International Journal for Numerical Methods in Engineering

WILEY

RESEARCH ARTICLE OPEN ACCESS

A Multi-Time Stepping Algorithm for the Modelling of Heterogeneous Structures With Explicit Time Integration

Kin Fung Chan[1] | Nicola Bombace[2] | Duygu Sap[1,3] | David Wason[1] | Simone Falco[1] | Nik Petrinic[1]

[1]Department of Engineering Science, University of Oxford, Oxford, England, UK | [2]Adaptive, Embedded and AI (AEAI) Group, Advanced Micro Devices Inc., Edinburgh, Scotland, UK | [3]Mathematical Institute, University of Oxford, Oxford, England, UK

Correspondence: Kin Fung Chan (kin.chan@eng.ox.ac.uk)

Received: 18 December 2023 | **Revised:** 5 September 2024 | **Accepted:** 25 November 2024

Funding: The authors gratefully acknowledge the financial support from the Engineering and Physical Sciences Research Council (EPSRC) and Rolls-Royce's ASiMoV Prosperity Partnership with References EP/S005072/1 (Kin Fung Chan, Duygu Sap, and Nik Petrinic), EP/R029423/1 (Duygu Sap), EP/R513295/1 (David Wason) and Advanced Micro Devices Inc. (Nicola Bombace). The authors also thank the reviewers for their helpful comments, improving this manuscript.

Keywords: asynchronous time integrators | metamaterials | multi-step | stress wave propagation | subcycling

ABSTRACT

Heterogeneous solids often exhibit complex dynamic behavior, requiring simulations to use varying time steps. However, the conventional use of a single-time step for the entire domain can be inefficient. This article proposes a multi-time stepping algorithm that addresses this challenge by relaxing the constraint for an integer or constant time step ratio between subdomains and eliminating the need for kinematic interpolation. The algorithm ensures the satisfaction of the Courant-Friedrichs-Lewy condition, deviating only to allow subdomains to remain in synchronization. Consequently, less integration steps are performed in comparison to state-of-the-art asynchronous integrators. We extend to the coupling of multiple subdomains, where each subdomain has its time step. Simulating stress wave propagation in metamaterials demonstrates that the proposed algorithm significantly accelerates simulation time, without sacrificing accuracy.

1 | Introduction

The critical time step of a domain is a fundamental limit on the computational efficiency of explicit time integration methods. These methods remain attractive when resolving the highly dynamic behaviour of heterogeneous structures, or solids subject to short-term phenomena. Commonly, finite element methods are used to model stress wave propagation problems in such structures. The varied material properties within these heterogeneous structures significantly influence the critical time step, as determined by the Courant-Friedrichs-Lewy (CFL) condition [1]. This can lead to large portions of a domain utilising a time step much smaller than required, hence increasing computation unnecessarily. Subsequently, pioneering works by Belythscko and Mullen [2, 3], and Hughes, Pister, and Taylor [4] introduced the concept of subcycling, enabling subdomains to integrate with different time steps. Their implementations contrasted with the first utilising nodal partitioning, and the latter opting for elements, both requiring the primal overhead of defining clocks to determine which subdomains to advance. The idea of asynchronous variational integrators followed similar ideas to subcycling methods, however, they chose the time step for each element, subject to stability considerations [5]. Commonly in all algorithms, kinematic quantities of varying choice are synchronised, or interpolated, at the interfaces between differently integrated partitions [6–8].

A key concept is the time step ratio m between subdomains. For a large subdomain Ω_L and small subdomain Ω_s, their

time step ratio; $m = \Delta t_L / \Delta t_s$. A significant constraint lies in the assumption that this ratio remains constant throughout a simulation. Many existing methods, such as Mota, Tezaur, and Phlipot [9] adhere to this assumption, potentially limiting their application to scenarios where time steps suddenly change. Another common limitation is those restricted to integer time step ratios. Although algorithms that explore non-integer ratios exist [6, 10], the majority of approaches including those based of the FETI method [11] and mortar methods [12] still maintain integer ratios. Moreover, many studies have primarily focused on problems that integrate just two subdomains. The extension to multiple domains is a requirement for the practical use of these algorithms. Gravouil, Combescure, and Brun developed the heterogeneous asynchronous time integrators (HATI), where multiple subdomains can be used, as well as utilising differing time integration schemes [13]. Chantrait, Rannou, and Gravouil [14]. Notably, the work that preceded looked to combine "mixed" and "multi-time stepping" integration to couple explicit and implicit nonlinear schemes together, where time step ratios were of the order of 100 [15]. Gravouil and Combescure [16] Recently it has been shown that those based on the Gravouil-Combescure (GC) approach do not necessarily fulfill the continuity of all three kinematic fields. The phenomenon, named drifting, is therefore observed when displacement continuity is not met on the interfaces [17]. Consequently, despite the extensive exploration of the interface dynamics, challenges still remain with respect to stability and accuracy.

Given explicit methods are only conditionally stable, the choice to integrate at a larger time step than the critical time step introduces instability. A common method for analysing stability is to use amplification matrices and eigenvalue decomposition [18]. Another approach uses pseudo-energy norms that depict stability if the energy remains constant at every time step [7, 19]. In nonlinear settings such as plasticity, this does not necessarily ensure stability, as systems can regain stability. As a result, an energy check is often preferred [6]. The partial fulfillment of continuity is another source of instability and inaccuracies in simulations. The lack of enforcement of continuity of displacement, velocity, and acceleration across subdomain interfaces can be found in the works initially proposed by Gravouil and Combescure [15, 16]. While methods by Mahjoubi, Krenk, and Dhia prioritise the continuity of displacement [20–23], and others with velocity continuity [13, 14], few address all three kinematic quantities simultaneously. Recently Cho et al. [17] and Dvorak et al. [24] have enforced all three. Their energy conserving method follows on from the use of Lagrange Multipliers where coupling forces and a frame acceleration are enforced on the interface between subdomains. However, these methods suffer from computing time steps much smaller than required by the CFL condition in regions of a subdomain.

As an initial comparison, we present the number of integration steps computed by each element for a benchmark where the time step ratio $m = \pi$ in Table 1. Integrating monolithically (single-time step) allows the use of the CFL condition on the small domain, however, the time step is applied for all elements in all subdomains, hence summating the largest number of steps. The limitation of running on integer time step ratios is evident in the works of Cho et al. [17], whereas Dvorak et al. defines regions that require the solution of very small time steps to find common times. The proposed method steps closer to the CFL condition overall, as shown with the same $\min\{\Delta t\}$ as the monolithic solution. Through comparing state-of-the-art algorithms, this simple benchmark shows the motivation to still reduce computational effort through integrating elements with larger time steps.

TABLE 1 | Comparison of methods for minimum time step and integration steps for a square wave benchmark with $m = \pi$.

Method	Time step ratio	Minimum Δt (s)	Element integration steps
Single-time Step (Monolithic)	1 (CFL of Ω_s)	**5.3052e-7**	2,714,400
Cho et al. [17]	4	4.1666e-7	2,599,680
Dvorak et al. [24]	π	1.5992e-11	2,112,900
Proposed (2024)	π	**5.3052e-7**	**2,112,600**

Compared to the state-of-the-art, we improve the computational efficiency of multi-time stepping algorithms, whilst also circumventing the restriction of integer or constant time step ratios. This is significant across applications, granted distortion, fracture [25] or material behaviour [26, 27] can cause sudden variation in element time steps across a domain. The algorithm negates the need to solve an additional system of equations, through explicit calculation of coupling quantities, contrasting to those methods that employ Lagrange multipliers. The majority of the previously aforementioned algorithms do not present a method that extends to more than two subdomains. For practical use of multi-time stepping, we declare our algorithm in such a setting, where more than two subdomains can be solved concurrently, with ease of implementation as a key contribution. For the stability of the solution, we enforce the continuity of the three kinematic fields, as well as tractions. We quantify the energy computed in the coupling, and maintain similar conditional stability as the conventional CFL conditions. In summary:

- Algorithm circumvents constant or integer time step ratios,
- Coupling quantities are explicitly computed with minimum communication between subdomains,
- Irregular time steps are avoided for synchronisation of subdomain time steps,
- Subdomains step close to the CFL condition and reduce computational effort in comparison to the state-of-the-art,
- Continuity of all three kinematic variables with equal coupling energy on subdomain interfaces computed.

The contents of the paper are organised as follows. Section 2 formulates the structural dynamics problem and describes the proposed algorithm for two time domains in an updated Lagrangian formulation with the leapfrog integration scheme. In Section 3 we demonstrate the algorithm's stability, comparing it to methods that utilise Lagrange multipliers and assess for continuity. The energy associated with the coupling is assessed, computing the increment over large and small time steps. Section 4 presents the comparison to the state-of-the-art with the results to the square

wave benchmark. Key remarks surrounding extension to multiple domains and ease of implementation are made in Section 5. 3-D numerical examples are showcased in Section 6, demonstrating applications of the multi-time domain algorithms, whilst emphasising the significant speedup in runtime. Section 7 concludes the article with a summary of the key findings.

2 | Formulation of Subcycling Algorithms

To investigate the deformation of a heterogeneous structure under an impact load, the propagation of elastic stress waves across several of these subdomains is initially studied. The following section briefly overviews the fundamental equations of motion of a dynamic problem and the novel multi-time stepping algorithm, starting with the coupling of two subdomains. In this article, we use superscripts to denote quantities associated with time and subscripts to label subdomains.

2.1 | Governing Equations for Multiple Solid Body Subdomains

Let $\mathcal{B}$ be a solid body in an open region $\Omega \subset \mathbb{R}^3$ subjected to an external force, and denote by $\partial\Omega$ its boundary. The domain Ω is divided into S non-overlapping subdomains with (see Figure 1):

$$\Omega = \bigcup_{i=1}^{S} \Omega_i \text{ and } \Omega_i \cap \Omega_j = \emptyset \text{ for } i \neq j \tag{1}$$

Let Ω_L and Ω_s be the respective large and small subdomains in a two-subdomain case, and denote by Γ their interface. We formulate the motion of the solid body via the momentum balance equation for each subdomain i, assuming constant acceleration over the boundary Γ_i between Ω_L and Ω_s:

$$\rho_L \ddot{\mathbf{u}} = \nabla \cdot \boldsymbol{\sigma}_\mathbf{L} + \rho_L \mathbf{b}, \text{ in } \Omega_L \times [0, T] \tag{2}$$

$$\rho_s \ddot{\mathbf{u}} = \nabla \cdot \boldsymbol{\sigma}_\mathbf{s} + \rho_s \mathbf{b}, \text{ in } \Omega_s \times [0, T] \tag{3}$$

$$\ddot{\mathbf{u}}_L = \ddot{\mathbf{u}}_s \text{ on } \Gamma \tag{4}$$

$$\mathbf{t}_L = \mathbf{t}_s \text{ on } \Gamma \tag{5}$$

where $\boldsymbol{\sigma}$, $\mathbf{u}$, ρ and $\mathbf{b}$ denote the Cauchy stress tensor, the displacement field, the density of the solid, and the body forces, respectively. We enforce both the Dirichlet and Neumann boundary conditions, as well as utilising matching conditions at the subdomain interfaces Γ_i with the continuity of acceleration $\ddot{\mathbf{u}}$ and tractions $\mathbf{t}$.

We describe the deformation of $\mathcal{B}$ at time $t \in [0, T]$ for a specified constant $T > 0$ by a smooth, invertible function $\boldsymbol{\phi}(\mathbf{X}, t)$: $\Omega^0 \to \Omega^t$ where Ω^0 denotes the initial state of Ω and Ω^t denotes the (deformed) state of Ω at time t. Regardless of the time step, both single-time step (monolithic) and multi-time step solutions follow a similar procedure in describing the motion, as depicted in Figure 2. In the configuration, the spatial coordinates are expressed in terms of a function of the material point $\mathbf{X}$ and time t. The deformation gradient $\mathbf{F}$ of the motion $\boldsymbol{\phi}$ is defined by:

$$\mathbf{F}(\mathbf{X}, t) = \nabla \boldsymbol{\phi}(\mathbf{X}, t) = \frac{\partial \mathbf{x}^t}{\partial \mathbf{X}} \tag{6}$$

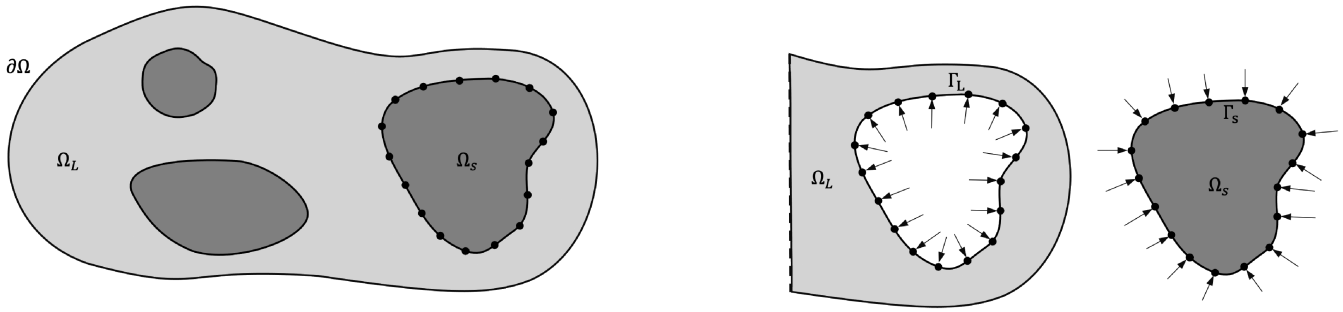

FIGURE 1 | Decomposition into S subdomains where Ω_L and Ω_s denote large and small subdomain respectively. Dots on Γ_i represent the coupling nodes where variables are exchanged.

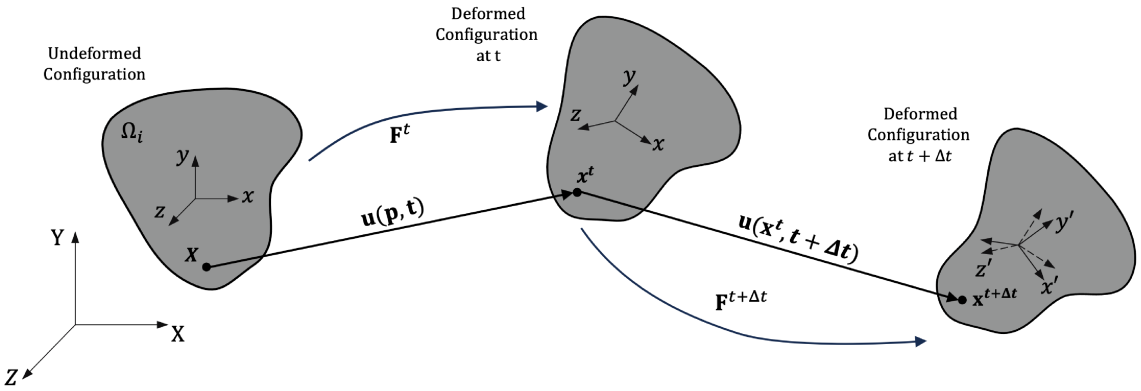

FIGURE 2 | Motion of a solid subdomain Ω_i in its original configuration and deformed configurations at times t and $t + \Delta t$.

It is often convenient to use the formulations in terms of rate quantities, in particular for modelling elasto-plastic behaviour [28]. De Souza Neto, Peric, and Owen [29] The rate of deformation $\mathbf{D}$ is considered as the measure of strain. The symmetric part of the deformation gradient corresponds to $\mathbf{D}$, whereas its anti-symmetric part corresponds to $\mathbf{W}$, the continuum spin

$$\mathbf{D} = \text{sym}\left(\frac{\partial \mathbf{v}}{\partial \mathbf{x}}\right) = \frac{1}{2}\left(\frac{\partial \mathbf{v}}{\partial \mathbf{x}} + \frac{\partial \mathbf{v}}{\partial \mathbf{x}}^T\right)$$
$$\mathbf{W} = \text{skew}\left(\frac{\partial \mathbf{v}}{\partial \mathbf{x}}\right) = \frac{1}{2}\left(\frac{\partial \mathbf{v}}{\partial \mathbf{x}} - \frac{\partial \mathbf{v}}{\partial \mathbf{x}}^T\right) \tag{7}$$

Note that $\mathbf{D}$ and $\mathbf{W}$ are defined with the spatial field $\partial \mathbf{v}/\partial \mathbf{x}$, the velocity gradient

$$\frac{\partial \mathbf{v}}{\partial \mathbf{x}} = (\text{grad } \mathbf{v})^T = \dot{\mathbf{F}}\mathbf{F}^{-1} \tag{8}$$

To obtain the stresses from the constitutive model, integration of $\dot{\boldsymbol{\sigma}}$ is required. For the majority of the analysis, small elastic strains are assumed, hence the total rate of deformation $\mathbf{D}$ can be decomposed into elastic $\mathbf{D}^e$ and plastic $\mathbf{D}^p$ parts. The same is assumed of the spin.

$$\mathbf{D} = \mathbf{D}^e + \mathbf{D}^p; \quad \mathbf{W} = \mathbf{W}^e + \mathbf{W}^p \tag{9}$$

The chosen plastic flow rule determines $\mathbf{D}^p$. The definition of this is elaborated upon in the hypoelastic-plastic numerical example later in the paper. For now, this relation allows for the rate of elastic deformation to be re-written as $\mathbf{D}^e = \mathbf{D} - \mathbf{D}^p$. Following which the rate of stress $\dot{\boldsymbol{\sigma}}$ is determined from the objective Jaumann stress rate $\overset{\nabla}{\boldsymbol{\sigma}}$ using Hooke's law and the rate of elastic deformation

$$\overset{\nabla}{\boldsymbol{\sigma}} = 2G\mathbf{D}^e + \lambda \text{Tr}(\mathbf{D}^e)\mathbf{I} \tag{10}$$

with G as the shear modulus and λ as Lamé's first constant. Thus, the final material rate of the Cauchy stress tensor is as follows

$$\dot{\boldsymbol{\sigma}} = \overset{\nabla}{\boldsymbol{\sigma}} + \mathbf{W}\boldsymbol{\sigma} + \boldsymbol{\sigma}\mathbf{W}^{\mathbf{T}} \tag{11}$$

The derivative of Cauchy stress is therefore shown to be separable into the rate of change in material response and the change of stress due to rotation. At this point, the contributions of previous stresses can be summed to calculate the total stress as one integrates stress through time. These now provide the building blocks for deformation. Next, we discretise the governing equations to describe this mechanical behaviour for each subdomain.

