow.

Time	Case	U (ms^{-1})	P (atm)	T (K)	OH	CH_2O	H_2O
Asynchrony-tolerant (AT)							
4.48e-4	1	3.20×10^{-8}	1.10×10^{-10}	1.32×10^{-7}	6.67×10^{-13}	3.53×10^{-13}	4.48×10^{-12}
	2	3.89×10^{-8}	1.27×10^{-10}	1.70×10^{-7}	8.75×10^{-13}	4.82×10^{-13}	5.91×10^{-12}
	3	4.97×10^{-8}	1.66×10^{-10}	1.75×10^{-7}	9.06×10^{-13}	4.90×10^{-13}	6.08×10^{-12}
9.98e-4	1	3.09×10^{-9}	1.09×10^{-11}	3.32×10^{-8}	1.50×10^{-13}	9.12×10^{-14}	1.16×10^{-12}
	2	3.99×10^{-9}	1.38×10^{-11}	1.93×10^{-8}	1.20×10^{-13}	4.42×10^{-14}	6.33×10^{-13}
	3	4.77×10^{-9}	1.69×10^{-11}	2.82×10^{-8}	1.69×10^{-13}	7.69×10^{-14}	8.95×10^{-13}

5.3. Auto-ignition: temperature fluctuations at the inflow boundary

The effect of turbulent fluctuations on fuel-air mixing, wrinkling of flames or spontaneous ignition front propagation is central to turbulent combustion. To simulate this effect in one-dimension and to ensure that the AT schemes can propagate fluctuations accurately across processing element boundaries, a temperature perturbation is forced at the left inflow boundary. The initial condition, shown by the light blue line in Fig. 10, is the steady state solution obtained from the auto-ignition of premixed C_2H_4/air mixture at a pressure of 2 *atm*. A 22 species 18-step reduced mechanism describing the oxidation kinetics of ethylene/air [46] is used. The steady solution is perturbed with sinusoidal temperature fluctuations with a magnitude of $180K$ and frequency $20Khz$ at the left boundary using an oscillatory inflow boundary condition [47,48]. As the perturbations approach the igniting front, the adjacent temperature increases as is evident from the increase in the mass fraction of CH_2O radical in Fig. 10. This induces a secondary ignition kernel to the left of the initial front as can be seen in Fig. 10 (left column). The two kernels eventually interact and as time progresses, a steady ignition front develops to the left of the initial kernel as shown in the plots in the right column of Fig. 10. This front oscillates about a mean location, with the peak temperature and pressure also oscillating in response to the incoming sinusoidal fluctuations. The AT schemes accurately capture the transient and steady state evolution of both the temperature and intermediate species even in the presence of delays at the processor boundaries. An excellent qualitative agreement between the instantaneous profiles for AT and synchronous simulations in Fig. 10 is observed. Both L_1 and L_∞ norms of the errors tabulated in Table 8 and Table 9 are also reasonably small at both times.

5.4. Premixed flame propagation

The previous cases focused on investigation of the effect of asynchrony on spontaneous ignition dominated by unsteadiness and advection-reaction balance. The present test case is intended to study the effect of delayed data on laminar premixed flame propagation. The flame propagates at a subsonic velocity and is characterized by a balance between the reactive and diffusive terms in the steady species conservation equations. A mixture of C_2H_4/air at an equivalence ratio, $\phi = 0.42$, $T = 500\ K$ and $P = 1.72\ atm$ is considered. The one-dimensional domain is 4 *mm* and is discretized with 576 grid points that are distributed across 12 processors. The 22 species, 18-step reduced mechanism from [46] for ethylene/air used in the earlier case is also used here. The initial condition is generated using an auto-ignition case, and non-reflecting inflow and subsonic outflow are imposed at the left and right boundaries, respectively. The flame is initially located close to the

Table 9

Asynchrony-tolerant (AT): L_∞ norm of error in temperature, velocity, and mass fraction of OH and CH_2O for auto-ignition of C_2H_4/air flame with temperature fluctuations at the inflow.

Time	Case	U (ms^{-1})	P (atm)	T (K)	OH	CH_2O	H_2O
Asynchrony-tolerant (AT)							
	1	2.10×10^{-7}	8.22×10^{-10}	1.08×10^{-6}	5.76×10^{-12}	4.36×10^{-12}	5.12×10^{-11}
4.48e-4	2	2.20×10^{-7}	9.66×10^{-10}	1.53×10^{-6}	9.46×10^{-12}	6.58×10^{-12}	7.44×10^{-11}
	3	3.46×10^{-7}	1.30×10^{-9}	1.33×10^{-6}	8.23×10^{-12}	5.76×10^{-12}	6.54×10^{-11}
	1	2.05×10^{-8}	7.70×10^{-11}	5.33×10^{-7}	2.81×10^{-12}	2.15×10^{-12}	2.58×10^{-11}
1.9e-3	2	2.78×10^{-8}	1.01×10^{-10}	3.13×10^{-7}	1.68×10^{-12}	1.26×10^{-12}	1.50×10^{-11}
	3	3.57×10^{-8}	1.28×10^{-10}	4.41×10^{-7}	2.42×10^{-12}	1.83×10^{-12}	2.14×10^{-11}
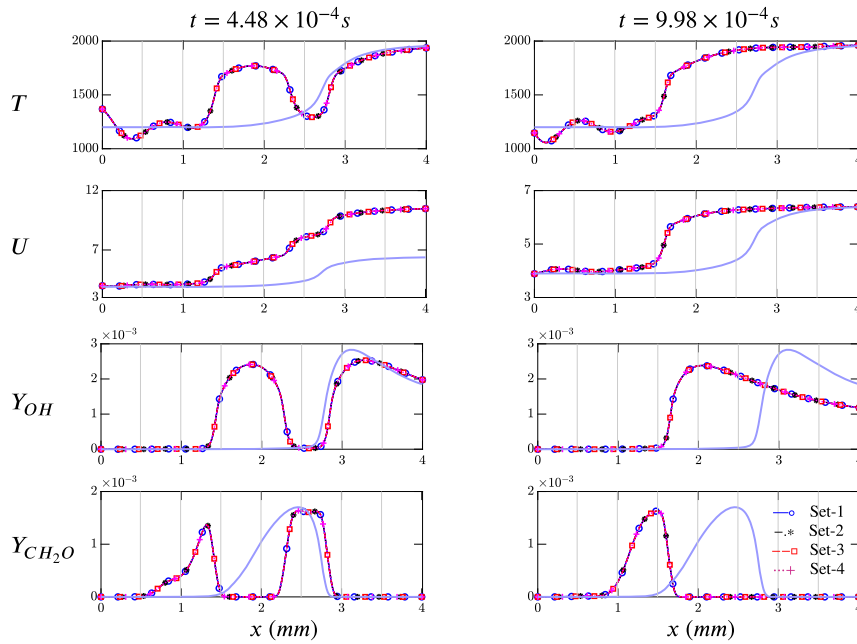
Fig. 10. Asynchrony-tolerant (AT): instantaneous profiles of temperature (K), velocity (ms^{-1}), and mass fractions of OH and CH_2O radicals for autoignition of premixed C_2H_4/air mixtures with temperature fluctuations at the inflow. The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3. The light blue line indicates the initial condition and gray vertical lines represent processor boundaries.
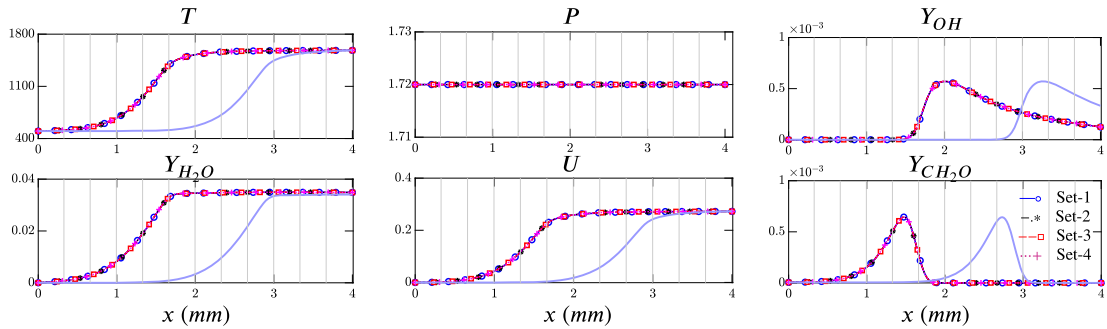

