Preprints.org

Article

Temporal and Spatial Coupling Methods for the Efficient Modelling of Dynamic Solids

Kin Fung Chan [*] , Nicola Bombace , Indrajeet Sahu , Simone Falco , Nik Petrinic

Article

Temporal and Spatial Coupling Methods for the Efficient Modelling of Dynamic Solids

Kin Fung Chan [1,*] [iD]**, Nicola Bombace** [2]**, Indrajeet Sahu** [1] [iD]**, Simone Falco** [1] **and Nik Petrinic** [1] [iD]

[1] Department of Engineering Science, University of Oxford, Parks Road, Oxford, OX1 3PJ, UK
[2] Adaptive, Embedded and AI (AEAI) Group, Advanced Micro Devices Inc., Darwin House, Edinburgh Technopole, Bush Estate, EH26 0PY, Edinburgh, UK
* Correspondence: kin.chan@eng.ox.ac.uk

Abstract: This paper presents efficient coupling methods that accurately reduce the computational cost for modelling solids dynamically with finite elements. A multi-time step integration algorithm is developed to leverage varying time steps throughout a domain. Interfaces between subdomains are resolved explicitly with the continuity of acceleration and tractions. A spatial coupling method is combined with multiple time steps, allowing for meshes that do not necessarily conform at their interfaces. The method avoids solving additional degrees of freedom at these interfaces, with parameter-free coupling operators defined between meshes. A speedup >12X is achieved in comparison to reference single time step methods.

Keywords: multi-time stepping; non-matching meshes; explicit finite elements

1. Introduction

The temporal and spatial discretisation of structural dynamic problems is directly related to the accuracy and computational cost of the explicit finite element method. Constrained by the Courant-Friedrichs-Lewy (CFL) condition, the critical time step, Δt_C is proportional to the element size, and inversely to the dilatational wave speed [1]. This leads to simulations being restricted to $\min\{\Delta t\}$ of the element with the smallest size or highest wave speed. Initially referred to as subcycling, pioneering works allowed for the integration of multiple time steps in a single domain [2,3]. The coupling of various kinematic fields was explored, along with the stability of such algorithms [4,5]. Asynchronous variational integration is briefly mentioned as an alternative; that discretise the functional instead of the equations of motion, with varying time steps [6]. The main drawback being the high complexity of its implementation. The importance of varying, non-integer and large time step ratios was also extended to, with heterogeneous asynchronous time integration [7,8]. However, maintaining the continuity of kinematics at the interface remains a challenge, especially for all three fields [9]. In recent works, energy conserving methods have been developed, however they depart far from the CFL condition to resolve the interface conditions at consistent time steps [10]. Spatially, non-matching mesh algorithms facilitate more flexible geometric modelling [11]. Nitsche's method has shown to weakly enforce conditions on the interfaces of non-matching meshes without additional unknowns, but commonly suffers from the sensitivity to parameters [12–15]. Following similar principles of weak continuity, Lagrange multipliers are called upon in the mortar-based methods [16–20]. These types of methods have proven to be very robust, however struggle to fulfil the inf-sup stability condition, and incur a large computational cost with mapping master and slave nodes [21]. In comparison, the use of localised Lagrange multipliers, introduces a frame that independently enforces compatibility constraints on each boundary [22–27]. However, as common with other methods that introduce an additional discretised interface, the construction of this interface is neither trivial or computationally cheap [28–31]. This brief review justifies the need for more efficient couplings; both temporally and spatially. Coupling algorithms that allow for varying time step ratios, whilst stepping close to the CFL

condition, concertedly those that solve for non-matching meshes, without increasing the degrees of freedom, remain a hot topic of research.

2. Governing Equations of Dynamic Solids

The problem of a solid body subject to impact, is described through the partitioning of a domain as illustrated in Figure 1. The deformation is governed by the momentum balance equation acting on solid domain Ω:

$$\rho\ddot{\mathbf{u}} = \nabla \cdot \sigma + \rho\mathbf{b}, \text{ in } \Omega \times [0, T] \tag{1}$$

where ρ, $\ddot{\mathbf{u}}$, σ and $\mathbf{b}$ denote the density, acceleration field, stress tensor and body forces. Deformation is described at time $t \in [0, T]$ for a specified constant $T > 0$. Updated Lagrangian formulations of a single body are found in the following [32–34], here we extend this to a partitioned formulation to solve multiple solid subdomains.
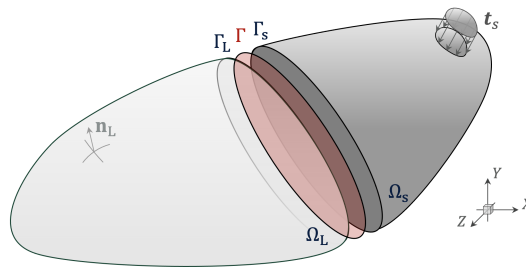

Figure 1. A 3-D domain Ω decomposed into Ω_L and Ω_s where Γ_L and Γ_s are non-matching interfaces to be externally resolved on Γ. Normal vector and tractions are visualised on Ω_L and Ω_s respectively.