2.2 | Variational Formulation and Finite Element Discretisation

The variational formulation of the governing equations describe the equilibrium of each subdomain Ω_i. Suppose δv denotes a variational velocity function in a space $\mathcal{V}_0$ where $\delta \dot{\mathbf{u}} \in \mathcal{V}_0, \forall \delta \dot{\mathbf{u}} \in H^1(\Omega)$, then the variational formulation for subdomain Ω_i is as follows:

$$\int_{\Omega_i} \rho_i \ddot{\mathbf{u}}_i \cdot \delta \dot{\mathbf{u}} dV = \int_{\Omega_i} \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}_i \cdot \delta \dot{\mathbf{u}} dV + \int_{\Omega_i} \rho_i \mathbf{b}_i \cdot \delta \dot{\mathbf{u}} dV \tag{12}$$

$$\int_{\Omega_i} \rho_i \ddot{\mathbf{u}}_i \cdot \delta \dot{\mathbf{u}} dV = \int_{\Omega_i} \boldsymbol{\sigma}_i \cdot \mathbf{n} \cdot \delta \dot{\mathbf{u}} dV - \int_{\Omega_i} \boldsymbol{\sigma}_i : \mathbf{D}_i dV + \int_{\Omega_i} \rho_i \mathbf{b}_i \cdot \delta \dot{\mathbf{u}} dV \tag{13}$$

$$\int_{\Omega_i} \rho_i \ddot{\mathbf{u}}_i \cdot \delta \dot{\mathbf{u}} dV = \int_{\partial\Omega_i} \mathbf{t}_i \cdot \delta \dot{\mathbf{u}} dA - \int_{\Omega_i} \boldsymbol{\sigma}_i : \mathbf{D}_i dV + \int_{\Omega_i} \rho_i \mathbf{b}_i \cdot \delta \dot{\mathbf{u}} dV \tag{14}$$

In a two-domain case, Ω_L and Ω_s are discretised with $\mathcal{N}^L$ and $\mathcal{N}^s$ number of finite elements. The spatial description of the domain results in a second-order hyperbolic ordinary differential equation for an undamped structurally dynamic system. The discrete approximation of the variational form is given by the following equation of motion:

$$\mathbf{M}_i \ddot{\mathbf{u}}_i = \mathbf{f}_i^{\text{ext}} - \mathbf{f}_i^{\text{int}} \tag{15}$$

where

$$\mathbf{M}_i = \sum_{e=1}^{N_e} \int_{\Omega^e} \rho \mathbf{N}^T \mathbf{N} dV_e; \quad \mathbf{f}_i^{\text{ext}} = \sum_{e=1}^{N_e} \int_{\Omega^e} \mathbf{N}^T \mathbf{t} dV_e;$$
$$\mathbf{f}_i^{\text{int}} = \sum_{e=1}^{N_e} \int_{\Omega^e} \mathbf{B}^T \boldsymbol{\sigma} dV_e \tag{16}$$

where $\mathbf{N}$ indicates the linear Lagrange basis functions and the relation $\mathbf{D} = \mathbf{B}\dot{\mathbf{u}}$ is employed. Both mass matrices $\mathbf{M}_i$ and force matrices $\mathbf{f}_i$ are evaluated over their current configuration in an updated Lagrangian formulation, where we evaluate the integrals in Equations (16) over elemental volume V_e. To avoid solving a system of linear equations within the explicit form of finite elements, the mass matrix is lumped. We discretise temporally with the use of leapfrog integration, where the kinematic quantities are staggered so that acceleration $\ddot{\mathbf{u}}$ is calculated at time step n, velocity $\dot{\mathbf{u}}$ at $n + \frac{1}{2}$ and displacement $\mathbf{u}$ at $n + 1$. Consequentially each subdomain Ω_i advances through time with

$$\ddot{\mathbf{u}}_i^n = \frac{(\mathbf{f}_i^{\text{ext}} - \mathbf{f}_i^{\text{int}})}{\mathbf{M}_i}; \quad \dot{\mathbf{u}}_i^{n+1/2} = \dot{\mathbf{u}}_i^{n-1/2} + \ddot{\mathbf{u}}_i^n \cdot \Delta t_i$$
$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \dot{\mathbf{u}}_i^{n+1/2} \cdot \Delta t_i \tag{17}$$

To ensure the stability of our method, we need the time step Δt to be less than or equal to the critical time step Δt_C. The critical time step Δt_C is required for the stability of the central difference based methods applied to a linear undamped system, which has the form

$$\Delta t_C = \frac{2}{\omega_C} \leq \min_e \left(\frac{h_e}{c_e}\right) \tag{18}$$

where ω_C is the maximum eigenfrequency of the problem, h_e is the characteristic length of the element, and c_e is the dilatational (longitudinal) wave speed. Generally, for the wave propagation in

elastic solids, the speed of the dilatational wave c_e, is significantly above that of shear and Rayleigh waves. From this definition of the critical time step, it is clear that both the spatial discretisation and material properties affect the speed at which the numerical solution is found. In the following sections, material heterogeneity drives the latter, and will be the focus of the article.

2.3 | Two-Domain Multi-Time Step Integration Algorithm

In the following section, we propose a multi-time step coupling method for two subdomains; Ω_L and Ω_s marching with large, Δt_L and small time step, Δt_s, respectively. Figure 3 depicts the system variable vectors that are calculated for the case where three small time steps synchronise with one large time step. A complete summary is shown in the appendix with Algorithm 3 for reference. There are three distinct features to the algorithm. They include the explicit solution of the subdomain interface problem, the order of integration with the definition of time step ratios, and the comparison of a time step reduction factor α_i.

2.3.1 | Explicit Solution of the Interface Problem

We start with the solution of the interface problem between Ω_L and Ω_s at a synchronised time step $t_L = t_s$. The advantage of this formulation is that we do not require the inversion of stiffness matrices, interpolation of interface kinematics, or the solution of the system of equations across subdomains. Large and small domains assemble the following system variable vectors;

$$\begin{aligned} \mathbf{M}_L &= \sum_{e=1}^{N_e} \int_{\Omega_L^e} \rho \mathbf{N}^T \mathbf{N} dV_e; \quad \mathbf{f}_L^{\text{ext}} = \sum_{e=1}^{N_e} \int_{\Omega_L^e} \mathbf{N}^T \mathbf{t} dV_e \\ \mathbf{f}_L^{\text{int}} &= \sum_{e=1}^{N_e} \int_{\Omega_L^e} \mathbf{B}^T \boldsymbol{\sigma} dV_e \end{aligned} \tag{19}$$

$$\begin{aligned} \mathbf{M}_s &= \sum_{e=1}^{N_e} \int_{\Omega_s^e} \rho \mathbf{N}^T \mathbf{N} dV_e; \quad \mathbf{f}_s^{\text{ext}} = \sum_{e=1}^{N_e} \int_{\Omega_s^e} \mathbf{N}^T \mathbf{t} dV_e \\ \mathbf{f}_s^{\text{int}} &= \sum_{e=1}^{N_e} \int_{\Omega_s^e} \mathbf{B}^T \boldsymbol{\sigma} dV_e \end{aligned} \tag{20}$$

On the interface between the subdomains we compute the assembled interface mass as

$$\mathbf{M}_\Gamma = (\mathbf{C}_s^T \mathbf{M}_s \mathbf{C}_s) + (\mathbf{C}_L^T \mathbf{M}_L \mathbf{C}_L) \tag{21}$$

where $\mathbf{C}_s$ and $\mathbf{C}_L$ are indicator vectors in 1-D, and matrices in 2-D and 3-D, that extract the interface degrees of freedom belonging to Ω_s and Ω_L respectively. We make use of the added-mass technique, similar to that of the works of Banks, Henshaw, and Schwendeman [30]. The external and internal force vectors on the interface can also be summated as

$$\mathbf{f}_\Gamma^{\text{int}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{int}} + \mathbf{C}_L^T \mathbf{f}_L^{\text{int}}; \quad \mathbf{f}_\Gamma^{\text{ext}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{ext}} + \mathbf{C}_L^T \mathbf{f}_L^{\text{ext}} \tag{22}$$

This then allows for the interface acceleration to be explicitly computed as

$$\ddot{\mathbf{u}}_\Gamma = \mathbf{M}_\Gamma^{-1} (\mathbf{f}_\Gamma^{\text{ext}} - \mathbf{f}_\Gamma^{\text{int}}) \tag{23}$$

where inversion of $\mathbf{M}_\Gamma$ is computationally inexpensive if the masses are lumped. Thus it follows that,

$$\ddot{\mathbf{u}}_\Gamma = \ddot{\mathbf{u}}_{\Gamma s} = \ddot{\mathbf{u}}_{\Gamma L} \tag{24}$$

where $\ddot{\mathbf{u}}_\Gamma$ is prescribed each Δt_s on the small subdomain's interface, until the next re-calculation at Δt_L. For a mass-conserving problem, the mass need only be added at $t = 0$, however $\mathbf{f}_{\Gamma_i}^{\text{int}}$ requires the summation after the advancement of each large time step. The continuity of acceleration and tractions is ensured for adjoining subdomains with the imposed condition.

2.3.2 | Order of Integration With Time Step Ratios

We solve Equation (15) over each subdomain with the relevant time step. The two subdomains start synchronised in time, $t_s^n = t_L^N$ and $k = 0$, where N and n are the respective time steps of the large and small subdomains, and k is the number of small time steps that elapsed since the last synchronisation ($t_s^n = t_L^N$). Before any integration, the algorithm computes the largest stable time step for each subdomain with the *trial time* t_{Ti} of each subdomain:

$$t_{Ts}^{n+k} = t_s + \Delta t_{Cs}; \quad t_{TL}^{N+1} = t_L + \Delta t_{CL} \tag{25}$$

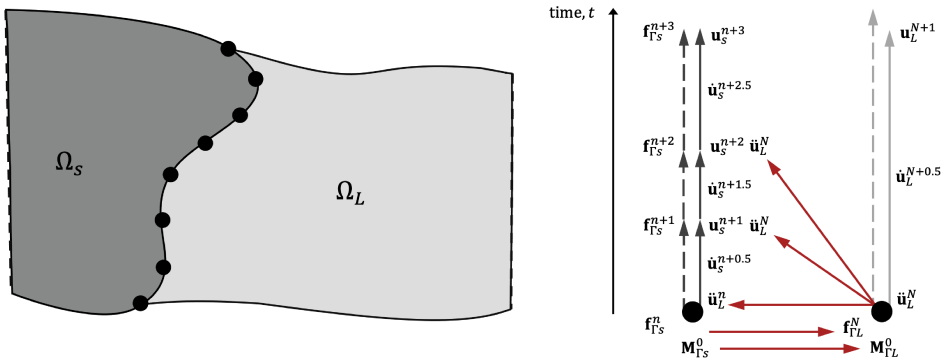

FIGURE 3 | A pictorial description of a full integration step for a coupling node on Γ solved with three-time steps on the small subdomain to one large step with $t_s^{n+3} = t_L^{N+1}$. The dashed lines represent the computed trial time T_{Ti}, whereas the solid lines represent the time step used for the integration. Summation of forces $\mathbf{f}_\Gamma$ and masses $\mathbf{M}_\Gamma^0$ take place on the interface.

where the CFL condition is utilised to determine Δt_{Cs} and Δt_{CL}. Next, we define the *current time step ratio*, t_{ratio}^{n+k}, and *next time step ratio*, t_{ratio}^{n+k+1} such that

$$t_{\text{ratio}}^{n+k} = \frac{t_s^{n+k} - t_L^N}{t_{TL}^{N+1} - t_L^N}; \quad t_{\text{ratio}}^{n+k+1} = \frac{(t_s^{n+k} + \Delta t_s^{n+k}) - t_L^N}{t_{TL}^{N+1} - t_L^N} \tag{26}$$

We then look to calculate the kinematic quantities and enforce the continuity of acceleration with

$$\ddot{\mathbf{u}}_s^n = (\mathbf{I} - \mathbf{C}_s\mathbf{C}_s^T)\tilde{\ddot{\mathbf{u}}}_s^n + (\mathbf{C}_s\mathbf{L}_s)\ddot{\mathbf{u}}_\Gamma^N \tag{27}$$

where the unconstrained acceleration is $\tilde{\ddot{\mathbf{u}}}_s^n = \mathbf{M}_s^{-1}(\mathbf{f}_s^{\text{ext}} - \mathbf{f}_s^{\text{int}})$ and $\mathbf{L}_s$ is a square incidence matrix relating the interface degrees of freedom to the small global vectors. The staggered leapfrog time integration scheme is then used to compute velocity and displacement on the small subdomain at $t_s^{n+k+\frac{1}{2}}$ and t_s^{n+k+1} with:

$$\dot{\mathbf{u}}_s^{n+k+\frac{1}{2}} = \dot{\mathbf{u}}_s^{n+k-\frac{1}{2}} + \ddot{\mathbf{u}}_s^n \cdot \Delta t_s \tag{28}$$

$$\mathbf{u}_s^{n+k+1} = \mathbf{u}_s^{n+k} + \dot{\mathbf{u}}_s^{n+k+\frac{1}{2}} \cdot \Delta t_s \tag{29}$$

Above, it is shown that the interface velocities and displacements are updated with the subdomain Ω_s. This contrasts to the likes of Dvorak et al. [24] where subdomains and their regions require different time steps. Later we show the continuity of velocity and displacement for the case of matching meshes.

$$\mathbf{C}_i^T\dot{\mathbf{u}}_i^{n+\frac{1}{2}} - \mathbf{L}_i\dot{\mathbf{u}}_\Gamma^{n+\frac{1}{2}} = 0 \tag{30}$$

$$\mathbf{C}_i^T\mathbf{u}_i^{n+1} - \mathbf{L}_i\mathbf{u}_\Gamma^{n+1} = 0 \tag{31}$$

Following the update of Ω_s to $n+1$, we evaluate the temporal quantities with the time step ratios t_{ratio}^{n+k} and t_{ratio}^{n+k+1}. The condition of $t_{\text{ratio}}^{n+k+1} \leq 1$ or ($t_{\text{ratio}}^{n+k} \leq 1$ and $t_{\text{ratio}}^{n+k+1} \leq 1$) determines if more updates on Ω_s at Δt_{CS} are necessary and thus the return to Equation (27). The computation of trial time step and the actual time step taken is illustrated in Figure 3 by the dotted and solid line respectively. The recalculation of trial time is advantageous for highly dynamic problems, where the time step may significantly change, as shown by the variation in the length of the arrows. It is important to compute the t_{ratio}^{n+k}, where a tolerance, $TOL = 1e-6$ has been used, solely for the case when $\Delta t_s \approx \Delta t_L$.

2.3.3 | Comparison of Time Step Reduction Factor

The final salient feature of the proposed algorithm selects the time for synchronisation between the different subdomains, chosen to minimise the deviation from the trial time t_{TL} and t_{Ts}. In an updated Lagrangian formulation, as the small subdomain's time catches up with the large subdomain's time, its time step Δt_s^{n+k} can change, therefore the synchronisation time is not known a priori, and needs to be computed at every small time step. Once the small trial time, overshoots the large trial time t_{TL}^{N+1}, two ratios α_L and α_s are computed, where $0 \leq \alpha \leq 1$. These represent the reduction factors to apply on large or small subdomains, to enforce synchronisation in time. We compute the time step reduction factors as:

$$\alpha_L = 1 - \frac{(t_{TL}^N - t_s^{n+k})}{(t_{TL}^{N+1} - t_L^N)}; \quad \alpha_s = 1 - \frac{(t_{Ts}^{n+k} - t_{TL}^{N+1})}{(t_{Ts}^{n+k+1} - t_s^{n+k})} \tag{32}$$

The method minimises the deviation from the CFL condition on just one domain while the other domain's time step remains unchanged. Figure 4 depicts the two possible cases of applying the reduction factor α_i on either a large or small subdomain.

Case 1: $\alpha_s \geq \alpha_L$

$$\ddot{\mathbf{u}}_s^{n+1} = (\mathbf{I} - \mathbf{C}_s\mathbf{C}_s^T)\tilde{\ddot{\mathbf{u}}}_s^{n+1} + (\mathbf{C}_s\mathbf{L}_s)\ddot{\mathbf{u}}_\Gamma^N \tag{33}$$

$$\dot{\mathbf{u}}_s^{n+3/2} = \dot{\mathbf{u}}_s^{n+1/2} + \ddot{\mathbf{u}}_s^{n+1} \cdot \alpha_s\Delta t_s \tag{34}$$

$$\mathbf{u}_s^{n+2} = \mathbf{u}_s^{n+1} + \dot{\mathbf{u}}_s^{n+3/2} \cdot \alpha_s\Delta t_s \tag{35}$$

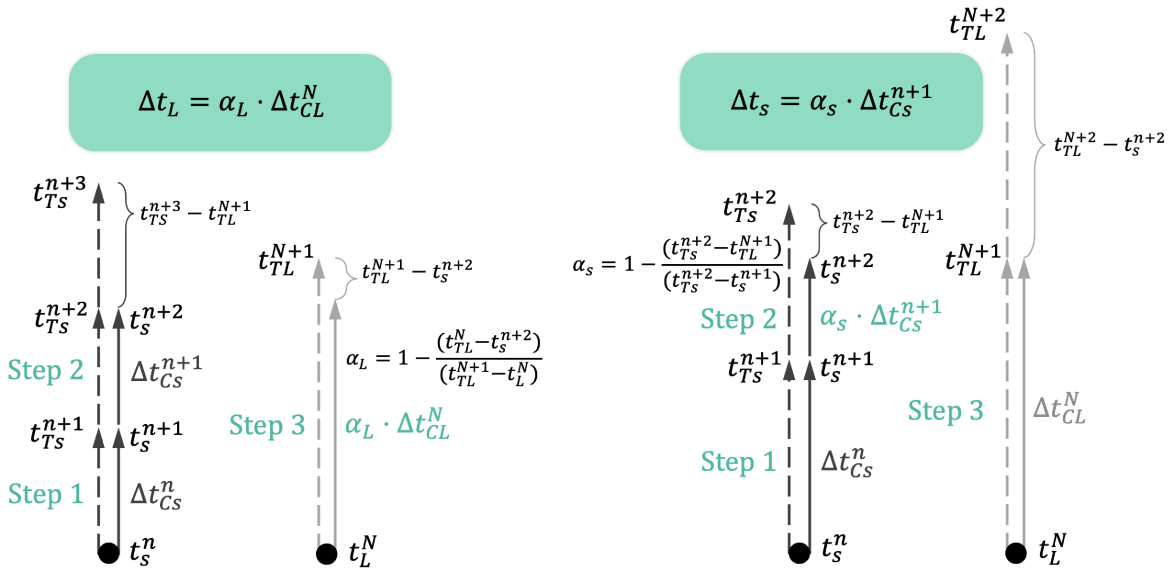

FIGURE 4 | Left: Case 2–Reduction Factor applied on large subdomain for $m \approx 2.4$ when $\alpha_L > \alpha_s$. The large subdomain time step is reduced to synchronise with the small time step; Right: Case 1–Reduction Factor applied on small subdomain for $m \approx 1.6$ when $\alpha_s \geq \alpha_L$. The small subdomain's second time step is reduced to synchronise with the large time step.

where these quantities over Ω_L can be computed via the following Equations (27), (28), and (29), through switching subscripts from s to L. The application of the reduction factor on the large subdomain gives:

Case 2: $\alpha_L > \alpha_s$

$$\ddot{\mathbf{u}}_L^N = (\mathbf{I} - \mathbf{C}_L\mathbf{C}_L^T)\tilde{\ddot{\mathbf{u}}}_L^N + (\mathbf{C}_L\mathbf{L}_L)\ddot{\mathbf{u}}_\Gamma^N \tag{36}$$

$$\tilde{\ddot{\mathbf{u}}}_L^N = \mathbf{M}_\Gamma^{-1}(\mathbf{f}_L^{\text{ext}} - \mathbf{f}_L^{\text{int}}) \tag{37}$$

$$\dot{\mathbf{u}}_L^{N+1/2} = \dot{\mathbf{u}}_s^{N-1/2} + \ddot{\mathbf{u}}_L^N \cdot \alpha_L \Delta t_L \tag{38}$$

$$\mathbf{u}_L^{N+1} = \mathbf{u}_L^N + \dot{\mathbf{u}}_L^{N+1/2} \cdot \alpha_L \Delta t_L \tag{39}$$

In both cases described, the two subdomains will always find a common time, where synchronisation can take place. Consequently, we have shown $t_s^n = t_L^N$, and after integrating over the following large time step $t_L^{N+1} = t_s^{n+k}$. At the same time, the coupled domains will minimise the deviation from the maximum possible time step.

$$t_L^{N+1} = t_s^{n+k} = \begin{cases} t_s^n + \sum_{k=0}^{k-1} \Delta t_{Cs}^{n+k} + \alpha_s \cdot \Delta t_{Cs}^{n+k}, & \alpha_s > \alpha_L \\ t_s^n + \alpha_L \cdot \Delta t_{CL}^N, & \alpha_L \geq \alpha_s \end{cases} \tag{40}$$

In Figure 4, the difference in the trial times is represented with the braces around the dotted and solid arrows in time. Both can be found in the numerators of the ratios. A summary of a singular computational cycle over a large time step is given below in Algorithm 1, as well as the full algorithm for the two-domain over multi-time steps in Algorithm 3.