Fig. 11. Asynchrony-tolerant (AT): instantaneous profiles of temperature (K), pressure (atm), velocity (ms^{-1}), and mass fractions of product, H_2O, and radicals, OH and CH_2O, at $t = 1 \times 10^{-2}s$ for premixed C_2H_4 flame propagation. The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3. The light blue line indicates the initial condition and gray vertical lines represent processor boundaries.

Table 10

Asynchrony-tolerant (AT): L_1 and L_∞ norm of error in velocity, pressure, temperature, and mass fractions of OH, CH_2O and H_2O for premixed C_2H_4 flame propagation.

Time	Case	U (ms^{-1})	P (atm)	T (K)	OH	CH_2O	H_2O
L_1 error							
	1	6.22×10^{-11}	2.32×10^{-13}	2.72×10^{-9}	2.51×10^{-15}	3.21×10^{-15}	8.60×10^{-14}
1e-2	2	9.96×10^{-11}	3.81×10^{-13}	7.12×10^{-10}	8.73×10^{-16}	8.04×10^{-16}	2.04×10^{-14}
	3	9.42×10^{-11}	3.61×10^{-13}	1.27×10^{-9}	1.16×10^{-15}	1.35×10^{-15}	3.42×10^{-14}
L_∞ error							
	1	5.72×10^{-10}	2.07×10^{-12}	1.42×10^{-8}	3.02×10^{-14}	4.66×10^{-13}	2.38×10^{-14}
1e-2	2	1.06×10^{-9}	3.07×10^{-12}	4.18×10^{-9}	8.81×10^{-15}	1.23×10^{-13}	6.31×10^{-15}
	3	6.52×10^{-10}	3.36×10^{-12}	6.61×10^{-9}	1.42×10^{-14}	2.09×10^{-13}	1.19×10^{-14}
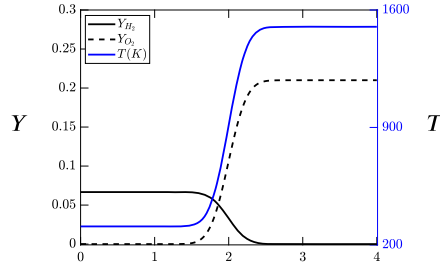
Fig. 12. Initial condition for non-premixed ignition with diluted fuel on the left and vitiated air on the right.

right edge of the domain and in time propagates to the left while consuming the reactants mixture. The initial flame front, comprised of preheat and reaction zones, spans across three PEs as shown by the light blue line Fig. 11. This flame traverses across multiple processing element boundaries and the errors are computed to assess the effect of the asynchronous data encountered at the boundaries on flow and flame quantities.

The flame structure is invariant with time, therefore, only the instantaneous profiles and errors at the time instant when the flame approaches near the left boundary is considered for accuracy analysis. There is excellent agreement between the spatial profiles for the synchronous and the AT simulations, as shown in Fig. 11 for temperature, velocity as well as major and minor species. The drop in pressure in the reaction zone is negligibly small and thus the pressure values depicted in Fig. 11 are nearly constant. As is evident from Table 10, both L_1 and L_∞ norms of the error between synchronous and AT simulations are negligibly small and as low as $\mathcal{O}(10^{-14})$ for some of the species. Furthermore, the flame speed computed from the time evolution of the location of the peak heat release rate is equal to 12.59 cm/s for both synchronous and asynchronous simulations. The thermal flame thickness is equal to $\delta_T = \frac{T_2-T_1}{\max(\partial T/\partial x)} = 7.81 \times 10^{-4} m$, irrespective of the delays.

5.5. Non-premixed ignition

An ignition of non-premixed H_2 using Burke's mechanism [49] with 9 species is considered next. The setup is similar to the one used in [43] and [50] with fuel diluted with nitrogen on the left and air heated to 1500K on the right, as shown in Fig. 12. An inflow boundary condition from [47] is used on the left boundary with zero velocity and an outflow boundary condition is used on the right boundary with appropriate viscous conditions on each side. The one-dimensional domain is 4 mm in length, discretized into 576 grid-points that are distributed across 12 PEs.

As time progresses and diffusion process tends to homogenize the gradients, there is ignition close to the stoichiometric mixture fraction, resulting in a rise in peak temperature and concentration of intermediate species. The instantaneous profiles of key quantities are shown in Fig. 13 at two times. The first series of plots on the left column in Fig. 13 are at an earlier time when ignition is localized. With time both fuel and intermediate species diffuse across processing element boundaries and the flame expands, which is shown in the plots on the right column in Fig. 13. The evolution of both major and minor species as well as temperature and velocity is accurately resolved by the AT schemes. The average and maximum errors, tabulated in Table 11 and Table 12, respectively, are both negligibly small. The errors were similarly negligible for other species that are not shown here.

5.6. Propagation of a detonation wave

To test the numerical accuracy of AT-WENO schemes for reacting flows, a detonation of stoichiometric hydrogen/oxygen mixture diluted with argon is considered. The mixture with molar ratio $H_2 : O_2 : Ar = 2 : 1 : 7$ is at an initial temperature of 305K and pressure of 6670 Pa. A detailed hydrogen/air mechanism [51] comprised of 9 species and 34 reversible reactions

Table 11
Asynchrony-tolerant (AT): L_1 norm of error in temperature, velocity, and mass fractions of H_2, O, H and H_2O for non-premixed H_2/air ignition.