ALGORITHM 1 | A Two-Domain Multi-Time Integration Step.

while $t_{\text{ratio}}^{n+k+1} \leq 1$ or ($t_{\text{ratio}}^{n+k} \leq 1$ and $t_{\text{ratio}}^{n+k+1} \leq 1 + TOL$) **do**
  Integrate small domain Ω_s with $\ddot{\mathbf{u}}_\Gamma$ and compute internal force and external force vectors $\mathbf{f}_s^{\text{int}}, \mathbf{f}_s^{\text{ext}}$
  Compute trial times t_{Ts}^{n+k}, t_{TL}^{N+1} and time step ratios t_{ratio}^{n+k}, t_{ratio}^{n+k+1}
end while
Compute time step reduction factors α_L, α_s
if $\alpha_L \geq \alpha_s$ **then**
  $\Delta t_L = \alpha_L \cdot \Delta t_L$
else
  $\Delta t_s = \alpha_s \cdot \Delta t_s$
  Integrate small domain Ω_s with $\ddot{\mathbf{u}}_\Gamma$ and recompute internal force and external force vectors $\mathbf{f}_s^{\text{int}}, \mathbf{f}_s^{\text{ext}}$
  Recompute trial times t_{Ts}^{n+k}, t_{TL}^{N+1} and time step ratios t_{ratio}^{n+k}, t_{ratio}^{n+k+1}
end if
Integrate Large domain Ω_L with $\ddot{\mathbf{u}}_\Gamma$ and compute internal force and external force vectors $\mathbf{f}_L^{\text{int}}, \mathbf{f}_L^{\text{ext}}$
Compute interface acceleration $\ddot{\mathbf{u}}_\Gamma$ with Equation (23) for next time step
Recompute trial times t_{Ts}^{n+k}, t_{TL}^{N+1} and time step ratios t_{ratio}^{n+k}, t_{ratio}^{n+k+1}

3 | On Stability and Drifting

This section reviews the use of localised Lagrange multipliers (LLMs) in comparison to our algorithm. We verify the stability of our method over both the large time step and small time step domains. We show the continuity of velocity and then displacement, to prohibit drifting. We demonstrate the behaviour of our algorithms in extreme conditions via the square wave benchmark through a 1-D heterogeneous bar.

3.1 | Comparison With Lagrange Multiplier Methods

Many of the previous works mentioned utilise Lagrange multipliers [11, 13, 16, 20, 31], where specifically those that enforce the continuity of acceleration [17, 24] solve the following equations of motion for a two subdomain problem

$$\mathbf{M}_L\ddot{\mathbf{u}}_L + \mathbf{C}_L\lambda_L = \mathbf{f}_L^{\text{ext}} - \mathbf{K}_L\mathbf{u}_L; \quad \mathbf{M}_s\ddot{\mathbf{u}}_s + \mathbf{C}_s\lambda_s = \mathbf{f}_s^{\text{ext}} - \mathbf{K}_s\mathbf{u}_s \tag{41}$$

where λ_L and λ_s represent the Lagrange multiplier fields and $\mathbf{K}_L$ and $\mathbf{K}_s$ denote the global stiffness matrices on Ω_L and Ω_s, respectively. The LLM method also introduces a frame, denoted by f, that enforces the kinematic continuity equations.

$$\mathbf{C}_L^T\ddot{\mathbf{u}}_L - \mathbf{L}_L\ddot{\mathbf{u}}_f = 0; \quad \mathbf{C}_s^T\ddot{\mathbf{u}}_s - \mathbf{L}_s\ddot{\mathbf{u}}_f = 0 \tag{42}$$

and the equilibrium over the interface is ensured by the Lagrange multiplier force loads by

$$\mathbf{L}_L^T\lambda_L + \mathbf{L}_s^T\lambda_s = 0 \tag{43}$$

Thus, at every small time step, we obtain an interface system of equations in the form

$$\begin{bmatrix} \mathbf{C}_L^T\mathbf{M}_L^{-1}\mathbf{C}_L & 0 & \mathbf{L}_L \\ 0 & \mathbf{C}_s^T\mathbf{M}_s^{-1}\mathbf{C}_s & \mathbf{L}_s \\ \mathbf{L}_L^T & \mathbf{L}_s^T & 0 \end{bmatrix} \begin{bmatrix} \lambda_L^{n+k} \\ \lambda_s^{n+k} \\ \ddot{\mathbf{u}}_f^{n+k} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_L^T\tilde{\ddot{\mathbf{u}}}_L^{n+k} \\ \mathbf{C}_s^T\tilde{\ddot{\mathbf{u}}}_s^{n+k} \\ 0 \end{bmatrix} \tag{44}$$

The unknowns of the frame are evaluated to obtain

$$\lambda_L^{n+k} = (\mathbf{C}_L^T\mathbf{M}_L^{-1}\mathbf{C}_L)^{-1}(\mathbf{C}_L^T\tilde{\ddot{\mathbf{u}}}_L^{n+k} - \mathbf{L}_L\ddot{\mathbf{u}}_f^{n+k}) \tag{45}$$

$$\lambda_s^{n+k} = (\mathbf{C}_s^T\mathbf{M}_s^{-1}\mathbf{C}_s)^{-1}(\mathbf{C}_s^T\tilde{\ddot{\mathbf{u}}}_s^{n+k} - \mathbf{L}_s\ddot{\mathbf{u}}_f^{n+k}) \tag{46}$$

$$\ddot{\mathbf{u}}_f^{n+k} = \left[\sum_{i=1}^{2}\left(\mathbf{L}_i^T(\mathbf{C}_i^T\mathbf{M}_i^{-1}\mathbf{C}_i)^{-1}\mathbf{L}_i\right)\right]^{-1} \times \sum_{i=1}^{2}\left(\mathbf{L}_i^T(\mathbf{C}_i^T\mathbf{M}_i^{-1}\mathbf{C}_i)^{-1}\mathbf{C}_i^T\tilde{\ddot{\mathbf{u}}}_i^{n+k}\right) \tag{47}$$

where λ_L^{n+k} and λ_s^{n+k} are computed each Δt_s, and hence at every step k. The same can be said for the frame acceleration $\ddot{\mathbf{u}}_f$. Finally, the substitution of the unconstrained acceleration allows for the rearrangement of Equation (41) to obtain the constrained acceleration on Ω_L and Ω_s.

$$\ddot{\mathbf{u}}_L^{n+k} = \mathbf{M}_L^{-1}(\mathbf{f}_L^{\text{ext}} - \mathbf{K}_L\mathbf{u}_L) - \mathbf{M}_L^{-1}\mathbf{C}_L\lambda_L^{n+k} = \tilde{\ddot{\mathbf{u}}}_L^{n+k} - \mathbf{M}_L^{-1}\mathbf{C}_L\lambda_L^{n+k} \tag{48}$$

$$\ddot{\mathbf{u}}_s^{n+k} = \mathbf{M}_s^{-1}(\mathbf{f}_L^{\text{ext}} - \mathbf{K}_s\mathbf{u}_s) - \mathbf{M}_s^{-1}\mathbf{C}_s\lambda_s^{n+k} = \tilde{\ddot{\mathbf{u}}}_s^{n+k} - \mathbf{M}_s^{-1}\mathbf{C}_s\lambda_s^{n+k} \tag{49}$$

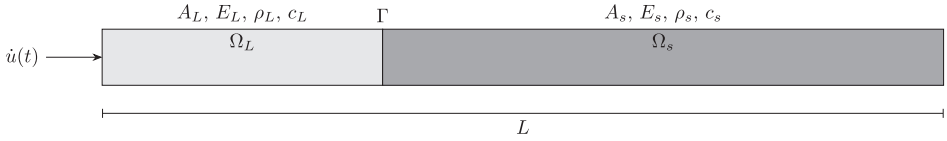

FIGURE 5 | A one-dimensional heterogeneous domain Ω split into large Ω_L and small time step subdomain Ω_s of length $L_L = 50$ mm and $L_s = 100$ mm with an axial square wave velocity boundary condition where $\nu = 0$.

Next, we show the effects of our imposed acceleration $\ddot{\mathbf{u}}_\Gamma$ across Δt_L and its ability to resolve the interface. The LLMs are the forces used to couple the subdomains, thus allowing the analysis of the energy across the interface Γ through time. Subsequently, we demonstrate this with a simple 1-D example.

3.2 | Square Wave Propagation Through Heterogeneous Bar

We choose an elastic discontinuous wave propagation problem to compare the proposed algorithm to those found in the literature. Suppose a domain Ω is split into two subdomains Ω_L and Ω_s with isotropic elastic properties of $E_L = 0.02$ GPa, $E_s = \rho\left(\frac{\pi}{0.02}\right)^2$ GPa and a singular density $\rho = 8000$ kgm^{-3} for both subdomains. Both subdomains share the same spatial discretisation, and have consecutive lengths $L_L = 50$ mm and $L_s = 100$ mm. A prescribed velocity boundary condition is applied to the large domain at $x = 0$ with a square wave function, where we define an equivalent sine period of $L_L/2 \cdot \sqrt{\rho/E_L}$. The velocity function reveals that the forward travelling elastic wave travels half of Ω_L, by the time the pulse is fully applied. Both Ω_L and Ω_s are meshed with the same $h_e = 0.1667$ mm, where time steps initially integrate with $\Delta t_L = 1.667e-06$ s and $\Delta t_s = 5.305e-07$ s with $Co = 0.5$ for both subdomains. Further evaluations on time step are made in the following Section 4 (Figure 5).

3.3 | Stability Checks via Energy Conservation

In this section, we present the results related to the stability of the method. The square wave benchmark exemplifies the solution of the interface system of equations with Lagrange multipliers. We compute $\ddot{\mathbf{u}}_f$ via the LLMs in Equation (47), then assume

$$\ddot{\mathbf{u}}_\Gamma^{N+1} = \ddot{\mathbf{u}}_f^{N+1} \tag{50}$$

These coupling forces are equivalent to the LLMs, λ, or referred to as linking forces in [15, 31], from the imposed acceleration at the interface. From the same set of Equation (44), the Lagrange multipliers can be shown to satisfy

$$\mathbf{L}_L^T \lambda_L^{N+1} = -\mathbf{L}_s^T \lambda_s^{N+1} \tag{51}$$

We first solve the interface equations of motion in Equation (44) at each large time step Δt_L. The stability of the overall system can be evaluated from the perspective of the energy method as initially proposed by Hughes and Liu [32] and adopted in subsequent works [33]. Gravouil and Combescure [15]. First assessing for the two subdomains Ω_L and Ω_s, the energy method gives

$$\begin{aligned} &W_{\text{ext},L} - W_{\text{int},L} - W_{\text{kin},L} + W_{\text{ext},s} - W_{\text{int},s} - W_{\text{kin},s} \\ &\quad = W_{Link,L} + W_{\text{Link},s} \end{aligned} \tag{52}$$

where each energy term is expressed generally for subdomain Ω_i as

$$W_{\text{int},i} = \frac{1}{2}\mathbf{u}_i^T(\mathbf{f}_{\text{int},i}); \quad W_{\text{ext},i} = \frac{1}{2}\mathbf{u}_i^T(\mathbf{f}_{\text{ext},i}) \quad W_{\text{kin},i} = \frac{1}{2}(\dot{\mathbf{u}}_i)^T\mathbf{M}\dot{\mathbf{u}}_i \tag{53}$$

and the link energy, coupling the two subdomains is referred to as

$$W_{Link,i} = \frac{1}{2}(\mathbf{C}_i^T\mathbf{u}_i)\lambda_i \tag{54}$$

Isolating just the coupling energy W_{Link} over a large time step from N to $N+1$, we can calculate energy associated to the coupling forces over an increment with

$$\begin{aligned} W_{Link}^{N+1} &= W_{Link,L}^N + \delta W_{Link,L} + W_{Link,s}^N + \delta W_{Link,s} \\ &= W_{Link,L}^N + \frac{1}{2}(\mathbf{C}_L^T\mathbf{u}_L^{N+1} - \mathbf{C}_L^T\mathbf{u}_L^N)^T(\lambda_L^N + \lambda_L^{N+1}) \\ &\quad + W_{Link,s}^N + \frac{1}{2}(\mathbf{C}_s^T\mathbf{u}_s^{N+1} - \mathbf{C}_s^T\mathbf{u}_s^N)^T(\lambda_s^N + \lambda_s^{N+1}) \end{aligned} \tag{55}$$

However, to comprehensively evaluate stability, changes over Δt_s must be accounted for. Coupling forces λ_L^{n+k} at Δt_s must be recovered to compute the increments of energy. Accounting for the variation in kinematic quantities over Δt_s we obtain

$$\begin{aligned} W_{Link}^{n+1} &= W_{Link,L}^N + \delta W_{Link,L} + W_{Link,s}^n + \delta W_{Link,s} \\ &= W_{Link,L}^N + \frac{1}{2}(\mathbf{C}_L^T\mathbf{u}_L^{N+1} - \mathbf{C}_L^T\mathbf{u}_L^N)^T(\lambda_L^N + \lambda_L^{N+1}) \\ &\quad + W_{Link,s}^n + \sum_{k=0}^{m}\frac{1}{2}(\mathbf{C}_s^T\mathbf{u}_s^{n+k+1} - \mathbf{C}_s^T\mathbf{u}_s^{n+k})^T(\lambda_s^{n+k} + \lambda_s^{n+k+1}) \end{aligned} \tag{56}$$

The unknowns of the system now become $\mathbf{f}_{\text{int},L}^{n+k+1}$, λ_s^{n+k+1} and λ_L^{n+k+1}. For the benchmark where $m = \pi$, the LLMs computed at large time steps solve the same interface problem so that

$$\mathbf{L}_L^T\lambda_L^{N+1} + \mathbf{L}_s^T\lambda_s^{N+1} = \mathbf{L}_L^T\lambda_L^{n+3} + \mathbf{L}_s^T\lambda_s^{n+3} = 0 \tag{57}$$

In the left of Figure 6, we plot the increment in the coupling energy for both subdomains. For this benchmark each increment in $\delta W_{Link,L}$ requires the summation of three increments of $\delta W_{Link,s}$. Increments δW_{Link} are computed at each Δt_L for Ω_L and Δt_s for Ω_s.

The equilibrium of forces on the interface nodes Γ, can be used to evaluate the energy across the interface through time. The forces acting on Γ_L and Γ_s are balanced as

$$\begin{aligned} &\mathbf{C}_L^T\lambda_L^{N+1} + (\mathbf{C}_L^T\mathbf{M}_L^{-1}\mathbf{C}_L)^{-1}\ddot{\mathbf{u}}_\Gamma^{N+1} + \mathbf{C}_L^T\mathbf{f}_{\text{int},L}^{N+1} \\ &\quad = \mathbf{C}_s^T\lambda_s^{N+1} + (\mathbf{C}_s^T\mathbf{M}_s^{-1}\mathbf{C}_s)^{-1}\ddot{\mathbf{u}}_\Gamma^{N+1} + \mathbf{C}_s^T\mathbf{f}_{\text{int},s}^{N+1} = 0 \end{aligned} \tag{58}$$

where $\mathbf{f}_{\text{int},L}$ and $\mathbf{f}_{\text{int},s}$ are the quantities used to compute the unconstrained accelerations $\tilde{\ddot{\mathbf{u}}}_i$ on each subdomain. Our proposed algorithm does not require the solution of the interface problem on both Γ_L and Γ_s at Δt_s, rather we solve just over Γ_s. Using the force equilibrium in Equation (58), over Δt_L the energy across the interface Γ is

$$\begin{aligned} W_\Gamma^{N+1} &= W_{\Gamma L}^N + \delta W_{\Gamma L} + W_{\Gamma s}^N + \delta W_{\Gamma s} = 0 \\ &= W_{\Gamma L}^N + \frac{1}{2}(\mathbf{C}_L^T\mathbf{u}_L^{N+1} - \mathbf{C}_L^T\mathbf{u}_L^N)^T\Big(\mathbf{C}_L^T(\lambda_L^N + \lambda_L^{N+1}) \\ &\quad +(\mathbf{C}_L^T\mathbf{M}_L^{-1}\mathbf{C}_L)^{-1}(\ddot{\mathbf{u}}_\Gamma^N + \ddot{\mathbf{u}}_\Gamma^{N+1}) + \mathbf{C}_L^T(\mathbf{f}_{\text{int},L}^N + \mathbf{f}_{\text{int},L}^{N+1})\Big) \\ &\quad + W_{\Gamma s}^N + \frac{1}{2}(\mathbf{C}_s^T\mathbf{u}_s^{N+1} - \mathbf{C}_s^T\mathbf{u}_s^N)^T\Big(\mathbf{C}_s^T(\lambda_s^N + \lambda_s^{N+1}) \\ &\quad +(\mathbf{C}_s^T\mathbf{M}_s^{-1}\mathbf{C}_s)^{-1}(\ddot{\mathbf{u}}_\Gamma^N + \ddot{\mathbf{u}}_\Gamma^{N+1}) + \mathbf{C}_s^T(\mathbf{f}_{\text{int},s}^N + \mathbf{f}_{\text{int},s}^{N+1})\Big) \end{aligned} \tag{59}$$

Accounting for changes only over Δt_L, the right of Figure 6 shows zero energy production is achieved with $\delta W_{\Gamma L}^N + \delta W_{\Gamma s}^N < 1e-17$ using Equation (59) with the forces acting on both interface subdomains. However, upon closer inspection of Equation (59), the energies across each Δt_s must also be computed for the evaluation of stability, to give

$$\begin{aligned} W_\Gamma^{N+1} &= W_{\Gamma L}^N + \delta W_{\Gamma L} + W_{\Gamma s}^n + \delta W_{\Gamma s} = 0 \\ &= W_{\Gamma L}^N + \frac{1}{2}(\mathbf{C}_L^T\mathbf{u}_L^{N+1} - \mathbf{C}_L^T\mathbf{u}_L^N)^T\Big(\mathbf{C}_L^T(\lambda_L^N + \lambda_L^{N+1}) \\ &\quad +(\mathbf{C}_L^T\mathbf{M}_L^{-1}\mathbf{C}_L)^{-1}(\ddot{\mathbf{u}}_\Gamma^N + \ddot{\mathbf{u}}_\Gamma^{N+1}) + \mathbf{C}_L^T(\mathbf{f}_{\text{int},L}^N + \mathbf{f}_{\text{int},L}^{N+1})\Big) \\ &\quad + W_{\Gamma s}^n + \sum_{k=0}^{m}\frac{1}{2}(\mathbf{C}_s^T\mathbf{u}_s^{n+k+1} - \mathbf{C}_s^T\mathbf{u}_s^n)^T\Big(\mathbf{C}_s^T(\lambda_s^{n+k} + \lambda_s^{n+k+1}) \\ &\quad +(\mathbf{C}_s^T\mathbf{M}_s^{-1}\mathbf{C}_s)^{-1}(\ddot{\mathbf{u}}_\Gamma^{n+k} + \ddot{\mathbf{u}}_\Gamma^{n+k+1}) + \mathbf{C}_s^T(\mathbf{f}_{\text{int},s}^{n+k} + \mathbf{f}_{\text{int},s}^{n+k+1})\Big) \end{aligned} \tag{60}$$

The right of Figure 6 depicts the interface energy associated with the constant acceleration, when accounting for variation each Δt_s. This highlights a limitation of the reduced computation of $\ddot{\mathbf{u}}_\Gamma$ to each Δt_L. To elucidate this, the inertial forces on Γ over a large Δt_L are given by:

$$\mathbf{M}_\Gamma^N\ddot{\mathbf{u}}_\Gamma^N = \mathbf{C}_L^T\lambda_L^N + \mathbf{C}_s^T\lambda_s^n + \mathbf{C}_L^T\mathbf{f}_{\text{int},L}^N - \mathbf{C}_s^T\mathbf{f}_{\text{int},s}^n = 0 \tag{61}$$

where $N = n$, whereas over a small Δt_s, the imposed acceleration creates a small contribution:

$$\mathbf{M}_\Gamma^N\ddot{\mathbf{u}}_\Gamma^N = \mathbf{C}_L^T\lambda_L^N + \mathbf{C}_L^T\mathbf{f}_{\text{int},L}^N + \mathbf{C}_s^T\lambda_s^{n+k} - \mathbf{C}_s^T\mathbf{f}_{\text{int},s}^{n+k} + \delta\mathbf{f}_{\Gamma s}^{n+k} = 0 \tag{62}$$

where this extra inertial term is $\delta\mathbf{f}_{\Gamma s}^{n+k}$ for each small step. The associated energy can be seen as the contribution from Equation (62), where $\delta W_{\Gamma s}^{n+k} = 0$, only if the motion of Γ_s follows a constant acceleration. For the case of $m = \pi$, this difference in energy is $< 0.25\%$ of the increment in coupling energies δW_{Link}, and $< 0.001\%$ of the internal and kinetic energy of the system. For this reason, it is deemed the use of $\ddot{\mathbf{u}}_\Gamma$ reasonable over Δt_L. However, this may differ for highly heterogeneous configurations, where in appendix section Square Wave Propagation With A High Time step Ratios, a time step ratio $m = 100$ is demonstrated. In such special cases, the authors suggest one of two simple methods:

- Reducing the Courant number Co_L for integration on Ω_L;
- Integrating the element on Ω_L closest to the material interface with Δt_s.

The latter option follows similar ideas to Dvorak et al. [24], without redefining the computation of $\ddot{\mathbf{u}}_\Gamma$. Advantageously, the algorithm does not require alteration for high values of m. Fundamentally, both these methods increase the number of element integration steps, highlighting the trade-off between computational efficiency and the accuracy of multi-time stepping methods.

3.4 | Continuity of Acceleration, Velocity and Displacement

In the context of central differences, the motion of the coarse scale node is a constant acceleration motion between two-time steps. Through imposing the coupling as a boundary condition on the interface nodes Γ_L and Γ_s, drifting is avoided.

In Figure 7 we use a relative squared error as a measure to check for the convergence of interface kinematics on both subdomains.

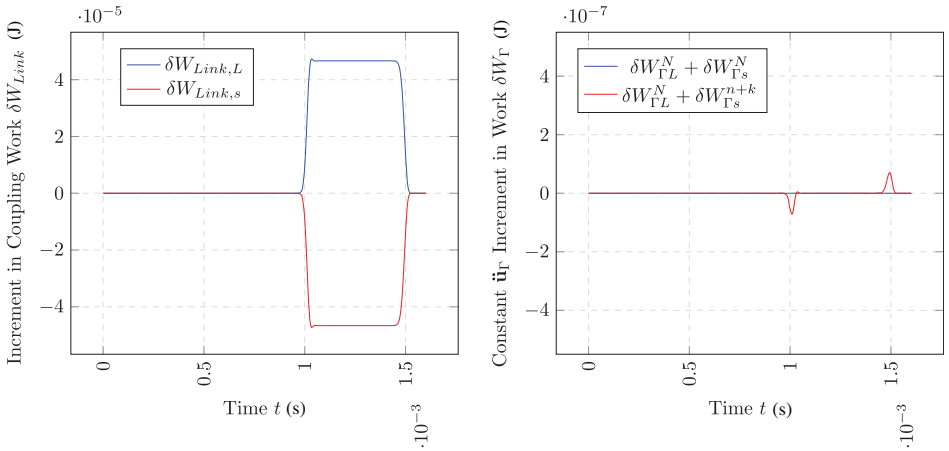

FIGURE 6 | Increments in work with Energy Methods accounting for variation over Δt_L and Δt_s; Left: Increment in work associated to the coupling of the subdomains recovered at each Δt_s; Right: Increment in work done to maintain constant $\ddot{\mathbf{u}}_\Gamma$.

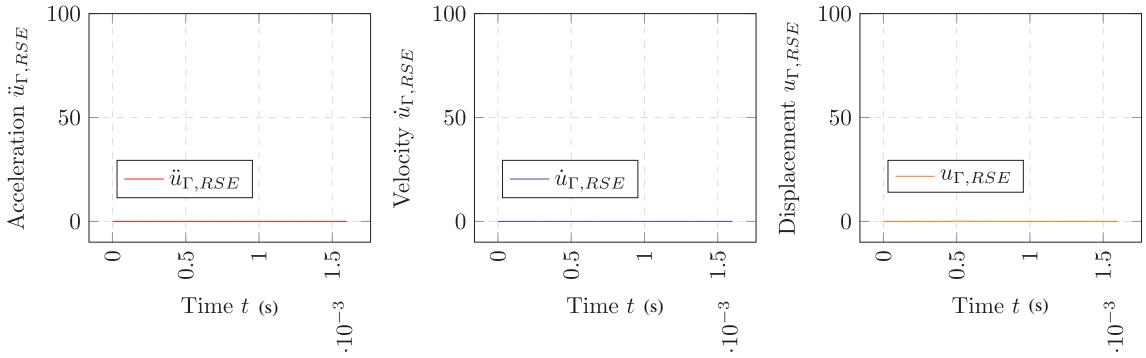

FIGURE 7 | Continuity of kinematic quantities Left: Error of interface accelerations; Centre: Error of interface velocities; Right: Error of interface displacements.

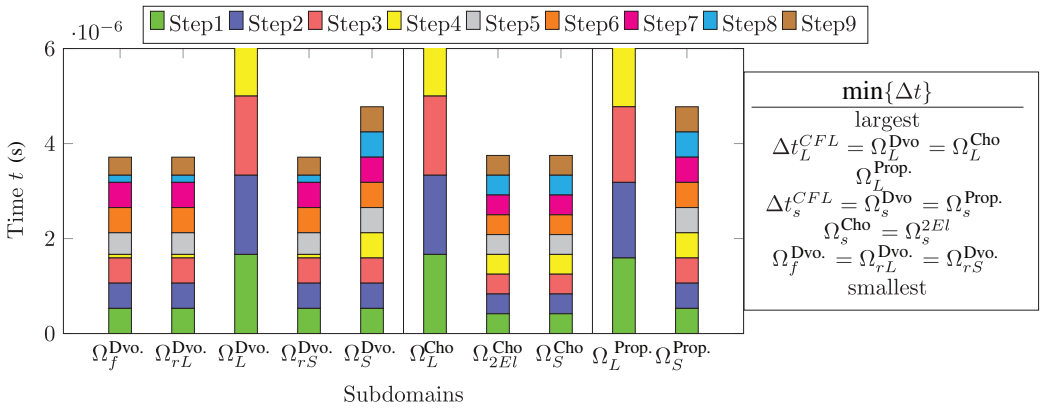

FIGURE 8 | Comparison of subdomain integration steps for recent Multi-Time Stepping algorithms by Dvorak et al. [24], Cho et al. [17] and the proposed method.

For each quantity, we express the following relative squared error

$$\mathbf{A}_{\Gamma,RSE} = \left(\frac{|\mathbf{A}_{\Gamma L} - \mathbf{A}_{\Gamma s}|}{\mathbf{A}_{\Gamma L}} \right)^2 \tag{63}$$

for a kinematic quantity $\mathbf{A}$. We report an absolute difference of $\ddot{u}_\Gamma < 1e-32, \dot{u}_\Gamma < 5e-16, u_\Gamma < 5e-9$ for interface acceleration, velocity and displacement respectively. For round-off errors, one could enforce Equations (30) and (31), however with the consistent use of integration schemes in subdomains, we do not expect the violation of continuity on subdomain interfaces when utilising our algorithm.

4 | Comparison to the State-Of-The-Art

4.1 | Evaluation of Time Steps

We present the local time steps taken by each of the subdomains in the heterogeneous bar benchmark, with a non-integer time step ratio $m = \pi$ in Figure 8. It is desirable to integrate with the largest time step possible to reduce computational effort. In order to compare to Cho et al. and Dvorak et al., we use the total Lagrangian formulation, where Δt_C remains constant for each domain throughout. The solution of Dvorak et al. [24] shows the integration of regions over very small time steps. These small time steps are performed when determining the nearest desired subdomain time level for integrating their regions $\Omega_{rS}^{\text{Dvo.}}$ and $\Omega_{rL}^{\text{Dvo.}}$. The key limitation in the works of Cho et al. is the requirement of integer ratios [17]. For the ratio $m = \pi$, two elements in the large domain $\Omega_{2El}^{\text{Cho}}$ and the domain Ω_s^{Cho} are limited to a $\Delta t_s = \Delta t_{2EL} = \Delta t_L/4$ rather than $\Delta t_L/\pi$, therefore giving a time step further away from the CFL condition.

In the proposed method, we do not require the definition of regions or additional domains to be solved. Figure 8 depicts how we integrate further in time, compared to the two recent methods with $\Omega_s^{\text{Proposed}}$ at the CFL condition of the small elements and $\Omega_L^{\text{Proposed}}$ reducing only to synchronise in time. The limitation of the proposed method is that we do not have both domains integrating at the CFL condition, however without the use of α_L and α_s, the solutions to each domain would not match for non-integer ratios.

On the right of Figure 8, we order the regions and domains from largest to smallest $\min\{\Delta t\}$. Dvorak and Cho step at the critical time step for their large domains, however, there is a shortcoming with the use of the regions and integer ratios respectively. In Table 1, we reiterate how our minimum Δt is closer to the CFL condition in comparison to the other methods. This benchmark emphasises how we can progress further in time with the same number of steps overall.

4.2 | Square Wave Propagation Along Heterogeneous Bar

Next, we plot the velocity distributions across the bar, comparing solutions from each of the methods discussed. In addition to the aforementioned methods, we also implement two methods that alleviate spurious oscillations. The pushforward-pullback method (PFPB) [34] introduces the combination of two methods to smoothen the front and post-shock oscillations. We implement the single-time step method here to compare to our own. The fifth method is combined with our proposed multi-time stepping algorithm to also filter out the oscillations. We favour the proposed method with the traditional implementation of bulk viscosity (BV) by Von Neumann and Richtmyer [35]. Other more recent methods are available such as those described by Maheo, Grolleau, and Rio [36], however, the BV method still remains robust to this day. At each time step, we look to add stress to term to our computation

$$\boldsymbol{\sigma}^{bv} = \rho_i h_e (C_0 \mathrm{Tr}\mathbf{D})^2 - C_1 c_e (\mathrm{Tr}\mathbf{D}) \tag{64}$$

where ρ_i, h_e, c_e and $\mathbf{D}$ maintain the same definitions as before. Common in the case of 1-D elasticity, we omit the quadratic function of the bulk viscosity, and so set constants $C_0 = 0.0$ and $C_1 = 0.06$. The method works particularly well for nonlinear analysis without the requirement to invert or modify the global stiffness matrices. In Figures 9–11, we plot three distinct time steps, where the square wave reaches the interface, partially propagates through, and both transmitted and reflected waves are produced. Each of the plots utilise the same Courant number where $\Delta t = 0.5 h_e / c_e$.

Figure 9 captures the behaviour of the wave at $t = 1$ ms, where the incident wave has been fully applied from Ω_L and reaches the interface Γ_i. Cho, Dvorak, and the proposed method perform similarly, with matching magnitudes of dispersion. The monolithic PFPB and proposed method with bulk viscosity also produce similar results. They are both effective in removing spurious oscillation from the system before the waves arrive at the interface. At $t = 1.25$ ms, in Figure 10, the wave decreases to half of its original value and is partially transmitted into Ω_s. The comparison of methods follows a similar trend to the previous time. The wave is reflected into Ω_L to half of its magnitude, with the other half in Ω_s. The variation in dilatational wave speed is evident with the faster travelling transmitted wave. Observations comparing the methods remain consistent through the wave's interaction with the interface.

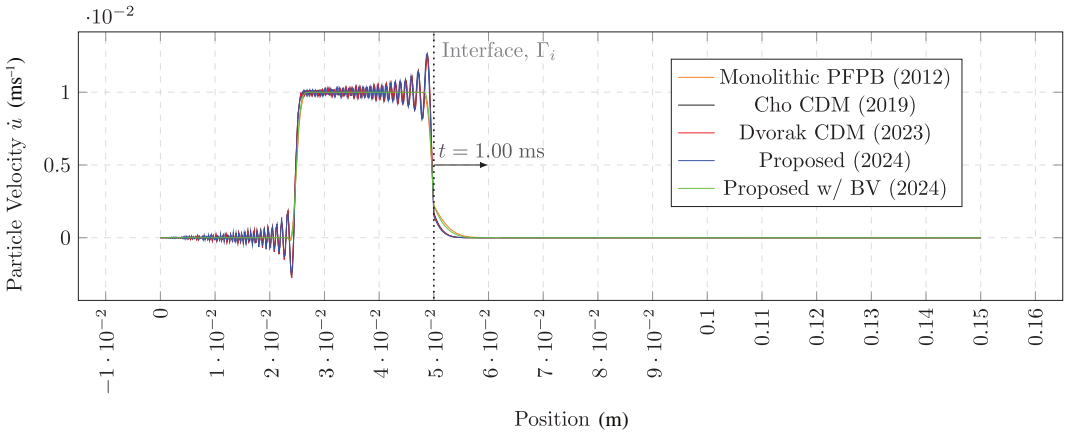

FIGURE 9 | Square wave propagation through a 1-D heterogeneous bar at $t = 1.00$ ms with Monolithic [34], Cho et al. [17], Dvorak et al. [24] and the proposed method with and without bulk viscosity.

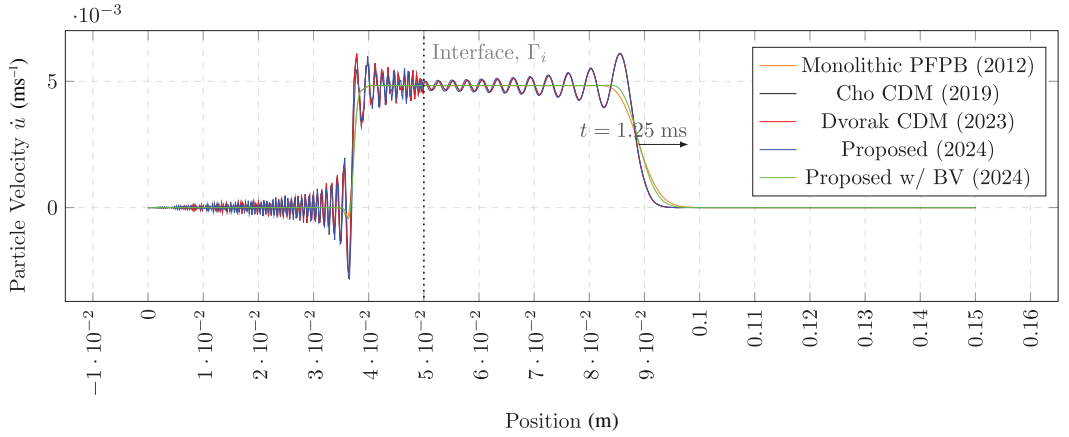

FIGURE 10 | Square wave propagation through a 1-D heterogeneous bar at $t = 1.25$ ms with Monolithic [34], Cho et al. [17], Dvorak et al. [24] and the proposed method with and without bulk viscosity.

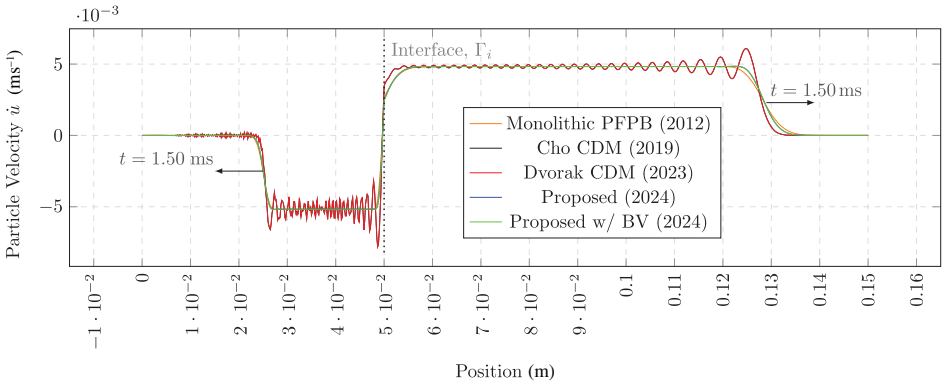

FIGURE 11 | Square wave propagation through a 1-D heterogeneous bar at $t = 1.50$ ms with Monolithic [34], Cho et al. [17], Dvorak et al. [24] and the proposed method with and without bulk viscosity.

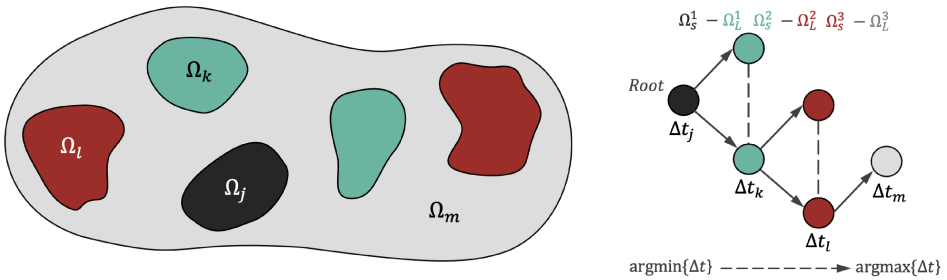

FIGURE 12 | Left: Computational Partitioning of a Domain with four distinct time steps where $\Delta t_i < \Delta t_j < \Delta t_k < \Delta t_l$; Right: Tree representation of the order of solving subdomains, as small-large $\Omega_s - \Omega_L$ pairs. At the point of synchronisation, the integration starts from the root of the tree with minimum Δt_i.

5 | Implementation and Extending to Multi-Domains

5.1 | Multi-Time Step Integration Algorithm for Multiple Domains

In the following section, our coupling procedure is extended to allow for the number of subdomains $\mathcal{N}_S > 2$. We refer to Algorithm 1 as the `two-domain-algorithm`, which takes the small and large subdomains Ω_s, Ω_L as inputs to the function. This therefore allows for the same mathematical operations in Section 2.3 to be utilised. Figure 12 depicts the order in which subdomains are solved, with each colour denoting a subdomain with a different time step. Given the collection of subdomains $\mathcal{S} = \{\Omega_j, \Omega_k, \Omega_l, \Omega_m\}$ with their corresponding time steps $\{\Delta t_i | \Omega_i \in \mathcal{S}\}$, we can define a sequence of subdomains $\Omega_{i1}, \Omega_{i2}, \Omega_{i3}, \Omega_{i4}$ such that

$$\begin{aligned} \Omega_{i1} &:= \underset{\Omega_i \in \mathcal{S}}{\text{argmin}}\{\Delta t_i\}; \quad \Omega_{i2} := \underset{\Omega_i \in \mathcal{S}\setminus\{\Omega_{i1}\}}{\text{argmin}}\{\Delta t_i\} \\ \Omega_{i3} &:= \underset{\Omega_i \in \mathcal{S}\setminus\{\Omega_{i1},\Omega_{i2}\}}{\text{argmin}}\{\Delta t_i\}; \quad \Omega_{i4} := \underset{\Omega_i \in \mathcal{S}\setminus\{\Omega_{i1},\Omega_{i2},\Omega_{i3}\}}{\text{argmin}}\{\Delta t_i\} \end{aligned} \tag{65}$$

where the subdomains are sorted in ascending order of their time steps. $\mathcal{S} \setminus \{\Omega_{i1}, \Omega_{i2}, \dots\}$ removes the already defined subdomains Ω_i from the set of subsequently smaller time step subdomains. We implement a serial merge sort algorithm with a time complexity of $\mathcal{O}(\mathcal{S} \log \mathcal{N}_S)$. This has proven to be robust for current applications, however, for increasing numbers of $\mathcal{N}_S$, other parallel sorting algorithms could be explored [37]. We initially define the small and large domain pairs as

$$\Omega_s := \Omega_{i1}; \quad \Omega_L := \Omega_{i2} \tag{66}$$

with the corresponding time step $\Delta t_s := \Delta t_{i1}$ and $\Delta t_L := \Delta t_{i2}$.