Time	Case	U (ms^{-1})	P (atm)	T (K)	H_2	O	H	H_2O
Asynchrony-tolerant (AT)								
7.92e-5	1	1.94×10^{-8}	4.43×10^{-11}	1.52×10^{-7}	6.04×10^{-12}	3.63×10^{-12}	4.85×10^{-13}	2.54×10^{-11}
	2	2.59×10^{-8}	5.94×10^{-11}	1.97×10^{-7}	6.72×10^{-12}	5.40×10^{-12}	6.60×10^{-13}	3.74×10^{-11}
	3	3.14×10^{-8}	6.44×10^{-11}	3.13×10^{-7}	1.12×10^{-11}	7.77×10^{-12}	9.38×10^{-13}	5.42×10^{-11}
3.19e-4	1	5.94×10^{-10}	1.14×10^{-12}	2.57×10^{-8}	6.04×10^{-13}	2.69×10^{-13}	1.95×10^{-14}	4.43×10^{-12}
	2	5.25×10^{-10}	1.06×10^{-12}	3.45×10^{-8}	6.98×10^{-13}	2.55×10^{-13}	2.22×10^{-14}	5.91×10^{-12}
	3	6.18×10^{-10}	1.30×10^{-12}	5.01×10^{-8}	1.61×10^{-12}	2.56×10^{-13}	4.91×10^{-14}	4.80×10^{-12}

Table 12
Asynchrony-tolerant (AT): L_∞ norm of error in temperature, velocity, and mass fractions of H_2, O, H and H_2O for non-premixed H_2/air ignition.

Time	Case	U (ms^{-1})	P (atm)	T (K)	H_2	O	H	H_2O
Asynchrony-tolerant (AT)								
7.92e-5	1	1.09×10^{-7}	1.91×10^{-10}	6.10×10^{-7}	3.89×10^{-11}	1.31×10^{-10}	2.63×10^{-11}	4.05×10^{-12}
	2	1.11×10^{-7}	2.25×10^{-10}	8.52×10^{-7}	4.83×10^{-11}	1.85×10^{-10}	3.69×10^{-11}	5.24×10^{-12}
	3	1.21×10^{-7}	2.14×10^{-10}	1.25×10^{-6}	7.16×10^{-11}	2.73×10^{-10}	5.27×10^{-11}	7.10×10^{-12}
7.92e-5	1	5.08×10^{-9}	9.80×10^{-12}	9.89×10^{-8}	2.38×10^{-12}	1.65×10^{-11}	1.50×10^{-12}	7.53×10^{-14}
	2	4.01×10^{-9}	8.79×10^{-12}	8.94×10^{-8}	2.49×10^{-12}	1.64×10^{-11}	1.29×10^{-12}	6.47×10^{-14}
	3	4.93×10^{-9}	7.91×10^{-12}	1.52×10^{-7}	5.15×10^{-12}	1.41×10^{-11}	1.17×10^{-12}	1.38×10^{-13}

is used. A similar setup has also been used as a test case in previous studies to investigate the efficacy of numerical schemes [52–54]. A supersonic outflow condition is applied at the left boundary while a subsonic inlet condition is applied at the right boundary. The simulation is initialized with a shock that ignites the mixture at the left end of the computational domain. The domain length is 30 cm which is discretized into 6000 grid-points that are distributed across 100 PEs.

The simulation results are shown in Fig. 14 with pressure at different time instants in the leftmost plot. There is a leading shock front that compresses the fuel/air mixture followed by the induction and the reaction zones. The pressure obtained with the synchronous WENO scheme is shown in solid magenta. It is in excellent agreement with the pressure computed with the AT-WENO scheme shown in black circles and dashed lines. A zoomed-in view of the detonation structure depicted by temperature and pressure is shown in the centre plots of Fig. 14. The mixture reacts after the induction zone leading to an increase in temperature and a corresponding decrease in pressure. The mass fractions for different radicals including fuel H_2 are plotted on the rightmost plot in Fig. 14. The results of this simulation show that the detonation structure is captured accurately with the AT-WENO scheme. Furthermore, the detonation velocity obtained from standard WENO and AT-WENO simulations is 1619.7 ms^{-1} and 1620.1 ms^{-1}, respectively. These values are in good agreement with the detonation velocity reported in previous studies [52–54]. The peak von-Neumann pressure is 165.35 kPa and 165.34 kPa for WENO and AT-WENO which are also similar to the previous studies. Overall, the AT-WENO schemes derived in the present study exhibit excellent numerical accuracy despite the use of delayed data at processor boundaries. Since multidimensional simulations of detonation phenomena in combustion devices face highly intensive computational resource requirements [55,56], AT-WENO schemes hold the potential of offsetting a certain portion of communication overhead associated with standard WENO schemes at large node counts, thereby leading to improved scalability.

6. Discussion

DNS of turbulent combustion at higher Reynolds numbers with detailed chemical mechanisms and at conditions relevant for practical devices will require efficient utilization of massive computing resources anticipated on next generation exascale machines. These simulations at scales and conditions that are currently infeasible will provide fine-grained details into the interactions between turbulence and chemistry and will provide fundamental insights into the physical processes that result in pollutant formation, flame stabilization, blow-off, higher efficiency, etc. This in turn will not only aid in the design of fuel-flexible, low-emission combustion engines but also in the development of physics-based models for engineering simulations that will be used to design advanced clean engines. However, a successful transition of DNS codes to exascale machines and beyond is possible only through redesign and development of new computational algorithms in order to overcome some of the obvious bottlenecks and challenges.