Following the solution of a single large time step Δt_L, with Algorithm 3, the sorting algorithm is called with newly computed Δt_s and Δt_L. We also update the trial time $t_{Ti} = t_i + \Delta t_i$ for each Ω_i. In the presence of more subdomains, we evaluate whether the pair of subdomains Ω_s, Ω_L can advance further in time or whether the next pair should be integrated. To do so, the trial times are utilised to evaluate;

$$t_{Ts} < t_{TL+1} \tag{67}$$

where t_{Ts} is the trial small subdomain's time and t_{TL+1} is the trial time for the subdomain next in the sorted collection, the next $\Delta t_i > \Delta t_L$. If inequality (67) is valid the current pair of subdomains are integrated once more. However if not fulfilled, the large subdomain is assigned as the small domain and the next largest Ω_i is assigned to be the large subdomain. Following the integration of all subdomain pairs, the system reaches t^N where $\forall t_i = t^N$. Figure 12 highlights the ability to perform the concurrent advancement of subdomains with similar time steps. For example those subdomains with Δt_k as a time step are permitted to advance together. This is advantageous for systems that consist of multiple subdomains with similar material properties.

ALGORITHM 2 | Multi-Time Step Integration Algorithm for Multplie Domains.

Sort subdomains Ω_i from argmin$\{\Delta t_i\}$ to argmax$\{\Delta t_i\}$;
while All subdomains Ω_i have a $t_i < t_{max}$ and number of subdomains $\mathcal{N}_S > 1$ **do**
  Integrate with the `two-domain-algorithm` (Ω_s, Ω_L) ▷ Using Algorithm 1
  Sort subdomains Ω_i from argmin$\{\Delta t_i\}$ to argmax$\{\Delta t_i\}$;
  if Larger time step subdomains Ω_{L+1} exist **then**
    if $t_{Ts} < t_{TL+1}$ **then**
      Integrate current small-Large (Ω_s, Ω_L) pair;
    else
      Define the next largest $\Delta t_s, \Delta t_L$ pair with subdomains $\Omega_s := \Omega_L$ then $\Omega_L := \Omega_{L+1}$;
    end if
  else
    Exit the Loop and Break; ▷ All subdomains are synchronised in time $\forall t_i = t^N$
  end if
end while

The description of the multi-time domain procedure is summarised in Algorithm 2. We observe that the re-sorting of subdomains after Δt_L^N, allows for a time step ratio m that may change as the simulation advances. As previously mentioned, large deformation is just one example of how a subdomain's Δt may suddenly increase or decrease. The algorithm allows for the switching of Ω_s and Ω_L, following a single large time step. It should be noted, that the aforementioned Algorithm 3 does require slight modification to allow for subdomains that are adjacent spatially but not temporally. As an example of this, we can show that Ω_j and Ω_m are spatially adjacent, however are not integrated as an $\Omega_s - \Omega_L$ pair. As a result, a slight modification is made:

$$\mathbf{M}_\Gamma = (\mathbf{C}_s^T \mathbf{M}_s \mathbf{C}_s) + (\mathbf{C}_L^T \mathbf{M}_{\overline{L}} \mathbf{C}_L) \tag{68}$$

$$\mathbf{f}_\Gamma^{\text{int}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{int}} + \mathbf{C}_{\overline{L}}^T \mathbf{f}_{\overline{L}}^{\text{int}}; \quad \mathbf{f}_\Gamma^{\text{ext}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{ext}} + \mathbf{C}_{\overline{L}}^T \mathbf{f}_{\overline{L}}^{\text{ext}} \tag{69}$$

$$\ddot{\mathbf{u}}_\Gamma = \ddot{\mathbf{u}}_{\Gamma s} = \ddot{\mathbf{u}}_{\Gamma \overline{L}} \tag{70}$$

where we denote subdomain $\overline{L}$ as the spatially adjacent large domain. In the following section, we look to demonstrate the capability of the algorithm with a domain partitioned into three subdomains, each running on their respective time step.

ALGORITHM 3 | Two-Domain Constant Acceleration Multi-Time Step Integration Description.

while t_s and $t_L < t_{max}$ **do**
  $k = 0$;
  if t_s and $t_L == 0$ **then**
    compute $\mathbf{f}_s^{\text{int}}, \mathbf{f}_s^{\text{ext}}$ and $\mathbf{M}_s$; ▷ Small Force and Mass Vectors
    compute $\mathbf{f}_L^{\text{int}}, \mathbf{f}_L^{\text{ext}}$ and $\mathbf{M}_L$; ▷ Large Force and Mass Vectors
    for all nodes $\mathcal{N}_i^\Gamma$ **do**
      $\mathbf{M}_\Gamma = (\mathbf{C}_s^T \mathbf{M}_s \mathbf{C}_s) + (\mathbf{C}_L^T \mathbf{M}_L \mathbf{C}_L)$; ▷ Mass Exchange
      $\mathbf{f}_\Gamma^{\text{int}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{int}} + \mathbf{C}_L^T \mathbf{f}_L^{\text{int}}$; ▷ Internal Force Summation
      $\ddot{\mathbf{u}}_\Gamma = \mathbf{M}_\Gamma^{-1}(\mathbf{f}_\Gamma^{\text{ext}} - \mathbf{f}_\Gamma^{\text{int}})$ ▷ Large Acceleration
    end for
    compute $t_{Ts}^{n+k}, t_{TL}^{N+1}, t_{\text{ratio}}^{n+k}$ and t_{ratio}^{n+k+1}; ▷ Time Step Ratios
  end if
  while $t_{\text{ratio}}^{n+k+1} \leq 1$ or $(t_{\text{ratio}}^{n+k} \leq 1$ and $t_{\text{ratio}}^{n+k+1} \leq 1)$ **do** ▷ Small Subdomain Integration
    let $\ddot{\mathbf{u}}_{\Gamma s}^n = \ddot{\mathbf{u}}_\Gamma^n$;
    solve subdomain Ω_s Equation~(17) for Δt_s
    $t_s = t_s + \Delta t_s$, k += 1;
    compute $\mathbf{f}_s^{\text{int}}, \mathbf{f}_s^{\text{ext}}$;
    compute $t_{Ts}, t_{\text{ratio}}^{n+k}$ and t_{ratio}^{n+k+1};
  end while
  compute α_L and α_s; ▷ Large and Small Reduction Factor
  if $\alpha_L \geq \alpha_s$ **then**
    $\Delta t_L = \alpha_L \cdot \Delta t_L$;
  else ▷ Small Subdomain Integration
    $\Delta t_s = \alpha_s \cdot \Delta t_s$;
    let $\ddot{\mathbf{u}}_{\Gamma s}^n = \ddot{\mathbf{u}}_\Gamma^n$;
    solve subdomain Ω_s Equation~(17) for Δt_s
    $t_s = t_s + \Delta t_s$, k += 1;
    compute $\mathbf{f}_s^{\text{int}}, \mathbf{f}_s^{\text{ext}}$;
    compute $t_{Ts}, t_{\text{ratio}}^{n+k}$ and t_{ratio}^{n+k+1};
  end if ▷ Large Subdomain Integration
  let $\ddot{\mathbf{u}}_{\Gamma L}^n = \ddot{\mathbf{u}}_\Gamma^n$;
  solve subdomain Ω_L Equation~(17) for Δt_L
  $t_L = t_L + \Delta t_L$;
  for all nodes $\mathcal{N}_i^\Gamma$ **do**
    compute $\mathbf{f}_L^{\text{int}}, \mathbf{f}_L^{\text{ext}}$;
    $\mathbf{f}_\Gamma^{\text{int}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{int}} + \mathbf{C}_L^T \mathbf{f}_L^{\text{int}}$;
    $\ddot{\mathbf{u}}_\Gamma = \mathbf{M}_\Gamma^{-1}(\mathbf{f}_\Gamma^{\text{ext}} - \mathbf{f}_\Gamma^{\text{int}})$
  end for
  compute $t_{TL}, t_{\text{ratio}}^{n+k}$ and t_{ratio}^{n+k+1};
end while

5.2 | A Note on Implementation

The algorithms have been implemented in an in-house code, namely DEST [38], developed by the University of Oxford's Impact and Shock Mechanics Laboratory. We divide the domain into subdomains, which are executed concurrently on separate threads. The main overhead associated with this algorithm is the communication of kinematic and kinetic quantities for those nodes found on boundary Γ. We will later argue, in the 3-D numerical examples, that this overhead, still allows for a large speedup in the runtime of larger numerical simulations. Communication is controlled from a main thread with parallel barriers and signals, which instructs and sorts the subdomains to advance in time.

While the multi-time stepping algorithm has been demonstrated using Lagrangian linear hexahedral finite elements, the same methodology can be extended to other element types, such as Lagrangian linear tetrahedral elements, and higher order elements. The method is not limited by the choice of the number of nodes utilised to couple subdomains, for example, simply a definition that utilises three or four coupling nodes for a

conventional tetrahedral discretisation of a square cross section bar requires no change in the implementation.

An open-source Python code is also available in the following repository https://github.com/kinfungchan/multi-time-step-integration to depict the two-domain multi-time step integration algorithm. The code illustrates an implementation of how to couple two monolithic subdomains, with the exchange of system variables and the evaluation of time step reduction for both. The same non-integer example in Section 4 can be run from the repository to simulate wave propagation in the 1-D heterogeneous domain. For reference, an implementation of the other methods by Belytschko and Mullen [2], Cho et al. [17] and Dvorak et al. [24] are also found in the repository.

6 | 3-D Numerical Experiments

To demonstrate the effectiveness of the proposed multi-time stepping integration algorithm, we present the results of stress wave propagation in heterogeneous structures subject to impact loading. We demonstrate the multiple domain algorithm by solving the kinematics and kinetics in a novel 3-D metamaterial called metaconcrete. To evaluate the stability of the algorithm, the energy balance for each subdomain is used throughout the examples. The single-time step (monolithic) solution is used for comparison to assess the accuracy and speedup of the multi-time stepping solution.

6.1 | Wave Propagation in Metaconcrete

Recently, metamaterials have gained increasing interest due to their ability to be engineered with properties suitable for a variety of applications, including acoustics, electromagnetics, and stress waves [39]. The concept of metaconcrete, as introduced by Mitchell, Pandolfi, and Ortiz [40], utilises resonance induced negative effective masses. The aggregates in metaconcrete can be tuned so that the resonant oscillations of the internal mass are induced as the stress wave propagates through the structure. This therefore allows engineers to predict the fraction of mechanical energy that can be absorbed by the material. Metaconcrete is a three-phase composite, where a unit cell comprises of a heavy inclusion and compliant coating in a matrix. Experimentally, recent works look to characterise the dynamic behaviour of different configurations and materials of these three constituents [41–43]. Here, the focus is to show the efficiency of our algorithm when compared to monolithic solutions, for a heterogeneous structure. A unit cell is defined, similar to that of [44], with an inclusion radius r_I and coating radius r_c of 9 mm and 12 mm respectively. The spherical geometries are housed within a cube of $d = 30$ mm edge length. Figure 13 depicts the partitioning of the domain. This results in the volume fractions stated in Table 2. Each phase is assigned its distinct subdomain, Ω_i. Firstly, a constant time step ratio m between Ω_i is demonstrated, followed by a non-constant m through invoking plasticity in the structure. In this numerical study, the dynamic excitation follows a half sinusoidal wave, with varying amplitude and frequency. All simulations utilise Algorithm 2 we described in Section 5.1.

6.1.1 | Elastic, Structured Domain With Constant Time Step Subcycling

A structured hexahedral FE mesh is created, of uniform mesh size $h = 0.65$ mm, allowing for reduction around volumes of interest, as shown in Figure 14. The chosen isotropic elastic material properties for the three phases are $E_m = 3$ GPa, $\rho_m = 1.10e-6$ kg/mm^3, $\nu_m = 0.37$, $E_c = 10$ GPa, $\rho_c = 1.50e-6$ kg/mm^3, $\nu_c = 0.4$, $E_I = 210$ GPa, $\rho_I = 7.57e-6$ kg/mm^3 and $\nu_I = 0.3$. Elastic material constants can widely be found in the literature by the likes of [40, 43, 45–47]. In 3-D, to compute the dilatational wave speed in each subdomain, the following equation is used:

$$c_{3-D} = \sqrt{\frac{E_i(1-\nu_i)}{(1+\nu_i)(1-2\nu_i)\rho_i}} \tag{71}$$

TABLE 2 | Discretisation of structured Metaconcrete bar mesh data and dilatational wave speed.

Material	Nodes	Elements	Volume fraction $V_i(\%)$	Wave speed c_{3-D} (mm/ms)
Matrix	447,065	403,840	73.19	2.20e + 03
Coating	194,784	162,240	15.50	3.78e + 03
Inclusion	189,928	173,056	11.31	6.11e + 03

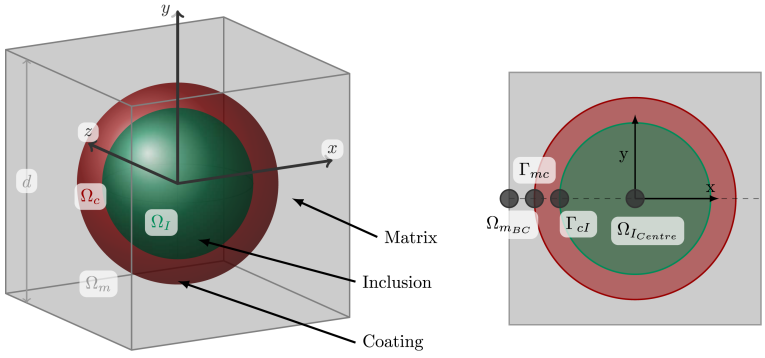

FIGURE 13 | Left: Metaconcrete unit cell, partitioned into three individual subdomains including the matrix Ω_m, coating Ω_c and inclusion Ω_i; Right: Sampling points for elastic wave propagation metaconcrete simulation plot taken along the centre axis with interfaces between matrix-coating and coating-inclusion denoted Γ_{mc} and Γ_{cI} respectively.

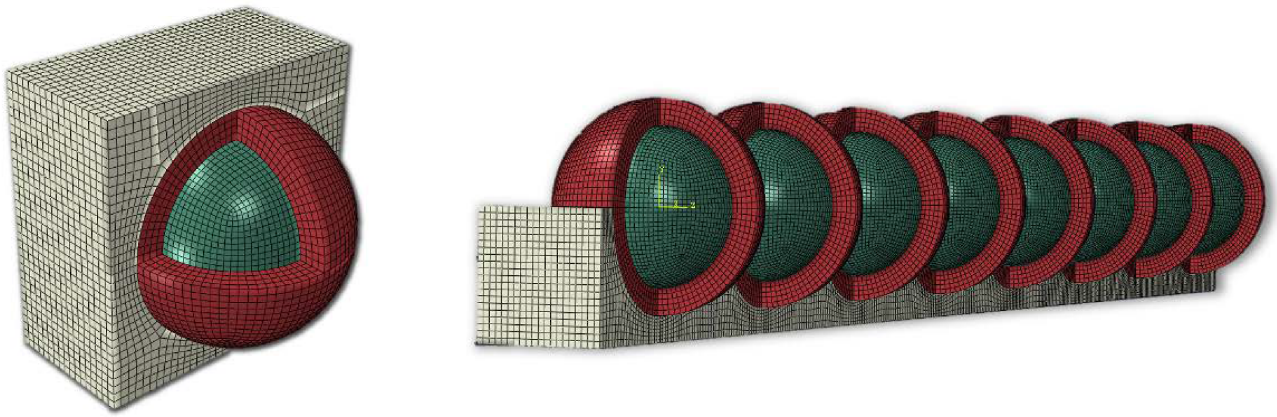

FIGURE 14 | Left: Mesh of a Metaconcrete unit cell used for the elastic simulation, Right: Section cut of long bar mesh.

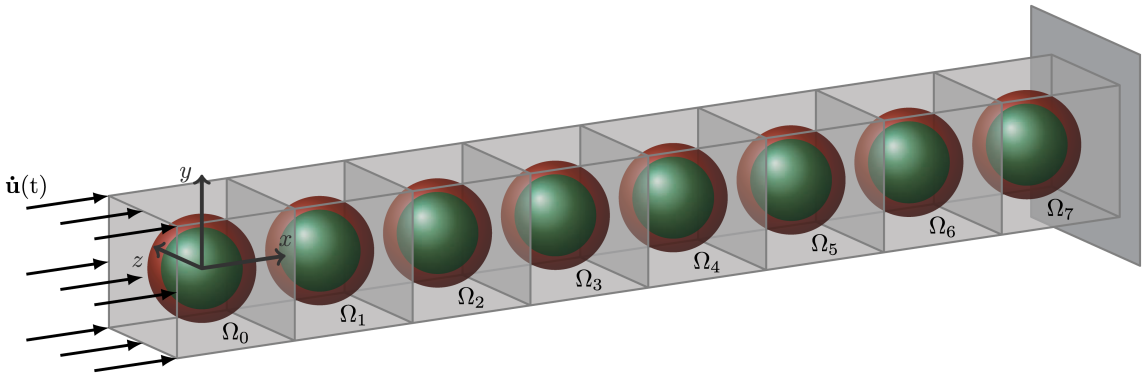

FIGURE 15 | Diagram of Metaconcrete bar consisting of 8 unit cells labelled Ω_0 to Ω_7 with prescribed velocity at $x = -15$ mm and support at $x = 225$ mm.

The discretisation data and dilatational wave speed are tabulated in Table 2. A bar of unit cells is modelled, similar to Xu et al. [43] with the assessment of experimental and numerical stress wave attenuation in various metaconcrete bars. A total of eight coated inclusions are modelled in a matrix with a repeated configuration throughout the bar, each unit cell with the same mesh refinement. All nodes at $x = -15$ mm on the matrix are prescribed a half sine velocity boundary condition with $\dot{u}(t) = 0.01\sin(2\pi\omega_m t)$ mm/ms, where $\omega_m = 18.35$ rads^{-1}. The equivalent wavelength is 90 mm, with the slowest dilatational wave speed being the matrix. The bar amplitude of the stress pulse is small in magnitude, so to maintain the deformation well within the elastic region of each material.

A pre-processing step is required with the definition of those nodes on Γ between phases. In the case of metaconcrete, we define those nodes on Γ_m that couple with Γ_c, as well as those situated on Γ_c with Γ_I. Determination of which subdomain is Ω_L and Ω_s is not required a priori, as it is determined within Algorithm 2. As previously mentioned, each subdomain Ω_i can be partitioned in such a way that all those belonging to similar time steps Δt_m, Δt_I and Δt_c can be solved concurrently. As depicted in Figure 15, the individual components of matrix, coating, and inclusion material can be integrated respectively. This is particularly advantageous for systems that have periodic or repeating structures, highlighting a key use case for multi-time stepping with metamaterials.

The results in Figure 16 depict the axial velocity $\dot{\mathbf{u}}_x$ through the centre of the metaconcrete bar in the first unit cell Ω_0. The responses are taken along the central axis, at nodes on the boundary condition, Γ_{Mc}, Γ_{cI} and at the centre of the inclusion, as shown on the right of Figure 13. As shown previously the coupling between the subdomains leads to no difference in nodal velocities, within machine tolerance. Consequently, just the values from Ω_c are taken for those nodes that are situated on Γ_{mc} and Γ_{cI} for the multi-time step solution. The sampling rates are kept consistent for each nth point. Using the node at the boundary condition, it can be seen that for every singular multi-time stepping (abbreviated as *mts*) value, four monolithic (*mono*) values are solved. Remarkably, Figure 16 reveals the *mts* velocity for Ω_0 performs very well when compared to the monolithic velocity, even for those nodes on the boundary between subdomains.