The anticipated computing power of a quintillion (10^{18}) double precision floating point operations per second on the exascale machine will be realized through extreme levels of parallelism with millions of processing elements (PEs), including CPUs, GPUs and FPGAs. For example, Frontier at the Oak Ridge Leadership Computing Facility (OCLF) will have 1:4 CPU to GPU ratio with unified memory. Despite the use of state-of-the-art low-latency, high-bandwidth communication links, the sheer number of communications will significantly hinder scaling to large node counts. To this end, efficient strategies to
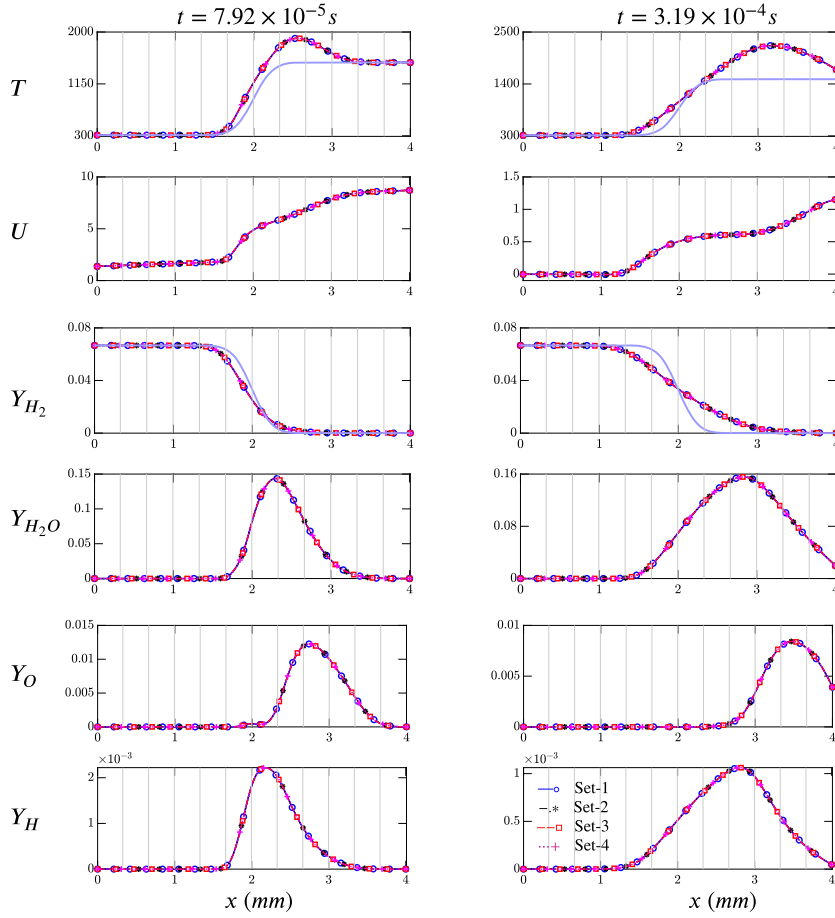
Fig. 13. Asynchrony-tolerant (AT): Instantaneous profiles and error in temperature (K), velocity (ms^{-1}) and mass fractions of H_2, H_2O, O and H for non-premixed ignition of H_2. The different lines: blue (Set-1: $p = [0.8, 0.1, 0.1]$), black (Set-2: $p = [0.6, 0.3, 0.1]$), red (Set-3: $p = [0.4, 0.5, 0.1]$) and magenta (Set-4: $p = [1.0, 0.0, 0.0]$) are defined in Table 3. The light blue line indicates the initial condition and the gray vertical lines represent processor boundaries.
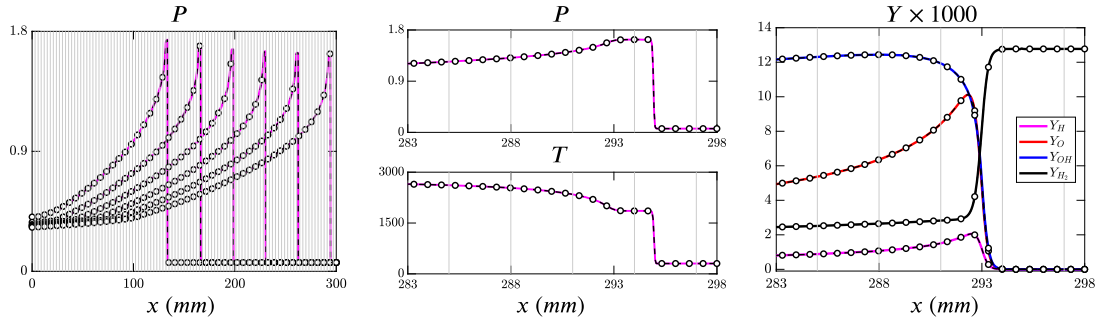

Fig. 14. Asynchrony-tolerant (AT): Pressure (atm) profiles (left) at different times, structure of detonation wave (middle) for pressure (atm) and temperature (K) and mass fractions of H, O, OH, H_2 (right). The solid lines are standard WENO and black-dashed line with black symbols (Set-3 in Table 3) corresponds to AT-WENO. Gray vertical lines represent processor boundaries.

mask communications and data movement will be needed to improve scalability. The use of AT schemes at the PE boundaries for computation of derivatives accurately, despite delayed data, will facilitate further overlap between communications and computations. However, a challenge in extending the AT schemes to three-dimensions is that the derivatives of convective and diffusive terms need to be expanded in terms of derivatives of primitive variables by applying the chain rule. For example, $\partial(\rho u e)/\partial x := (\rho u e)_x = \rho u e_x + \rho e u_x + e u \rho_x$. This expansion increases the total number of derivatives that need to be computed, especially when a large number of species are involved. Alternatively, one can compute the convective and diffusive terms (for example, $\rho u e$, $\rho Y_i u$, J_i, τ) from the primitive variables at different time levels as required by the AT schemes. These computations can be performed on the fly in order to avoid storing such product terms at multiple

time-levels. Essentially there is a trade-off between memory (data movement) and computation. Re-design of data structures for better memory coherency will also be extremely helpful since AT schemes use data from multiple time levels and can thus incur high cache miss rates. Besides these challenges, a critical first step in using AT schemes for scalable simulations of turbulent combustion on massive supercomputers is to ascertain that the numerical artifacts due to delayed data at PE boundary are significantly small and do not affect the underlying physics. The comprehensive set of numerical experiments presented in the present study specifically highlights the excellent numerical accuracy of AT schemes for compressible reacting unsteady flows.

A key component of simulations of reacting flows even for simple geometries includes the prescription of nonperiodic boundary conditions. The NSCBC [44,45] with improvements for reacting flows [47,48,50,57] has been successfully used for these simulations. In the normal direction, the terms in the NSCBC boundary conditions are not affected by AT schemes. For example, at the outflow, one-sided schemes are used, which require only local information within a PE. At the inflow, relaxation terms are used which again do not require PEs to communicate. Since transverse and viscous terms are also included for reacting flows [47,48,50], the AT schemes can be utilized in the transverse direction for computation of all the spatial derivatives involved. Extension to corners can be done following [57] with AT schemes used wherever necessary.

Another advantage for using asynchronous computations on exascale machines is to leverage the high flop-rate of GPUs, i.e. throughput computing. In hybrid computing architectures, GPUs are expected to handle most of the computations, while CPUs will facilitate communications between PEs. With AT schemes, GPUs will not have to wait on the CPUs for the most updated data from the neighboring PEs. This would enhance utilization of GPUs without affecting numerical accuracy or introducing idling penalties. A similar approach has been utilized in [58] where asynchronous copies between CPUs and GPUs are used to overlap computation and data movement, but delayed data with asynchronous computations have not been used. At a compiler level, new asynchronous run-time systems that are capable of dynamic task parallelism are being developed to improve the computation-communication overlap [59–61]. The AT schemes, which relax synchronization at a mathematical level, can be coupled with such data-centric programming models to create highly scalable PDE solvers. Asynchrony has also been utilized in [62] for scalable resilience to soft faults. However, in [62] all derivatives are still computed with the most updated data but the computations are re-arranged to ensure maximum overlap between communication and computation. The asynchronization approach utilized in [62] coupled with mathematical level asynchrony with AT schemes can be effective in pushing the scaling wall. Furthermore, since AT schemes do not require updated data, these schemes can also be used to recover from node failures without halting the simulation altogether.

7. Conclusions

In this paper, a series of numerical simulations that utilize delayed data at processor boundaries are presented for canonical reacting flows with single-step and detailed reaction mechanisms. It is shown that large numerical errors are incurred in physical as well as spectral space when standard schemes are used asynchronously, even for the simplest problems. To overcome this loss of accuracy, novel asynchrony-tolerant (AT) schemes are used for accurate computation of spatial derivatives with delayed data at the boundaries. These asynchronous simulations using AT schemes are compared with standard synchronous simulations and are shown to exhibit excellent qualitative and quantitative agreement. Both spatial and temporal evolution of hydrodynamic and thermo-chemical quantities are accurately captured by the AT schemes.