We also make an assessment of the stress in the metaconcrete bar. The results in Figures 17–19 depict good alignment when visualising the Cauchy stress field. A slice from the central $X - Y$ plane in the domain is plotted with the values of axial stress for each element. Three-time steps are presented to depict the stress wave travelling through the bar. In each of the figures, the monolithic solution is plotted above the multi-time step solution at each point in time. The difference in impedance gives reason for the wave travelling faster through the centre where the inclusion constitutes to a larger proportion of the unit cell volume. This

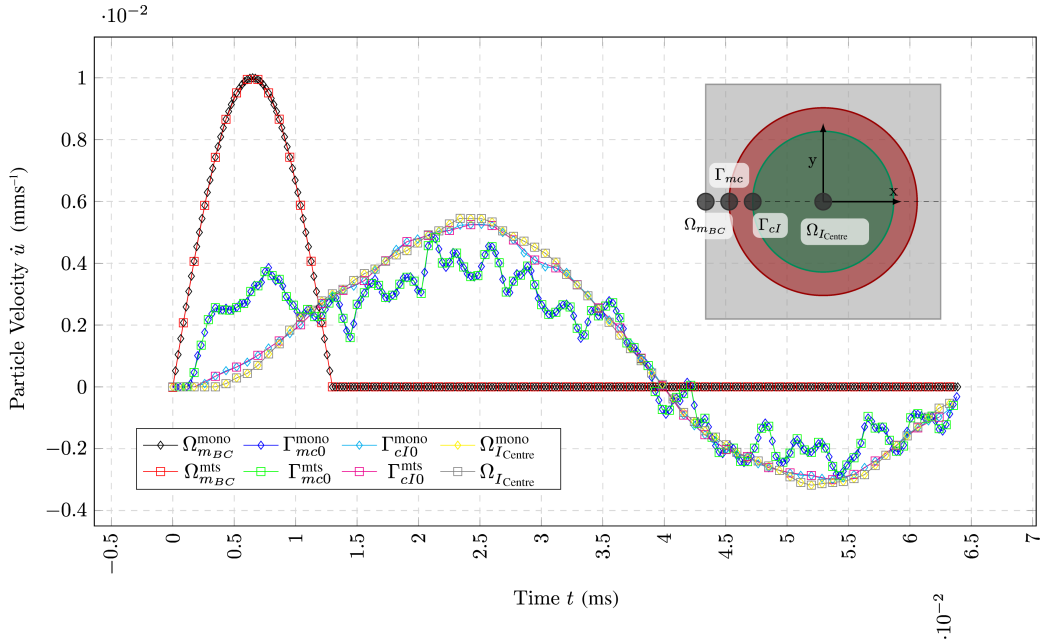

FIGURE 16 | Elastic wave propagation through a Metaconcrete bar with monolithic mono and multi-time step mts solutions in the first Unit Cell at the coordinates $\{-15.0, 0.0, 0.0\}$, $\{-12.0, 0.0, 0.0\}$, $\{-9.0, 0.0, 0.0\}$ and $\{0.0, 0.0, 0.0\}$ where *BC*, *mc*, *cI* and I_{Centre} represent the boundary condition, matrix-coating, coating-inclusion and the centre of the inclusion respectively.

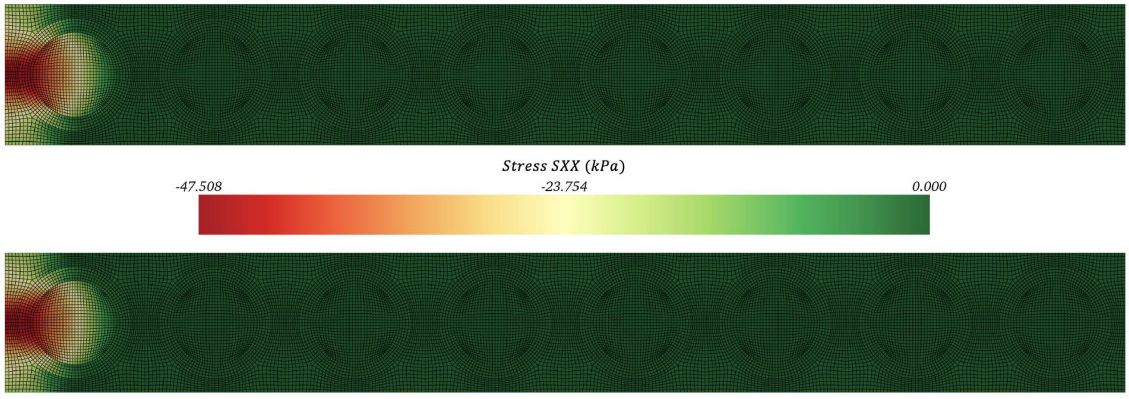

FIGURE 17 | Top: Monolithic solution $\Omega^{\text{mono}}_{\sigma_{xx}}$, Bottom: Multi-time step solution $\Omega^{\text{mts}}_{\sigma}$; Elastic wave propagation at $t = 0.0065$ ms.

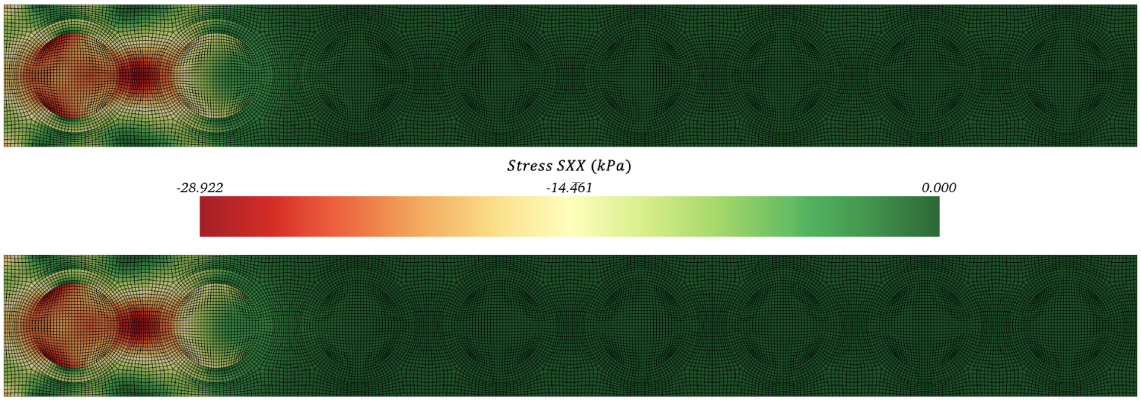

FIGURE 18 | Top: Monolithic solution $\Omega^{\text{mono}}_{\sigma_{xx}}$, Bottom: Multi-time step solution $\Omega^{\text{mts}}_{\sigma_{xx}}$; Elastic wave propagation $t = 0.0260$ ms.

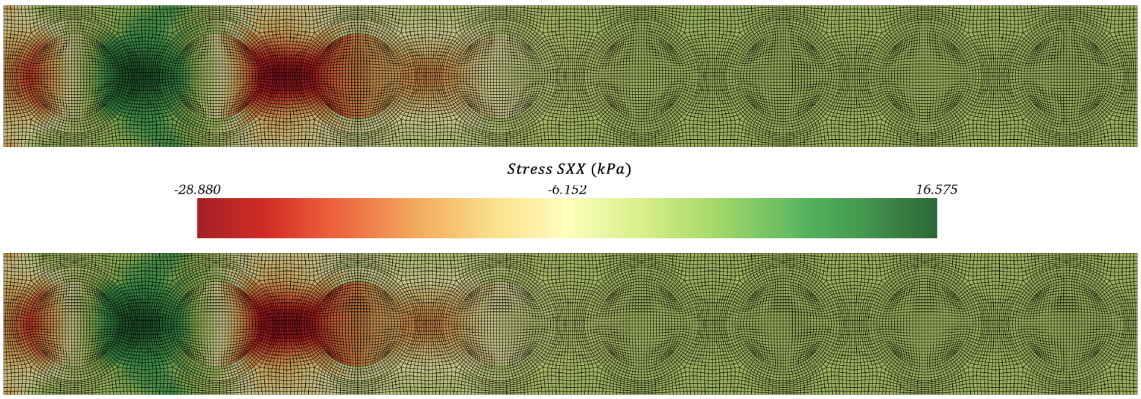

FIGURE 19 | Top: Monolithic solution $\Omega^{\text{mono}}_{\sigma_{xx}}$, Bottom: Multi-time step solution $\Omega^{\text{mts}}_{\sigma_{xx}}$; Elastic wave propagation $t = 0.0520$ ms.

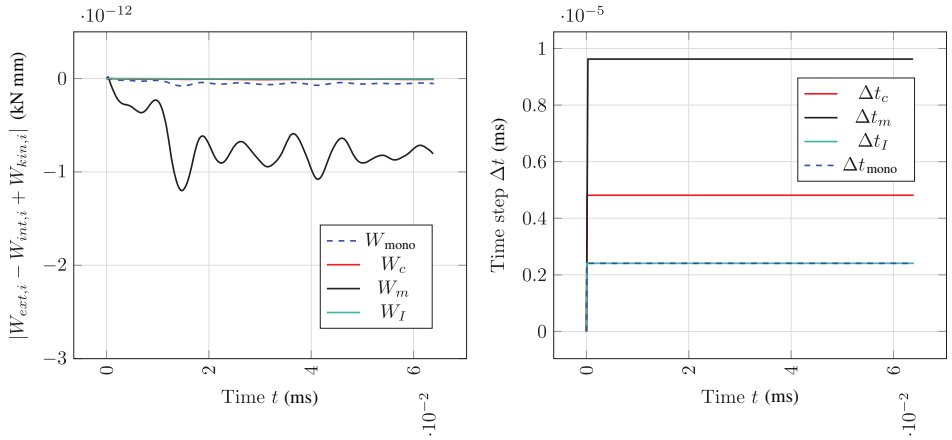

FIGURE 20 | Left: Energy balance check $|W_{\text{ext},i} - W_{\text{int},i} + W_{\text{kin},i}|$ of elastic wave propagation through a Metaconcrete bar with monolithic and multi-time step solution, Right: Time step history for each subdomain Ω_i.

leads to slower propagation near the edges of the bar where the value of the matrix dilatational wave is smaller.

To evaluate the overall stability, we check the energy balance of each subdomain Ω_i. Neal and Belytschko presented an energy balance method for multi-time stepping of subdomains, which is foundational to our approach [6]. Belytschko [48] The total energy W_i of a subdomain i satisfies the following inequality

$$|W_{\text{ext},i} - W_{\text{int},i} + W_{\text{kin},i}| \leq \delta ||W_i|| \tag{72}$$

where $W_{\text{ext},i}$ denotes the external work, $W_{\text{int},i}$ and $W_{\text{kin},i}$ denote the internal and kinetic energies of a subdomain, respectively, and δ is a tolerance value. The internal, external, and kinetic energy of Ω_i can be calculated with the velocities on the half-time step, aligning with our use of leapfrog time integration. Each subdomain evaluates the individual component as the following [49]:

$$\begin{aligned} W^{n+1}_{\text{int},i} &= W^{n}_{\text{int},i} + \frac{\Delta t_i^{n+1/2}}{2} (\dot{\mathbf{u}}_i^{n+1/2})^T (\mathbf{f}^n_{\text{int},i} + \mathbf{f}^{n+1}_{\text{int},i}) \\ &= W^{n}_{\text{int},i} + \frac{1}{2} \Delta \mathbf{u}_i^T (\mathbf{f}^n_{\text{int},i} + \mathbf{f}^{n+1}_{\text{int},i}) \end{aligned} \tag{73}$$

$$\begin{aligned} W^{n+1}_{\text{ext},i} &= W^{n}_{\text{ext},i} + \frac{\Delta t_i^{n+1/2}}{2} (\dot{\mathbf{u}}_i^{n+1/2})^T (\mathbf{f}^n_{\text{ext},i} + \mathbf{f}^{n+1}_{\text{ext},i}) \\ &= W^{n}_{\text{ext},i} + \frac{1}{2} \Delta \mathbf{u}_i^T (\mathbf{f}^n_{\text{ext},i} + \mathbf{f}^{n+1}_{\text{ext},i}) \end{aligned} \tag{74}$$

$$W^{n}_{\text{kin},i} = \frac{1}{2} (\dot{\mathbf{u}}_i^{n+1/2})^T \mathbf{M} \dot{\mathbf{u}}_i^{n+1/2} \tag{75}$$

A similar expression can also be developed for the evaluation of internal energy, using the integration points (denoted by Q below) in the elements.

$$\begin{aligned} W^{n+1}_{\text{int},i} = W^{n}_{\text{int},i} + \frac{1}{2} \sum_{e=1}^{N_e} \Delta \mathbf{u}_e^T (\mathbf{f}^n_{\text{int},e} + \mathbf{f}^{n+1}_{\text{int},e}) = W^{n}_{\text{int},i} \\ + \frac{\Delta t_i^{n+1/2}}{2} \sum_{e=1}^{N_e} \sum_{q=1}^{N_Q} w_Q \mathbf{D}_Q^{n+1/2} : (\boldsymbol{\sigma}^n_Q + \boldsymbol{\sigma}^{n+1}_Q) J_{\xi Q} \end{aligned} \tag{76}$$

These summate the contributions from each element e and its quadrature point Q, along with weights w, rate of deformation $\mathbf{D}$, Cauchy stress $\boldsymbol{\sigma}$ and the Jacobian J. In the following sections, we will evaluate each type of energy by plotting the balance with Equation (72). We equate on all subdomains i, looking to compare the values from those solved monolithically and with multi-time stepping. In Figure 20 we numerically evaluate the equations

TABLE 3 | Monolithic and Multi-Time Step runtimes and minimum time steps of the matrix Ω_m, coating Ω_c and inclusion Ω_I.

Elastic simulation	Runtime (s)	min{Δt} (ms)
Monolithic (single-time step)	14,400	2.4064e-6
Multi-Time Stepping (mts)	8,400	9.6256e-6 (Ω_m) | 4.8128e-6 (Ω_c) | 2.4064e-6 (Ω_I)

TABLE 4 | Discretisation of voxelised metaconcrete bar data and dilatational wave speed.

Material	Nodes	Elements	Volume fraction V_i(%)	Wave speed c_{3-D} (mm/ms)
Matrix	1,219,057	1,139,872	73.19	2.02e + 03
Coating	290,424	241,808	15.53	4.20e + 03
Inclusion	200,520	175,696	11.28	1.12e + 04

above. Through utilising the multi-time stepping algorithm, one observes slight dissipation by analysing the energy balance. This limitation is evident, especially with the comparison of Ω_m and the monolithic solution. However, the energy balance is close to the machine's precision, and several orders of magnitude below the total energy.

The ability to evaluate each Ω_i according to its time step is clearly advantageous, as evidenced by the time savings shown in Table 3. Multi-time stepping yields a 1.71× speedup with a 41.67% reduction in runtime. The reason for this is clear when we evaluate the time steps taken throughout the simulation, where the monolithic time step Δt_{mono} is limited by the inclusion Δt_I. This speedup in the multi-time step solution would be even larger for metaconcrete configurations, where the composite comprises a larger volume fraction of the matrix and coating material.

6.1.2 | Plastic, Voxelised Domain With Variable Time Step Subcycling

The previous section looks at near constant time step ratios m, however with a voxelised (raster) mesh and an isotropic hypoelastic-plastic model we can increase complexity, and demonstrate non-constant ratios. Plasticity enables the modelling of large deformations, facilitating further comparisons between monolithic and multi-time step solutions. However, to accurately capture the true behavior of metaconcrete, fracture and damage models could be incorporated [50]. Voxel meshes have been widely used across a variety of engineering applications [51–54], a key reason being their rapid generation. Voxel meshes allow the time step to be primarily driven by material properties and element distortion, rather than by the initial discretisation (Table 4).

The geometry is configured identically to the elastic metaconcrete bar in the previous section, and meshed through defining a minimum element size and then appropriately assigning materials based on the element's distance (d_i) from the centre of the composite unit cell (a, b, c). The element distance from the unit cell centre is calculated with $d_i = ((x-a)^2 + (y-b)^2 + (z-c)^2)^{1/2}$, following which we assign an element e_i to each subdomain i through considering the radius of the inclusion r_I and coating r_c:

$$e_i \in \Omega_I, d_i \leq r_I; \quad e_i \in \Omega_c, r_I < d_i < r_c; \quad e_i \in \Omega_m, d_i \geq r_c \quad (77)$$

To model elasto-plasticity, a finer discretisation is required. In doing so, voxels are refined to have a minimum element length of $h = 0.326$ mm, as well as utilising a quarter symmetry of the bar to reduce the total element count. The aforementioned velocity boundary conditions are exchanged for an external force acting on those nodes at $x = -15$ mm with $F(t) = 0.0025 \sin(2\pi\omega_m t)$kN, where $\omega_m = 17.20$ rads^{-1}. Symmetrical boundary conditions are applied to the domain, so that those nodes belonging to the edges see $0.5F(t)$ and those corner nodes $0.25F(t)$. Displacements perpendicular to the symmetry planes are set to zero. A wavelength of three unit cells is maintained for the simulation. The material properties are adjusted accordingly to provide a larger initial time step ratio m between subdomains with elastic material properties $E_m = 3.5$ GPa, $\rho_m = 1.46e-6$ kg/mm^3, $\nu_m = 0.363$, $E_c = 4$ GPa, $\rho_c = 1.05e-6$ kg/mm^3, $\nu_c = 0.46$, $E_I = 240$ GPa, $\rho_I = 2.74e-6$ kg/mm^3 and $\nu_I = 0.32$. Beyond the yield of the material, plasticity occurs, and the material behaviour is modelled with a nonlinear isotropic hardening constitutive model. The backward Euler radial return algorithm is used to iterate the state in stress space back to the von Mises yield criterion. Return mapping schemes are widely used for plasticity, enforcing consistency at the end of the time step. We describe the integration method for k iterations in four key steps. Firstly, the initialisation for $k = 0$ is required giving:

$$\begin{aligned} k = 0 &: \varepsilon^{p^{(0)}} = \varepsilon_n^p; \quad \overline{\varepsilon}^{(0)} = \overline{\varepsilon}_n; \quad \Delta p^{(0)} = 0 \\ \boldsymbol{\sigma}^{(0)} = \mathbf{C}^e &: (\varepsilon_{n+1} - \varepsilon^{p(0)}) \end{aligned} \quad (78)$$

where initial values of plastic strain, $\varepsilon^{p^{(0)}}$, and effective strain $\overline{\varepsilon}^{(0)}$ are assigned with elasticity tensor $\mathbf{C}^e$. The increment of the plasticity parameter $\Delta p^{(0)}$ starts from 0, as well as the elastic trial stress $\boldsymbol{\sigma}^{(0)}$ computed for the time step. Then, we let f be the yield function such that if $f^{(k)} < TOL$ convergence has been reached

$$f = \overline{\sigma}^{(k)} - \sigma_y(\overline{\varepsilon}^{(k)}) = (\overline{\sigma}^{(0)} - 3G\Delta p^{(k)}) - \sigma_y(\overline{\varepsilon}^{(k)}) \quad (79)$$

where G is used for shear modulus, $\overline{\sigma}^{(k)}$ denotes the effective stress. Derivations of such expressions can be found in computational plasticity literature such as [28, 29]. The increment in the plasticity parameter can then be found by substituting our yield function into $\delta p = \frac{f^{(k)}}{3G + H^{(k)}}$ to give:

$$\delta p = \frac{(\overline{\sigma}^{(0)} - 3G\Delta p^{(k)} - \sigma_y(\overline{\varepsilon}^{(k)})}{3G + H^{(k)}} \quad (80)$$

We utilise the backward Euler radial return method. To complete a single iteration of the return method, the plastic strain and internal variables are then updated:

$$\varepsilon^{p(k+1)} = \varepsilon^{p(k)} + \Delta\varepsilon^{p(k)} \tag{81}$$

$$\overline{\varepsilon}^{(k+1)} = \overline{\varepsilon}^{(k)} + \delta p^{(k)} \tag{82}$$

$$\Delta p^{(k+1)} = \Delta p^{(k)} + \delta p^{(k)} \tag{83}$$

$$\boldsymbol{\sigma}^{(k+1)} = \mathbf{C}^e : (\varepsilon_{n+1} - \varepsilon^{p(k+1)}) = \boldsymbol{\sigma}^{(k)} + \Delta\boldsymbol{\sigma}^{(k)} \tag{84}$$

We return to the check of the yield criterion $f^{(k)}$ for when $k \leftarrow k+1$ and repeat the process to solve for the increment in plasticity parameter. With this integration method in hand, we now demonstrate multi-time stepping for plasticity problems.