For simulations of high-speed flows with shocks and jump discontinuities, the standard shock-resolving WENO schemes have degraded numerical accuracy. Thus, following the procedure listed in [24] AT-WENO schemes are derived. However, the schemes so derived have non-convex linear weights which can lead to numerical instabilities and oscillations [39]. To overcome this, the asynchronous stencil is modified to include an additional grid-point. With this degree-of-freedom, an additional low-order truncation error term is selectively eliminated while preserving the convexity of the ideal weights. For evaluation of non-linear weights in AT-WENO schemes, asynchrony-tolerant smoothness indicators are also derived. The superior accuracy of AT-WENO schemes in comparison to the standard WENO schemes is exhibited through the order-of-accuracy tests for linear and non-linear equations. These schemes are also used for simulation of a detonation wave which is found to have excellent agreement with its synchronous counterpart.

A common conclusion one can arrive at through these simulations is that the AT schemes have excellent accuracy despite relaxed synchronizations at processor boundaries. Thus, AT schemes provide a potential path for highly scalable asynchronous simulations of turbulent combustion at the extreme scales promised by exascale and beyond.

CRediT authorship contribution statement

Komal Kumari: Conceptualisation, Methodology, Software, Analysis, Writing. **Emmet Cleary:** Methodology, Software, Analysis, Writing. **Swapnil Desai:** Methodology, Analysis, Review and Editing. **Diego A. Donzis:** Supervision, Review and Editing. **Jacqueline H. Chen:** Conceptualisation, Methodology, Analysis, Writing, Supervision. **Konduri Aditya:** Conceptualisation, Methodology, Analysis, Writing, Supervision.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Not applicable.

Data availability

Data will be made available on request.

Acknowledgements

Appendix A. AT schemes

Table A.13

Asynchrony-tolerant (AT) schemes for left and right boundary used in numerical simulations for first and second derivative.

(Derivative, Order)	Boundary	Scheme
(2,4)	Left	$\frac{1}{2}(\tilde{k}^2+3\tilde{k}+2)\left(-u_{j+2}^n+16u_{j+1}^n-30u_j^n+16u_{j-1}^{n-\tilde{k}}-u_{j-2}^{n-\tilde{k}}\right)/12\Delta x^2$ $-(\tilde{k}^2+2\tilde{k})\left(-u_{j+2}^n+16u_{j+1}^n-30u_j^n+16u_{j-1}^{n-\tilde{k}-1}-u_{j-2}^{n-\tilde{k}-1}\right)/12\Delta x^2$ $+\frac{1}{2}(\tilde{k}^2+\tilde{k})\left(-u_{j+2}^n+16u_{j+1}^n-30u_j^n+16u_{j-1}^{n-\tilde{k}-2}-u_{j-2}^{n-\tilde{k}-2}\right)/12\Delta x^2$
(2,4)	Right	$\frac{1}{2}(\tilde{k}^2+3\tilde{k}+2)\left(-u_{j+2}^{n-\tilde{k}}+16u_{j+1}^{n-\tilde{k}}-30u_j^n+16u_{j-1}^n-u_{j-2}^n\right)/12\Delta x^2$ $-(\tilde{k}^2+2\tilde{k})\left(-u_{j+2}^{n-\tilde{k}-1}+16u_{j+1}^{n-\tilde{k}-1}-30u_j^n+16u_{j-1}^n-u_{j-2}^n\right)/12\Delta x^2$ $+\frac{1}{2}(\tilde{k}^2+\tilde{k})\left(-u_{j+2}^{n-\tilde{k}-2}+16u_{j+1}^{n-\tilde{k}-2}-30u_j^n+16u_{j-1}^n-u_{j-2}^n\right)/12\Delta x^2$
(1,4)	Left	$\frac{1}{2}(\tilde{k}^2+3\tilde{k}+2)\left(-u_{j+2}^n+8u_{j+1}^n-8u_{j-1}^{n-\tilde{k}}+u_{j-2}^{n-\tilde{k}}\right)/12\Delta x$ $-(\tilde{k}^2+2\tilde{k})\left(-u_{j+2}^n+8u_{j+1}^n-8u_{j-1}^{n-\tilde{k}-1}+u_{j-2}^{n-\tilde{k}-1}\right)/12\Delta x$ $+\frac{1}{2}(\tilde{k}^2+\tilde{k})\left(-u_{j+2}^n+8u_{j+1}^n-8u_{j-1}^{n-\tilde{k}-2}+u_{j-2}^{n-\tilde{k}-2}\right)/12\Delta x$
(1,4)	Right	$\frac{1}{2}(\tilde{k}^2+3\tilde{k}+2)\left(-u_{j+2}^{n-\tilde{k}}+8u_{j+1}^{n-\tilde{k}}-8u_{j-1}^n+u_{j-2}^n\right)/12\Delta x$ $-(\tilde{k}^2+2\tilde{k})\left(-u_{j+2}^{n-\tilde{k}-1}+8u_{j+1}^{n-\tilde{k}-1}-8u_{j-1}^n+u_{j-2}^n\right)/12\Delta x$ $+\frac{1}{2}(\tilde{k}^2+\tilde{k})\left(-u_{j+2}^{n-\tilde{k}-2}+8u_{j+1}^{n-\tilde{k}-2}-8u_{j-1}^n+u_{j-2}^n\right)/12\Delta x$

Appendix B. Reconstruction approximation at the right boundary

For points located at the right processor boundary the modified smaller asynchronous stencil is given by, $\tilde{S}_R^{(0)} = \{u_{j-2}^n, u_{j-1}^n, u_j^n\}$, $\tilde{S}_R^{(1)} = \{u_{j-1}^n, u_j^n, u_j^{n-\tilde{k}}, u_{j+1}^{n-\tilde{k}} u_{j+1}^{n-\tilde{k}-1}\}$, and $\tilde{S}_R^{(2)} = \{u_j^n, u_{j+1}^{n-\tilde{k}}, u_{j+1}^{n-\tilde{k}-1}, u_{j+2}^{n-\tilde{k}} u_{j+2}^{n-\tilde{k}-1}\}$. Here the delay appears at the rightmost gird points. We derive the following reconstruction approximation at the right boundary,

$$\tilde{u}_{j+\frac{1}{2}}^{n,(0)} = \frac{u_{j-2}^n}{3} - \frac{7u_{j-1}^n}{6} + \frac{11u_j^n}{6}$$

$$\tilde{u}_{j+\frac{1}{2}}^{n,(1)} = -\frac{1}{3}\tilde{k}u_{j+1}^{-\tilde{k}+n-1} + \frac{1}{6}\left(2\tilde{k}+2\right)u_{j+1}^{n-\tilde{k}} - \frac{1}{6}u_{j-1}^n + \frac{5u_j^n}{6}$$

$$\tilde{u}_{j+\frac{1}{2}}^{n,(2)} = -\frac{5}{6}\tilde{k}u_{j+1}^{-\tilde{k}+n-1} + \frac{1}{6}\tilde{k}u_{j+2}^{-\tilde{k}+n-1} + \frac{1}{6}\left(5\tilde{k}+5\right)u_{j+1}^{n-\tilde{k}} + \frac{1}{6}\left(-\tilde{k}-1\right)u_{j+2}^{n-\tilde{k}} + \frac{u_j^n}{3}$$