As with the elastic results, we present the propagation of the velocity profile through the entire quarter metaconcrete bar. The propagation of the velocity profile through the quarter metaconcrete bar is presented. Two coupling nodes on Γ_{mc} (black sampling point) and Γ_{cI} (blue sampling point) for each unit cell are plotted with their corresponding solution on monolithic Ω_{mono} in Figures 21 and 22. Their sampling points are taken along the x axis where $y = z = 0$, as shown in Figure 23. Subsequently, we observe the gradual reduction in amplitude of the wave as it propagates in both simulations. Both matrix-coating and coating-inclusion combinations, show an example of the wave transitioning into a material with a comparatively higher impedance. The velocity profiles align well for these interface nodes in the heterogeneous bar to support this hypothesis. A much smoother solution is observed at the Γ_{cI}, as the higher frequencies are filtered out.

Figures 24–26 depict the stress wave propagation through the quarter of the bar. The maximum values of stress are found in very localised portions of the domain, but well below failure values. A clipping of the solution from −20 GPa to 1 GPa has been utilised to better visualise the comparison of the stresses between monolithic and multi-time step solutions. The stress plots for both monolithic and multi-time step solutions match again, with more severe deformation shown at the interfaces between the differing materials. While the energy balance's absolute value is of a larger magnitude compared to Figure 20, the multi-time stepping solution and monolithic solution are of a comparable order. From this, we conclude that the simulation is stable. As the metaconcrete bar deforms more severely, we see a larger distortion in the elements, hence causing a variable time step ratio. This is shown on the right of Figure 27. The variation in Δt_I presents the use of α_s. We find that utilising the small reduction factor on the subdomain of the inclusion minimises the deviation from the maximum time step, rather than applying α_L onto the matrix Δt_m. Table 5 captures the runtimes and minimum time steps for the elasto-plastic simulation. Once again, we achieve a significant speedup of 3.25×, leading to a 69.22% decrease in runtime.

7 | Conclusion and Future Works

This paper presents a novel multi-time stepping integration algorithm for the investigation of wave propagation in heterogeneous structures. The algorithm integrates subdomains explicitly, without the constraint of a constant or integer time step ratio. Structures are partitioned, such that individual subdomains may step through time with a value close to their conditional limit. The continuity of acceleration and tractions is enforced at subdomain interfaces, without the need for resolving additional field variables. We quantify the energy associated with the

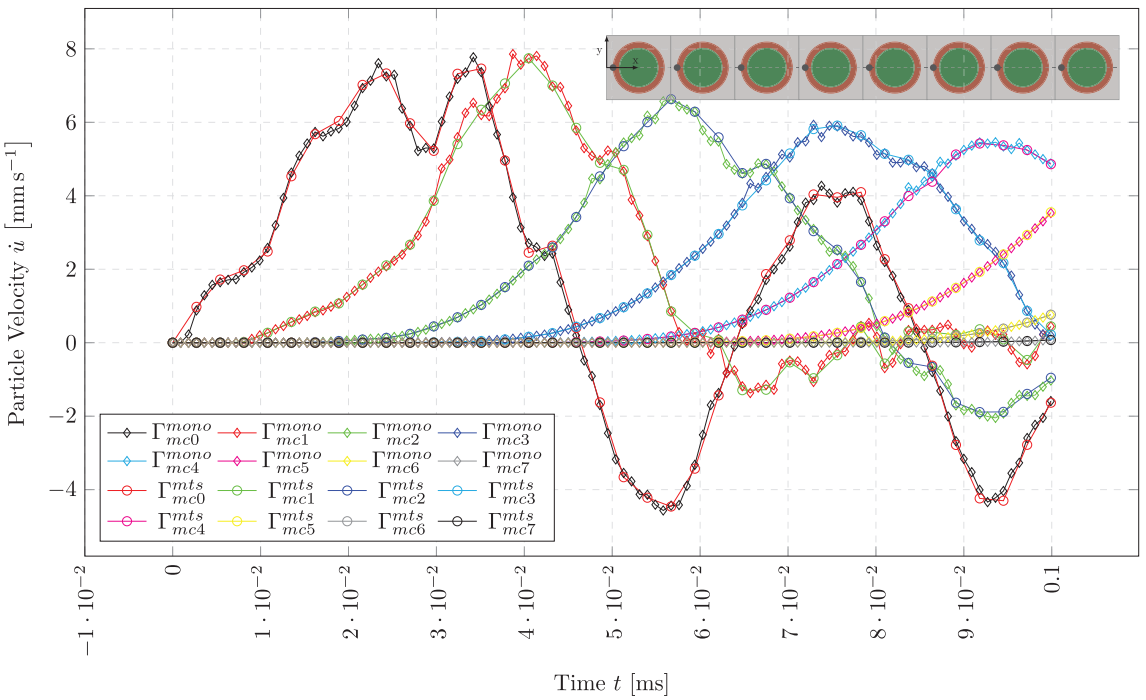

FIGURE 21 | Elasto-plastic wave propagation; matrix-coating velocity history of coupling nodes on Γ_{mc} for mono and mts solutions taken at coordinates {−12.0, 0.0, 0.0}, {18.0, 0.0, 0.0}, {48.0, 0.0, 0.0}, {78.0, 0.0, 0.0}, {108.0, 0.0, 0.0}, {138.0, 0.0, 0.0}, {168.0, 0.0, 0.0}, and {198.0, 0.0, 0.0}.

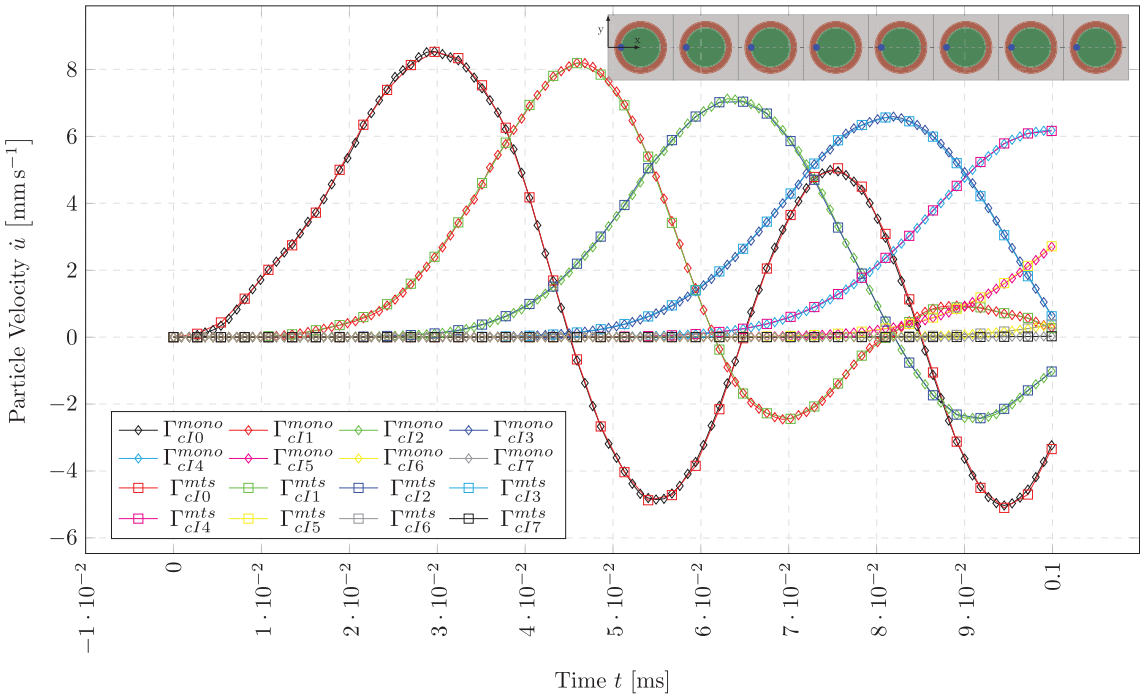

FIGURE 22 | Elasto-plastic wave propagation; coating-inclusion velocity history of coupling nodes on Γ_{cI} for mono and mts solutions taken at coordinates $\{-9.0, 0.0, 0.0\}$, $\{21.0, 0.0, 0.0\}$, $\{51.0, 0.0, 0.0\}$, $\{81.0, 0.0, 0.0\}$, $\{111.0, 0.0, 0.0\}$, $\{141.0, 0.0, 0.0\}$, $\{171.0, 0.0, 0.0\}$, and $\{201.0, 0.0, 0.0\}$.

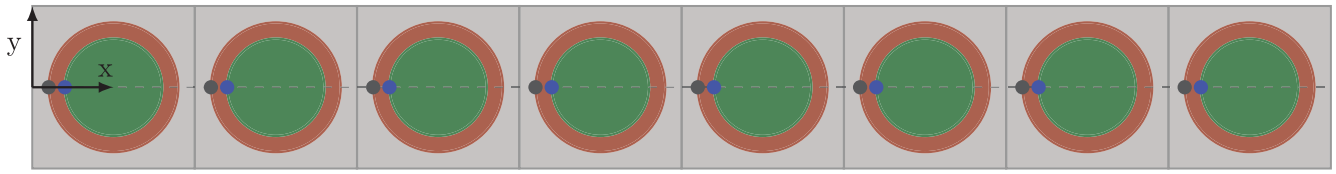

FIGURE 23 | Sampling points for elasto-plastic simulation taken along the $y = z = 0$ axis, with interfaces Γ_{mc} and Γ_{cI} representing interfaces between matrix-coating and coating-inclusion respectively.

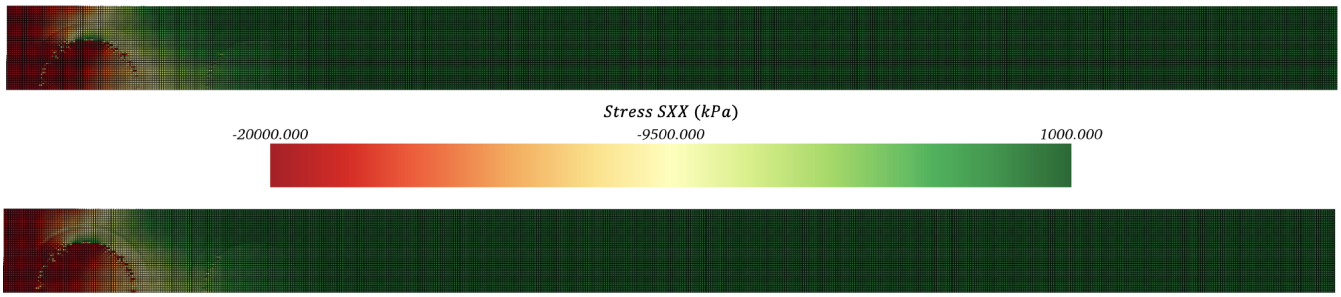

FIGURE 24 | Top: Monolithic solution $\Omega_{\sigma_{xx}}^{\text{mono}}$, Bottom: Multi-time step solution $\Omega_{\sigma_{xx}}^{\text{mts}}$; Elasto-plastic wave propagation at $t = 0.015$ ms.

FIGURE 25 | Top: Monolithic solution $\Omega_{\sigma_{xx}}^{\text{mono}}$, Bottom: Multi-time step solution $\Omega_{\sigma_{xx}}^{\text{mts}}$; Elasto-plastic wave propagation at $t = 0.0450$ ms.

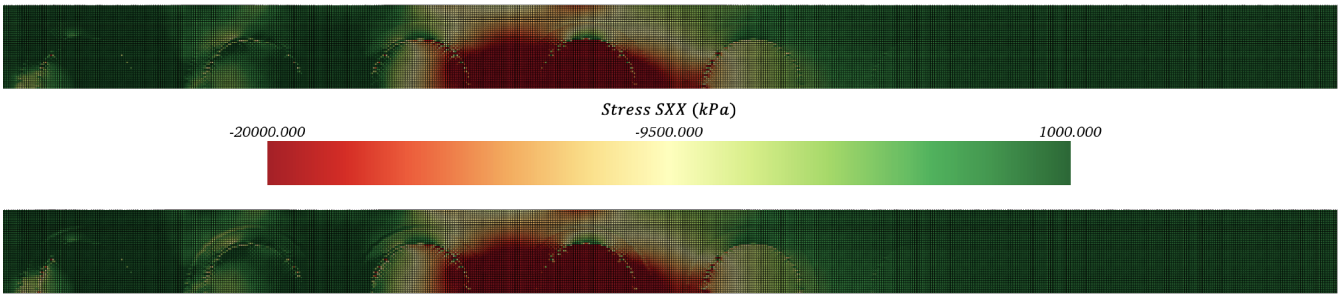

FIGURE 26 | Top: Monolithic solution $\Omega^{\text{mono}}_{\sigma_{xx}}$, Bottom: Multi-time step solution $\Omega^{\text{mts}}_{\sigma_{xx}}$; Elasto-plastic wave propagation at $t = 0.0600$ ms.

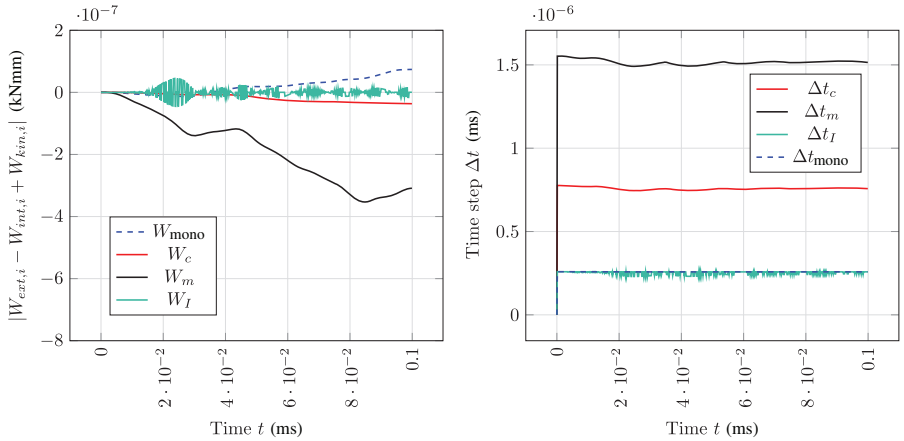

FIGURE 27 | Left: Energy balance check $|W_{\text{ext},i} - W_{\text{int},i} + W_{\text{kin},i}|$ of Elasto-plastic wave propagation through a Metaconcrete Bar with monolithic and multi-time step solution, Right: Time step history for each Ω_i.

TABLE 5 | Monolithic and Multi-Time Step runtimes and minimum time steps of the matrix Ω_m, coating Ω_c and inclusion Ω_I.

Elasto-plastic simulation	Runtime (s)	min{Δt} (ms)
Monolithic (single-time step)	396,780	2.577599e-7
Multi-Time Stepping (mts)	122,151	1.491133e-6 (Ω_m) | 7.455663e-7 (Ω_c) | 2.297713e-7 (Ω_I)

assumption of a constant acceleration at the interface, computing over both Δt_L and Δt_s. We compare to the state-of-the-art, under the severe conditions of a discontinuous square wave loading, depicting that we solve less integration steps, whilst also enforcing continuity of all three kinematic quantities. The two-domain algorithm can be utilised for multiple subdomains, with just the slight adaptation accounting for both temporal and spatial adjacency. The algorithm has been shown to robustly solve 3-D wave propagation problems, exhibiting significant speedup, compared to single-time step solutions, for both elastic and plastic metamaterials.

Variable time step ratio multi-time stepping algorithms are still a relatively nascent area of research, and many challenges remain. With the current interface coupling, the constant acceleration proves a limitation in capturing higher frequencies than a large time step can. Further work could be conducted in the adaptive solution of the behaviour on the interface, solving between Δt_L and Δt_s; dependent on the time step ratio and frequency content. Data-driven techniques could also be used to further speedup the computation of the reduction factor α_i and partitioning for multiple subdomains. Forecasting this change in Δt_i as the simulation progresses, offers an opportunity for investigation. Experimentally, scenarios where heterogeneous setups are required to simulate similar stress wave propagation problems are plentiful. Pulse shaping of Split Hopkinson tension bars [55, 56] and Taylor impact tests [57] are examples of such scenarios in the laboratory.

This paper focuses on the analysis of heterogeneous solids with a continuum-continuum coupling, however notable work has been completed within spatial multi-scale coupling applications that commonly necessitate the need for multiple time scales. Extensions of the Bridging Domain Method [58], with the multi-time step Verlet algorithm to couple continuum and molecular models offer potential research opportunities. Work by Liu et al. [59] and more recently, Li, Wang, and Jin [60] exemplify how multi-time stepping could benefit Discrete Element Modelling and Finite Element Modelling (DEM-FEM) couplings. Further extensions to the works of De Moerlose et al. [61] in fluid structure interaction (FSI), and multi-physics problems could also be solved with such non-constant multi-time stepping algorithms. Similar attention is required when addressing the coupling of such subdomains that do not necessarily conform in their spatial discretisations. This

becomes an exciting opportunity when modelling multi-scale problems both temporally and spatially.

Acknowledgements

The authors gratefully acknowledge the financial support from the Engineering and Physical Sciences Research Council (EPSRC) and Rolls-Royce's ASiMoV Prosperity Partnership with References EP/S005072/1 (Kin Fung Chan, Duygu Sap, and Nik Petrinic), EP/R029423/1 (Duygu Sap), EP/R513295/1 (David Wason) and Advanced Micro Devices Inc. (Nicola Bombace). The authors also thank the reviewers for their helpful comments, improving this manuscript.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are openly available upon request. The algorithm is described for two time-domains at https://github.com/kinfungchan/multi-time-step-integration. The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. R. Courant, K. Friedrichs, and H. Lewy, "On the Partial Difference Equations of Mathematical Physics," *IBM Journal of Research and Development* 11, no. 2 (1967): 215–234.

2. T. Belytschko and R. Mullen, "Stability of Explicit-Implicit Mesh Partitions in Time Integration," *International Journal for Numerical Methods in Engineering* 12, no. 10 (1978): 1575–1586.

3. T. Belytschko, H. J. Yen, and R. Mullen, "Mixed Methods for Time Integration," *Computer Methods in Applied Mechanics and Engineering* 17-18, no. 2 (1979): 259–275.

4. T. J. Hughes, K. S. Pister, and R. L. Taylor, "Implicit-Explicit Finite Elements in Nonlinear Transient Analysis," *Computer Methods in Applied Mechanics and Engineering* 17-18 (1979): 159–182.

5. A. Lew, J. E. Marsden, M. Ortiz, and M. West, "Asynchronous Variational Integrators," *Archive for Rational Mechanics and Analysis* 167 (2003): 85–146.

6. M. O. Neal and T. Belytschko, "Explicit-Explicit Subcycling with Non-Integer Time Step Ratios For Structural Dynamic Systems," *Computers and Structures* 31, no. 6 (1989): 871–880.

7. P. Smolinski, "Subcycling Integration with Non-Integer Time Steps for Structural Dynamics Problems," *Computers and Structures* 59, no. 2 (1996): 273–281.

8. W. J. T. Daniel, "Analysis and Implementation of a New Constant Acceleration Subcycling Algorithm," *International Journal for Numerical Methods in Engineering* 40, no. 15 (1998): 2841–2855.

9. A. Mota, I. Tezaur, and G. Phlipot, "The Schwarz Alternating Method for Transient Solid Dynamics," *International Journal for Numerical Methods in Engineering* 123 (2021): 5036–5071.

10. T. Belytschko and Y. Y. Lu, "Explicit Multi-Time Step Integration for First and Second Order Finite Element Semidiscretizations," *Computer Methods in Applied Mechanics and Engineering* 108, no. 3-4 (1993): 353–383.

11. A. Prakash and K. D. Hjelmstad, "A FETI-Based Multi-Time-Step Coupling Method for Newmark Schemes in Structural Dynamics," *International Journal for Numerical Methods in Engineering* 61, no. 13 (2004): 2183–2204.

12. M. Duval, J. C. Passieux, M. Salaün, and S. Guinard, "Non-Intrusive Coupling: Recent Advances and Scalable Nonlinear Domain Decomposition," *Archives of Computational Methods in Engineering* 23 (2016): 17–38.

13. A. Gravouil, A. Combescure, and M. Brun, "Heterogeneous Asynchronous Time Integrators for Computational Structural Dynamics," *International Journal for Numerical Methods in Engineering* 102, no. 3-4 (2015): 202–232.

14. T. Chantrait, J. Rannou, and A. Gravouil, "Low Intrusive Coupling of Implicit and Explicit Time Integration Schemes for Structural Dynamics: Application to Low Energy Impacts on Composite Structures," *Finite Elements in Analysis and Design* 86 (2014): 23–33.

15. A. Gravouil and A. Combescure, "Multi-Time-Step Explicit–Implicit Method for Non-Linear Structural Dynamics," *International Journal for Numerical Methods in Engineering* 50, no. 1 (2001): 199–225.

16. A. Gravouil and A. Combescure, "Multi-Time-Step and Two-Scale Domain Decomposition Method for Non-Linear Structural Dynamics," *International Journal for Numerical Methods in Engineering* 58, no. 10 (2003): 1545–1569.

17. S. Cho, R. Kolman, J. González, and K. Park, "Explicit Multistep Time Integration for Discontinuous Elastic Stress Wave Propagation in Heterogeneous Solids," *International Journal for Numerical Methods in Engineering* 118, no. 5 (2019): 276–302.

18. W. J. T. Daniel, "A Study of The Stability of Subcycling Algorithms in Structural Dynamics," *Computer Methods in Applied Mechanics and Engineering* 156, no. 1-4 (1998): 1–13.

19. P. Smolinski, "An Explicit Multi-Time Step Integration Method for Second Order Equations," *Computer Methods in Applied Mechanics and Engineering* 94, no. 1 (1992): 25–34.

20. N. Mahjoubi and S. Krenk, "Multi-Time-Step Domain Coupling Method With Energy Control," *International Journal for Numerical Methods in Engineering* 83, no. 13 (2010): 1700–1718.

21. H. B. Dhia, "Multiscale Mechanical Problems: The Arlequin Method," *Comptes Rendus de l'Academie des Sciences Series IIB Mechanics Physics Astronomy* 12, no. 326 (1998): 899–904.

22. H. B. Dhia and G. Rateau, "The Arlequin Method as a Flexible Engineering Design Tool," *International Journal for Numerical Methods in Engineering* 62, no. 11 (2005): 1442–1462.

23. A. Ghanem, M. Torkhani, N. Mahjoubi, T. Baranger, and A. Combescure, "Arlequin Framework for Multi-Model, Multi-Time Scale and Heterogeneous Time Integrators for Structural Transient Dynamics," *Computer Methods in Applied Mechanics and Engineering* 254 (2013): 292–308.

24. R. Dvořák, R. Kolman, M. Mračko, et al., "Energy-Conserving Interface Dynamics With Asynchronous Direct Time Integration Employing Arbitrary Time Steps," *Computer Methods in Applied Mechanics and Engineering* 413 (2023): 116110.

25. J. Selvaraj, L. Kawashita, and S. Hallett, "Mesh Independent Modelling of Tensile Failure in Laminates Using Mixed-Time Integration in Explicit Analysis," *Engineering Fracture Mechanics* 259 (2022): 108113.

26. S. Ghosh and J. Cheng, "Adaptive Multi-Time-Domain Subcycling for Crystal Plasticity FE Modeling of Discrete Twin Evolution," *Computational Mechanics* 61, no. 1 (2018): 33–54.

27. C. Grunwald, N. Durr, M. Sauer, W. Riedel, and S. Hiermaier, "A Concurrent Two-Scale Coupling for Wave Propagation Using Direct Solution Schemes With Explicit Time Integration," *International Journal for Numerical Methods in Engineering* 122, no. 21 (2021): 6361–6383.

28. F. Dunne and N. Petrinic, *Introduction to Computational Plasticity* (Oxford: OUP Oxford, 2005).

29. E. A. De Souza Neto, D. Peric, and D. R. Owen, *Computational Methods for Plasticity: Theory and Applications* (Oxford: John Wiley & Sons, 2011).

30. J. Banks, W. Henshaw, and D. Schwendeman, "An Analysis of a New Stable Partitioned Algorithm for FSI Problems. Part I: Incompressible Flow and Elastic Solids," *Journal of Computational Physics* 269 (2014): 108–137.

31. A. Combescure and A. Gravouil, "A Numerical Scheme to Couple Subdomains With Different Time-Steps for Predominantly Linear Transient Analysis," *Computer Methods in Applied Mechanics and Engineering* 191, no. 11-12 (2002): 1129–1157.

32. T. J. R. Hughes and W. K. Liu, "Implicit-Explicit Finite Elements in Transient Analysis: Stability Theory," *Journal of Applied Mechanics, Transactions ASM* 45, no. 2 (1978): 371–374.

33. T. J. R. Hughes and W. K. Liu, "Implicit-Explicit Finite Elements in Transient Analysis: Implementation and Numerical Examples," *Journal of Applied Mechanics, Transactions ASME* 45, no. 2 (1978): 375–378.

34. K. Park, S. Lim, and H. Huh, "A Method for Computation of Discontinuous Wave Propagation in Heterogeneous Solids: Basic Algorithm Description and Application to One-Dimensional Problems," *International Journal for Numerical Methods in Engineering* 91, no. 6 (2012): 622–643.

35. J. VonNeumann and R. D. Richtmyer, "A Method for the Numerical Calculation of Hydrodynamic Shocks," *Journal of Applied Physics* 21, no. 3 (1950): 232–237.

36. L. Maheo, V. Grolleau, and G. Rio, "Numerical Damping of Spurious Oscillations: a Comparison Between the Bulk Viscosity Method and the Explicit Dissipative Tchamwa–Wielgosz Scheme," *Computational Mechanics* 51 (2013): 109–128.

37. J. Darlington, "A Synthesis of Several Sorting Algorithms," *Acta Informatica* 11 (1978): 1–30.

38. K. Bronik, K. F. Chan, D. Sap, et al., "DEST's documentation," https://dest-doc.readthedocs.io/en/latest/ (2022).

39. N. Engheta and R. W. Ziolkowski, *Metamaterials: Physics and Engineering Explorations* (Oxford: John Wiley & Sons, 2006).

40. S. J. Mitchell, A. Pandolfi, and M. Ortiz, "Metaconcrete: Designed Aggregates to Enhance Dynamic Performance," *Journal of the Mechanics and Physics of Solids* 65 (2014): 69–81.

41. C. Kettenbeil and G. Ravichandran, "Experimental Investigation of the Dynamic Behavior of Metaconcrete," *International Journal of Impact Engineering* 111 (2018): 199–207.

42. D. Briccola, M. Cuni, A. De Juli, M. Ortiz, and A. Pandolfi, "Experimental Validation of the Attenuation Properties in the Sonic Range of Metaconcrete Containing Two Types of Resonant Inclusions," *Experimental Mechanics* 61 (2021): 515–532.

43. C. Xu, W. Chen, H. Hao, K. Bi, and T. M. Pham, "Experimental and Numerical Assessment of Stress Wave Attenuation of Metaconcrete Rods Subjected to Impulsive Loads," *International Journal of Impact Engineering* 159 (2022): 104052.

44. S. J. Mitchell, A. Pandolfi, and M. Ortiz, "Investigation of Elastic Wave Transmission in a Metaconcrete Slab," *Mechanics of Materials* 91 (2015): 295–303.

45. M. Khan, B. Li, and K. Tan, "Impact Load Wave Transmission in Elastic Metamaterials," *International Journal of Impact Engineering* 118 (2018): 50–59.

46. C. Xu, W. Chen, H. Hao, T. M. Pham, and K. Bi, "Damping Properties and Dynamic Responses of Metaconcrete Beam Structures Subjected to Transverse Loading," *Construction and Building Materials* 311 (2021): 125273.

47. C. Xu, W. Chen, H. Hao, and H. Jin, "Effect of Engineered Aggregate Configuration and Design on Stress Wave Attenuation of Metaconcrete Rod Structure," *International Journal of Solids and Structures* 232 (2021): 111182.

48. T. Belytschko, "An Overview of Semidiscretization and Time Integration Procedures," in *Computational Methods for Transient Analysis(A 84-29160)* (Amsterdam: *North-Holland*, 1983).

49. T. Belytschko, W. K. Liu, B. Moran, and K. Elkhodary, *Nonlinear Finite Elements for Continua and Structures* (Oxford: John wiley & sons, 2014).

50. S. J. Mitchell, A. Pandolfi, and M. Ortiz, "Effect of Brittle Fracture in a Metaconcrete Slab Under Shock Loading," *Journal of Engineering Mechanics* 142, no. 4 (2016): 04016010.

51. P. Frey, B. Sarter, and M. Gautherie, "Fully Automatic Mesh Generation for 3-D Domains Based Upon Voxel Sets," *International Journal for Numerical Methods in Engineering* 37, no. 16 (1994): 2735–2753.

52. B. van Rietbergen, H. Weinans, R. Huiskes, and B. Polman, "Computational Strategies for Iterative Solutions of Large FEM Applications Employing Voxel Data," *International Journal for Numerical Methods in Engineering* 39, no. 16 (1996): 2743–2767.

53. T. Belytschko, C. Parimi, N. Moës, N. Sukumar, and S. Usui, "Structured Extended Finite Element Methods for Solids Defined by Implicit Surfaces," *International Journal for Numerical Methods in Engineering* 56, no. 4 (2003): 609–635.

54. H. J. Kim and C. C. Swan, "Voxel-Based Meshing and Unit-Cell Analysis of Textile Composites," *International Journal for Numerical Methods in Engineering* 56, no. 7 (2003): 977–1006.

55. R. Gerlach, C. Kettenbeil, and N. Petrinic, "A New Split Hopkinson Tensile Bar Design," *International Journal of Impact Engineering* 50 (2012): 63–67.

56. J. Hoffmann, H. Cui, and N. Petrinic, "Determination of the Strain-Energy Release Rate of a Composite Laminate Under High-Rate Tensile Deformation in Fibre Direction," *Composites Science and Technology* 164 (2018): 110–119.

57. L. Zhang, A. Pellegrino, D. Townsend, and N. Petrinic, "Thermomechanical Constitutive Behaviour of a Near α Titanium Alloy Over a Wide Range of Strain Rates: Experiments and Modelling," *International Journal of Mechanical Sciences* 189 (2021): 105970.

58. S. Xiao and T. Belytschko, "A Bridging Domain Method for Coupling Continua With Molecular Dynamics," *Computer Methods in Applied Mechanics and Engineering* 193, no. 17-20 (2004): 1645–1669.

59. Y. Liu, W. Sun, Z. Yuan, and J. Fish, "A Nonlocal Multiscale Discrete-Continuum Model for Predicting Mechanical Behavior of Granular Materials," *International Journal for Numerical Methods in Engineering* 106, no. 2 (2016): 129–160.

60. T. Li, Q. Wang, and X. Jin, "A Staggered Asynchronous Step Integration Algorithm for Hybrid Finite-Element and Discrete-Element Modeling," *International Journal of Computational Methods* 19, no. 2 (2022): 2150064.

61. L. De Moerloose, L. Taelman, P. Segers, J. Vierendeels, and J. Degroote, "Analysis of Several Subcycling Schemes in Partitioned Simulations of a Strongly Coupled Fluid-Structure Interaction," *International Journal for Numerical Methods in Fluids* 89, no. 6 (2019): 181–195.

Appendix A

Two-Domain Multi-Time Step Integration Algorithm

We succinctly outline each of the operations with their corresponding equations for the two-domain multi-time stepping algorithm. Here we discuss a case where large subdomain Ω_L and Ω_s are integrated separately with a common interface of Γ. We include the computations at

$t_s = t_L = 0.0$, where for a mass conserving problem, the summation of masses on Γ takes place. The rest of the algorithm follows on from the computations described in Section 2.

Square Wave Propagation With a High Time Step Ratios

In this benchmark, we simulate a highly heterogeneous domain, such that the time step ratio $m = 100$. The domain Ω is split into two subdomains Ω_L and Ω_s with isotropic elastic properties of $E_L = E_s = 0.02$ GPa and two densities $\rho_L = 8000$ kgm^{-3}, $\rho_s = 0.8$ kgm^{-3}. Both subdomains share the same spatial discretisation, and have consecutive lengths $L_L = 50$ mm and $L_s = 100$ mm. The same square wave function as the $m = \pi$ numerical example, in Section 3.2, is utilised for the prescribed velocity boundary condition. Ω_L and Ω_s are meshed with the same $h_e = 0.1667$ mm, where time steps initially integrate with $Co = 0.5$ and $\Delta t_L = 1.667e-06$ s and $\Delta t_s = 1.667e-08$ s. As previously mentioned, when solving for severely heterogeneous problems, the proposed method may suffer from instability on the interface of two materials with vastly different dilatational wave speeds c_e. In Figure A1, we depict one of the simple remedies to the problem; shifting the temporal interface Γ away from the material interface.

Consequently, an extra element from Ω_L now solves with Δt_s; the same integration step as the rest of subdomain Ω_s. We plot the velocity distribution of the square wave propagation at three distinct time steps, comparing the single-time step (monolithic) solution to our proposed multi-time stepping (MTS) method. Figure A2 captures the behaviour of the wave following its full application, travelling through Ω_L. Similar conclusions to that of the $m = \pi$ example can be made at this time step, where bulk viscosity is applied as a means to reduce dispersion in the solution.

Figure A3 illustrates the interaction of the wave with the high impedance interface. From this, the transmitted wave is observed to be double the initial amplitude, comparing well to the analytical solution of a square wave. In Ω_s, a drastic increase in c_e leads to the wave travelling back and forth from the material interface to the end of the bar. Both monolithic and MTS align well in their velocity distributions. Figure A4 shows the reflection of the wave back into Ω_L, where its maximum amplitude is very similar to this initial wave; given the highly heterogeneous configuration.

Integrating an extra element with Δt_s, or reducing m, reduces the computational efficiency of the simulation. When resolving the material interface for highly heterogeneous problems with the proposed algorithm, numerical strategies such as the aforementioned are necessary. Despite this, we note that a key advantage is that there is no change in algorithm, even for these limit cases.

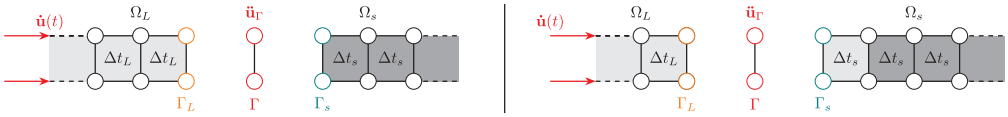

FIGURE A1 | Left: Conventional definition of Γ as seen in $m = \pi$ benchmark; Right: Shifting Γ for Highly heterogeneous material interfaces allowing for high frequency behaviour to be resolved by Δt_s.

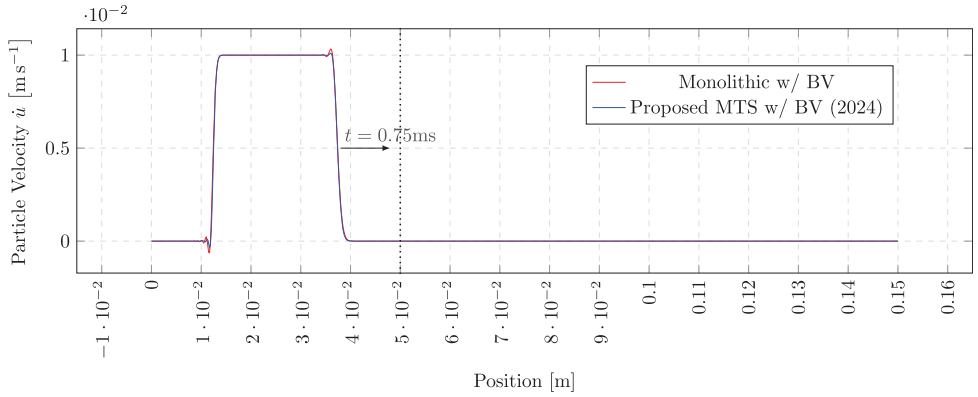

FIGURE A2 | Square wave propagation through a 1-D heterogeneous bar at $t = 0.75$ ms with Monolithic and the proposed method with bulk viscosity.

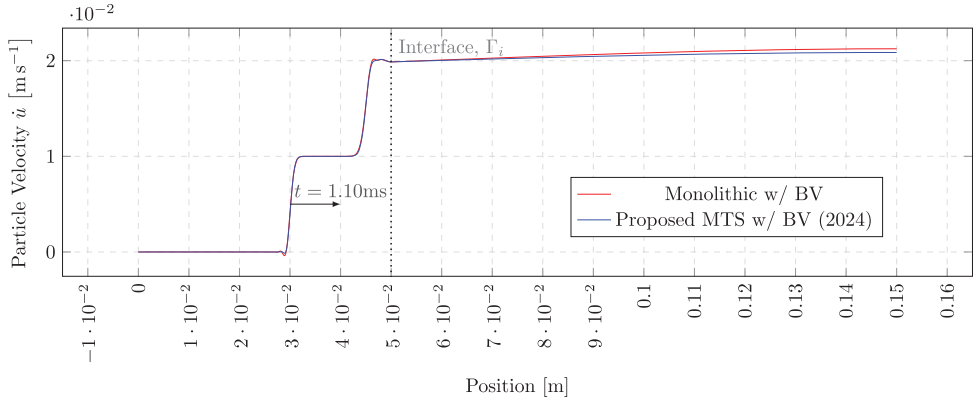

FIGURE A3 | Square wave propagation through a 1-D heterogeneous bar at $t = 1.10$ ms with Monolithic and the proposed method with bulk viscosity.

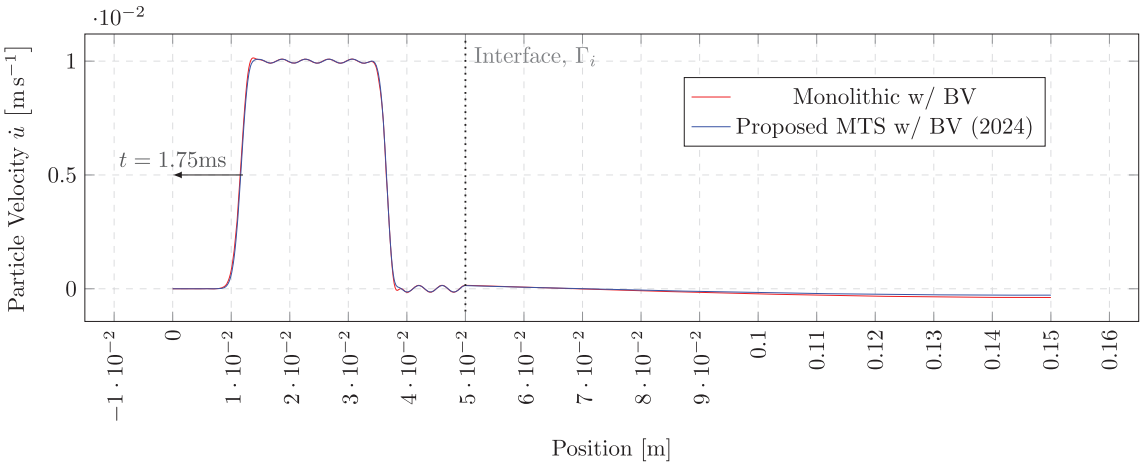

FIGURE A4 | Square wave propagation through a 1-D heterogeneous bar at $t = 1.77$ ms with Monolithic and the proposed method with bulk viscosity.