(B.1)

Similarly, the approximation at point $u_{j-1/2}$ on the right boundary at each of the candidate stencils is listed below

$$\tilde{u}_{j-\frac{1}{2}}^{n,(0)} = -\frac{1}{6}u_{j-2}^n + \frac{5u_{j-1}^n}{6} + \frac{u_j^n}{3}$$

$$\tilde{u}_{j-\frac{1}{2}}^{n,(1)} = \frac{1}{6}\tilde{k}u_{j+1}^{-\tilde{k}+n-1} + \frac{1}{6}\left(-\tilde{k}-1\right)u_{j+1}^{n-\tilde{k}} + \frac{u_{j-1}^n}{3} + \frac{5u_j^n}{6}$$

$$\tilde{u}_{j-\frac{1}{2}}^{n,(2)} = \frac{7}{6}\tilde{k}u_{j+1}^{-\tilde{k}+n-1} - \frac{1}{3}\tilde{k}u_{j+2}^{-\tilde{k}+n-1} + \frac{1}{6}\left(-7\tilde{k}-7\right)u_{j+1}^{n-\tilde{k}} + \frac{1}{6}\left(2\tilde{k}+2\right)u_{j+2}^{n-\tilde{k}} + \frac{11u_j^n}{6}$$

(B.2)

Following the procedure described in the previous sections, the smoothness indicator at the right-boundary can be computed to be

$$\tilde{\beta}^{(0)} = \frac{1}{3}\left(4\left(u^n\right)_{j-2}^2 + \left(11u_j^n - 19u_{j-1}^n\right)u_{j-2}^n + 25\left(u^n\right)_{j-1}^2 + 10\left(u^n\right)_j^2 - 31u_{j-1}^n u_j^n\right) \tag{B.3}$$

$$\tilde{\beta}^{(1)} = \frac{1}{3}\left(u_{j-1}^n\left(-5\tilde{k}u_{j+1}^{n-\tilde{k}-1} + 5\left(\tilde{k}+1\right)u_{j+1}^{n-\tilde{k}} - 13u_j^n\right) + 4\left(\tilde{k}u_{j+1}^{n-\tilde{k}-1} - \left(\tilde{k}+1\right)u_{j+1}^{n-\tilde{k}}\right)^2\right.$$
$$\left. + 13u_j^n\left(\tilde{k}u_{j+1}^{n-\tilde{k}-1} - \left(\tilde{k}+1\right)u_{j+1}^{n-\tilde{k}}\right) + 4\left(u_{j-1}^n\right)^2 + 13\left(u_j^n\right)^2\right) \tag{B.4}$$

$$\tilde{\beta}^{(2)} = \frac{1}{3}\left(\tilde{k}^2\left(25\left(u_{j+1}^{n-\tilde{k}-1}\right)^2 + \left(19\left(u_{j+2}^{n-\tilde{k}} - u_{j+2}^{n-\tilde{k}-1}\right) - 50u_{j+1}^{n-\tilde{k}}\right)u_{j+1}^{n-\tilde{k}-1} + 25\left(u_{j+1}^{n-\tilde{k}}\right)^2\right.\right.$$
$$\left. + 4\left(u_{j+2}^{n-\tilde{k}-1} - u_{j+2}^{n-\tilde{k}}\right)^2 + 19u_{j+1}^{n-\tilde{k}}\left(u_{j+2}^{n-\tilde{k}-1} - u_{j+2}^{n-\tilde{k}}\right)\right) + \tilde{k}\left(50\left(u_{j+1}^{n-\tilde{k}}\right)^2\right.$$
$$\left. + 19\left(u_{j+2}^{n-\tilde{k}-1} - 2u_{j+2}^{n-\tilde{k}}\right)u_{j+1}^{n-\tilde{k}} + 8u_{j+2}^{n-\tilde{k}}\left(u_{j+2}^{n-\tilde{k}} - u_{j+2}^{n-\tilde{k}-1}\right) + u_{j+1}^{n-\tilde{k}-1}\left(19u_{j+2}^{n-\tilde{k}} - 50u_{j+1}^{n-\tilde{k}}\right)\right)$$
$$\left. + 25\left(u_{j+1}^{n-\tilde{k}}\right)^2 + 4\left(u_{j+2}^{n-\tilde{k}}\right)^2 - 19u_{j+1}^{n-\tilde{k}}u_{j+2}^{n-\tilde{k}} + u_j^n\left(-31u_{j+1}^{n-\tilde{k}} + 11u_{j+2}^{n-\tilde{k}}\right.\right.$$
$$\left.\left. + \tilde{k}\left(31u_{j+1}^{n-\tilde{k}-1} - 31u_{j+1}^{n-\tilde{k}} - 11u_{j+2}^{n-\tilde{k}-1} + 11u_{j+2}^{n-\tilde{k}}\right)\right) + 10\left(u_j^n\right)^2\right) \tag{B.5}$$

Lastly, we note that the approximation of $u_{j-1/2}$ at the left-boundary is

$$\tilde{u}_{j-\frac{1}{2}}^{n,(0)} = \frac{1}{6}\tilde{k}u_{j-2}^{-\tilde{k}+n-1} - \frac{5}{6}\tilde{k}u_{j-1}^{-\tilde{k}+n-1} + \frac{1}{6}\left(-\tilde{k}-1\right)u_{j-2}^{n-\tilde{k}} + \frac{1}{6}\left(5\tilde{k}+5\right)u_{j-1}^{n-\tilde{k}} + \frac{u_j^n}{3}$$
$$\tilde{u}_{j-\frac{1}{2}}^{n,(1)} = -\frac{1}{3}\tilde{k}u_{j-1}^{-\tilde{k}+n-1} + \frac{1}{6}\left(2\tilde{k}+2\right)u_{j-1}^{n-\tilde{k}} + \frac{5u_j^n}{6} - \frac{u_{j+1}^n}{6} \tag{B.6}$$
$$\tilde{u}_{j-\frac{1}{2}}^{n,(2)} = \frac{11}{6}u_j^n - \frac{7}{6}u_{j+1}^n + \frac{1}{3}u_{j+2}^n.$$

References

[1] P. Moin, K. Mahesh, Direct numerical simulation: a tool in turbulence research, Annu. Rev. Fluid Mech. 30 (1998) 539–578.
[2] M. Visbal, D. Gaitonde, D. Rizzetta, High-Order Schemes for DNS/LES and CAA on Curvilinear Dynamic Meshes, DNS/LES Progress and Challenges, Greyden Press, 2001.
[3] J.H. Chen, A. Choudhary, B. de Supinski, M. DeVries, E.R. Hawkes, S. Klasky, W.K. Liao, K.L. Ma, J. Mellor-Crummey, N. Podhorszki, R. Sankaran, S. Shende, C.S. Yoo, Terascale direct numerical simulations of turbulent combustion using S3D, Comput. Sci. Discov. 2 (2009) 015001.
[4] A. Attili, F. Bisetti, M.E. Mueller, H. Pitsch, Formation, growth, and transport of soot in a three-dimensional turbulent non-premixed jet flame, Combust. Flame 161 (2014) 1849–1865.
[5] K. Aditya, A. Gruber, C. Xu, T. Lu, A. Krisman, M.R. Bothien, J.H. Chen, Direct numerical simulation of flame stabilization assisted by autoignition in a reheat gas turbine combustor, Proc. Combust. Inst. 37 (2019) 2635–2642.
[6] B. Savard, E.R. Hawkes, K. Aditya, H. Wang, J.H. Chen, Regimes of premixed turbulent spontaneous ignition and deflagration under gas-turbine reheat combustion conditions, Combust. Flame 208 (2019) 402–419.
[7] A. Gruber, E.S. Richardson, K. Aditya, J.H. Chen, Direct numerical simulations of premixed and stratified flame propagation in turbulent channel flow, Phys. Rev. Fluids 3 (2018) 110507.
[8] J. Zhang, M.B. Luong, F.E.H. Pérez, D. Han, H.G. Im, Z. Huang, Exergy loss characteristics of DME/air and ethanol/air mixtures with temperature and concentration fluctuations under HCCI/SCCI conditions: a DNS study, Combust. Flame 226 (2021) 334–346.
[9] L. Berger, R. Hesse, K. Kleinheinz, M.J. Hegetschweiler, A. Attili, J. Beeckmann, G.T. Linteris, H. Pitsch, A DNS study of the impact of gravity on spherically expanding laminar premixed flames, Combust. Flame 216 (2020) 412–425.
[10] Y.J. Kim, B.J. Lee, H.G. Im, Effects of differential diffusion on the stabilization of unsteady lean premixed flames behind a bluff-body, Flow Turbul. Combust. (2021) 1125–1141.
[11] G. Nivarti, S. Cant, Direct numerical simulation of the bending effect in turbulent premixed flames, Proc. Combust. Inst. 36 (2017) 1903–1910.
[12] S. Desai, Y.J. Kim, W. Song, M.B. Luong, F.E.H. Pérez, R. Sankaran, H.G. Im, Direct numerical simulations of turbulent reacting flows with shock waves and stiff chemistry using many-core/gpu acceleration, Comput. Fluids 215 (2021) 104787.
[13] G. Beardsell, G. Blanquart, Fully compressible simulations of the impact of acoustic waves on the dynamics of laminar premixed flames for engine-relevant conditions, Proc. Combust. Inst. 38 (2021) 1923–1931.
[14] K. Aditya, D.A. Donzis Poster, Asynchronous computing for partial differential equations at extreme scales, in: Proceedings of the 2012 SC Companion: High Performance Computing, Networking Storage and Analysis, SCC '12, IEEE Computer Society, Washington, DC, USA, 2012, p. 1444.
[15] D.A. Donzis, K. Aditya, Asynchronous finite-difference schemes for partial differential equations, J. Comput. Phys. 274 (2014) 370–392.

[16] J. Dongarra, P. Beckman, T. Moore, P. Aerts, G. Aloisio, J.-C. Andre, D. Barkai, J.-Y. Berthou, T. Boku, B. Braunschweig, F. Cappello, B. Chapman, Xuebin Chi, A. Choudhary, S. Dosanjh, T. Dunning, S. Fiore, A. Geist, B. Gropp, R. Harrison, M. Hereld, M. Heroux, A. Hoisie, K. Hotta, Zhong Jin, Y. Ishikawa, F. Johnson, S. Kale, R. Kenway, D. Keyes, B. Kramer, J. Labarta, A. Lichnewsky, T. Lippert, B. Lucas, B. Maccabe, S. Matsuoka, P. Messina, P. Michielse, B. Mohr, M.S. Mueller, W.E. Nagel, H. Nakashima, M.E. Papka, D. Reed, M. Sato, E. Seidel, J. Shalf, D. Skinner, M. Snir, T. Sterling, R. Stevens, F. Streitz, B. Sugar, S. Sumimoto, W. Tang, J. Taylor, R. Thakur, A. Trefethen, M. Valero, A. van der Steen, J. Vetter, P. Williams, R. Wisniewski, K. Yelick, The international exascale software project roadmap, Int. J. High Perform. Comput. Appl. 25 (2011) 3–60.
[17] D.P. Bertsekas, J.N. Tsitsiklis, Parallel and Distributed Computation: Numerical Methods, Prentice-Hall, Inc., USA, 1989.
[18] A. Frommer, D.B. Szyld, On asynchronous iterations, J. Comput. Appl. Math. 123 (2000) 201–216.
[19] K. Lee, R. Bhattacharya, J. Dass, V.N.S.P. Sakuru, R.N. Mahapatra, A relaxed synchronization approach for solving parallel quadratic programming problems with guaranteed convergence, in: 2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2016, pp. 182–191.
[20] K. Lee, R. Bhattacharya, V. Gupta, A switched dynamical system framework for analysis of massively parallel asynchronous numerical algorithms, in: 2015 American Control Conference (ACC), 2015, pp. 1095–1100.
[21] D. Amitai, A. Averbuch, S. Itzikowitz, M. Israeli, Parallel adaptive and time-stabilizing schemes for constant-coefficient parabolic pde's, Comput. Math. Appl. 24 (1992) 33–53.
[22] D. Amitai, A. Averbuch, S. Itzikowitz, E. Turkel, Asynchronous and corrected-asynchronous finite difference solutions of PDEs on MIMD multiprocessors, Numer. Algorithms 6 (1994) 275–296.
[23] A. Mittal, S. Girimaji, Proxy-equation paradigm: a strategy for massively parallel asynchronous computations, Phys. Rev. E 96 (2017) 033304.
[24] K. Aditya, D.A. Donzis, High-order asynchrony-tolerant finite difference schemes for partial differential equations, J. Comput. Phys. 350 (2017) 550–572.
[25] D. Mudigere, S.D. Sherlekar, S. Ansumali, Delayed difference scheme for large scale scientific simulations, Phys. Rev. Lett. 113 (2014) 218701.
[26] K. Aditya, T. Gysi, G. Kwasniewski, T. Hoefler, D.A. Donzis, J.H. Chen, A scalable weakly-synchronous algorithm for solving partial differential equations, 2019.
[27] K. Kumari, D.A. Donzis, Direct numerical simulations of turbulent flows using high-order asynchrony-tolerant schemes: accuracy and performance, J. Comput. Phys. 419 (2020) 109626.
[28] K. Kumari, D.A. Donzis, A generalized von Neumann analysis for multi-level schemes: stability and spectral accuracy, J. Comput. Phys. 424 (2021) 109868.
[29] T. Hoefler, T. Schneider, A. Lumsdaine, Characterizing the influence of system noise on large-scale applications by simulation, in: International Conference for High Performance Computing, Networking, Storage and Analysis (SC'10), 2010.
[30] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: A. Quarteroni (Ed.), Advanced Numerical Approximation of Nonlinear Hyperbolic Equations: Lectures Given at the 2nd Session of the Centro Internazionale Matematico Estivo (C.I.M.E.) Held in Cetraro, Italy, June 23–28, 1997, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 325–432.
[31] C.-W. Shu, High order weighted essentially nonoscillatory schemes for convection dominated problems, SIAM Rev. 51 (2009) 82–126.
[32] S. Pirozzoli, Numerical methods for high-speed flows, Annu. Rev. Fluid Mech. 43 (2011) 163–194.
[33] I. Bermejo-Moreno, J. Bodart, J. Larsson, B.M. Barney, J.W. Nichols, S. Jones, Solving the compressible Navier-Stokes equations on up to 1.97 million cores and 4.1 trillion grid points, in: SC '13: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, 2013, pp. 1–10.
[34] A. Mosedale, D. Drikakis, Assessment of very high order of accuracy in implicit LES models, J. Fluids Eng. 129 (2007) 1497–1503.
[35] K. Ritos, I.W. Kokkinakis, D. Drikakis, Physical insight into the accuracy of finely-resolved iLES in turbulent boundary layers, Comput. Fluids 169 (2018) 309–316.
[36] K. Ritos, I.W. Kokkinakis, D. Drikakis, Performance of high-order implicit large Eddy simulations, Comput. Fluids 173 (2018) 307–312.
[37] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes, Acta Numer. (2020) 63.
[38] G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted eno schemes, J. Comput. Phys. 126 (1996) 202–228.
[39] J. Shi, C. Hu, C.-W. Shu, A technique of treating negative weights in WENO schemes, J. Comput. Phys. 175 (2002) 108–127.
[40] Y.-y. Liu, C.-w. Shu, M.-p. Zhang, On the positivity of linear weights in weno approximations, Acta Math. Appl. Sin. Engl. Ser. 25 (2009) 503–538.
[41] R.J. Kee, F.M. Rupley, E. Meeks, J.A. Miller, CHEMKIN-III: a FORTRAN chemical kinetics package for the analysis of gas-phase chemical and plasma kinetics, Technical Report, Sandia National Labs., Livermore, CA (United States), 1996.
[42] R.J. Kee, G. Dixon-Lewis, J. Warnatz, M.E. Coltrin, J.A. Miller, A fortran computer code package for the evaluation of gas-phase multicomponent transport properties, Sandia National Laboratories Report SAND86-8246 13, 80401-1887 1986.
[43] M. Baum, T. Poinsot, D. Thévenin, Accurate boundary conditions for multicomponent reactive flows, J. Comput. Phys. 116 (1995) 247–261.
[44] K.W. Thompson, Time dependent boundary conditions for hyperbolic systems, J. Comput. Phys. 68 (1987) 1–24.
[45] T.J. Poinsot, S.K. Lele, Boundary conditions for direct simulations of compressible viscous flows, J. Comput. Phys. 101 (1992) 104–129.
[46] Z. Luo, C.S. Yoo, E.S. Richardson, J.H. Chen, C.K. Law, T. Lu, Chemical explosive mode analysis for a turbulent lifted ethylene jet flame in highly-heated coflow, Combust. Flame 159 (2012) 265–274.
[47] C.S. Yoo, Y. Wang, A. Trouvé, H.G. Im, Characteristic boundary conditions for direct simulations of turbulent counterflow flames, Combust. Theory Model. 9 (2005) 617–646.
[48] C.S. Yoo, H.G. Im, Characteristic boundary conditions for simulations of compressible reacting flows with multi-dimensional, viscous and reaction effects, Combust. Theory Model. 11 (2007) 259–286.
[49] M.P. Burke, M. Chaos, Y. Ju, F.L. Dryer, S.J. Klippenstein, Comprehensive h2/o2 kinetic model for high-pressure combustion, Int. J. Chem. Kinet. 44 (2012) 444–474.
[50] J.C. Sutherland, C.A. Kennedy, Improved boundary conditions for viscous, reacting, compressible flows, J. Comput. Phys. 191 (2003) 502–524.
[51] C.K. Westbrook, Chemical kinetics of hydrocarbon oxidation in gaseous detonations, Combust. Flame 46 (1982) 191–210.
[52] R. Deiterding, Parallel adaptive simulation of multi-dimensional detonation structures, 2003.
[53] Y. Lv, M. Ihme, Discontinuous Galerkin method for multicomponent chemically reacting flows and combustion, J. Comput. Phys. 270 (2014) 105–137.
[54] R.W. Houim, K.K. Kuo, A low-dissipation and time-accurate method for compressible multi-component flow with variable specific heat ratios, J. Comput. Phys. 230 (2011) 8527–8553.
[55] M.B. Luong, S. Desai, F.E.H. Pérez, R. Sankaran, B. Johansson, H.G. Im, A statistical analysis of developing knock intensity in a mixture with temperature inhomogeneities, Proc. Combust. Inst. (2020).
[56] M.B. Luong, S. Desai, F.E. Hernández Pérez, R. Sankaran, B. Johansson, H.G. Im, Effects of turbulence and temperature fluctuations on knock development in an ethanol/air mixture, Flow Turbul. Combust. (2020) 1–21.
[57] G. Lodato, H. Pitsch, Characteristic Outflows with Optimal Transverse Terms: The Edges and Corners Coupling Algorithm, Center for Turbulence Research, Annual Research Briefs, 2011, p. 12.
[58] P.K. Yeung, K. Ravikumar, Advancing understanding of turbulence through extreme-scale computation: intermittency and simulations at large problem sizes, Phys. Rev. Fluids 5 (2020) 110517.
[59] E. Slaughter, W. Lee, S. Treichler, M. Bauer, A. Aiken, Regent: a high-productivity programming language for hpc with logical regions, in: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC '15, Association for Computing Machinery, New York, NY, USA, 2015.

[60] B. Acun, A. Gupta, N. Jain, A. Langer, H. Menon, E. Mikida, X. Ni, M. Robson, Y. Sun, E. Totoni, L. Wesolowski, L. Kale, Parallel programming with migratable objects: Charm++ in practice, in: SC '14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, 2014, pp. 647–658.

[61] A. Kulkarni, A. Lumsdaine, A comparative study of asynchronous many-tasking runtimes: Cilk, charm++, parallex and am++, preprint, arXiv:1904.00518, 2019.

[62] H. Kolla, J.R. Mayo, K. Teranishi, R.C. Armstrong, Improving scalability of silent-error resilience for message-passing solvers via local recovery and asynchrony, in: 2020 IEEE/ACM 10th Workshop on Fault Tolerance for HPC at eXtreme Scale (FTXS), 2020, pp. 1–10.