Defining multiple subdomains, we state $\mathcal{B}$ is a solid body in an open region $\Omega \subset \mathbb{R}^3$, with its boundary denoted $\partial\Omega$. Ω is partitioned into $\mathcal{S}$ non-overlapping subdomains:

$$\Omega = \bigcup_{i=1}^{S} \Omega_i \text{ and } \Omega_i \cap \Omega_j = \varnothing \text{ for } i \neq j \tag{2}$$

Starting from the simplest case, $\mathcal{S} = 2$, for a two-subdomain partitioning, where Ω_L and Ω_s represent the large and small subdomains. The boundary between the two is defined Γ, where for matching conditions we enforce:

$$\ddot{\mathbf{u}}_L = \ddot{\mathbf{u}}_s \text{ on } \Gamma \tag{3}$$

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

to ensure the continuity of acceleration $\ddot{\mathbf{u}}$ and tractions $\mathbf{t}$, as well as enforcing Dirichlet and Neumann boundary conditions. The variational formulation of the dynamic equilibrium in Equation 1 can be described for both Ω_L and Ω_s as the following:

$$\int_{\Omega_L} \rho_L \ddot{\mathbf{u}}_L \cdot \delta\dot{\mathbf{u}} d\Omega = \int_{\Gamma_L} \mathbf{t}_L \cdot \delta\dot{\mathbf{u}} d\Gamma - \int_{\Omega_L} \sigma_L : \delta\mathbf{D}_L d\Omega + \int_{\Omega_L} \rho_L \mathbf{b}_L \cdot \delta\dot{\mathbf{u}} d\Omega \tag{5}$$

$$\int_{\Omega_s} \rho_s \ddot{\mathbf{u}}_s \cdot \delta\dot{\mathbf{u}} d\Omega = \int_{\Gamma_s} \mathbf{t}_s \cdot \delta\dot{\mathbf{u}} d\Gamma - \int_{\Omega_s} \sigma_s : \delta\mathbf{D}_s d\Omega + \int_{\Omega_s} \rho_s \mathbf{b}_s \cdot \delta\dot{\mathbf{u}} d\Omega \tag{6}$$

where we denote a variational velocity $\delta\dot{\mathbf{u}} \in \mathcal{V}_0$, in a space $\mathcal{V}_0$ where $\forall \delta\dot{\mathbf{u}} \in H^1(\Omega_i)$, and $\mathbf{D}$ as rate of deformation. A discrete approximation of the variational form can be reduced to the ordinary differential equations:

$$\mathbf{M}_L \ddot{\mathbf{u}}_L = \mathbf{f}_L^{\text{ext}} - \mathbf{f}_L^{\text{int}}; \quad \mathbf{M}_s \ddot{\mathbf{u}}_s = \mathbf{f}_s^{\text{ext}} - \mathbf{f}_s^{\text{int}} \tag{7}$$

where for subdomains Ω_L and Ω_s, we sum over a number of finite elements, $\mathcal{N}_L$ and $\mathcal{N}_s$, in each subdomain:

$$\mathbf{M}_L = \sum_{e=1}^{\mathcal{N}_L} \int_{\Omega_e} \rho_L \mathbf{N}^T \mathbf{N} d\Omega_e; \quad \mathbf{M}_s = \sum_{e=1}^{\mathcal{N}_s} \int_{\Omega_e} \rho_s \mathbf{N}^T \mathbf{N} d\Omega_e \tag{8}$$

$$\mathbf{f}_L^{\text{ext}} = \sum_{e=1}^{\mathcal{N}_L} \int_{\Omega_e} \mathbf{N}^T \mathbf{t}_L d\Omega_e; \quad \mathbf{f}_s^{\text{ext}} = \sum_{e=1}^{\mathcal{N}_s} \int_{\Omega_e} \mathbf{N}^T \mathbf{t}_s d\Omega_e \tag{9}$$

$$\mathbf{f}_L^{\text{int}} = \sum_{e=1}^{\mathcal{N}_L} \int_{\Omega_e} \mathbf{B}^T \sigma_L d\Omega_e \quad \mathbf{f}_s^{\text{int}} = \sum_{e=1}^{\mathcal{N}_s} \int_{\Omega_e} \mathbf{B}^T \sigma_s d\Omega_e \tag{10}$$

The Lagrangian shape functions are represented by $\mathbf{N}$, with their derivatives denoted $\mathbf{B}$. As common within the explicit finite element method, $\mathbf{M}$ is lumped for each subdomain, with $\mathbf{f}^{\text{ext}}$ and $\mathbf{f}^{\text{int}}$ both computed with vectors too. We elect to use the leapfrog time integration scheme to step through time, staggering the solution of each kinematic quantity such that:

$$\ddot{\mathbf{u}}_L^n = \mathbf{M}_L^{-1}(\mathbf{f}_L^{\text{ext}} - \mathbf{f}_L^{\text{int}}); \quad \ddot{\mathbf{u}}_s^n = \mathbf{M}_s^{-1}(\mathbf{f}_s^{\text{ext}} - \mathbf{f}_s^{\text{int}}) \tag{11}$$

$$\dot{\mathbf{u}}_L^{n+1/2} = \dot{\mathbf{u}}_L^{n-1/2} + \ddot{\mathbf{u}}_L^n \cdot \Delta t_L; \quad \dot{\mathbf{u}}_s^{n+1/2} = \dot{\mathbf{u}}_s^{n-1/2} + \ddot{\mathbf{u}}_s^n \cdot \Delta t_s \tag{12}$$

$$\mathbf{u}_L^{n+1} = \mathbf{u}_L^n + \dot{\mathbf{u}}_L^{n+1/2} \cdot \Delta t_L; \quad \mathbf{u}_s^{n+1} = \mathbf{u}_s^n + \dot{\mathbf{u}}_s^{n+1/2} \cdot \Delta t_s \tag{13}$$

noting that a diagonal mass matrix allows for a direct computation of acceleration. Velocity $\dot{\mathbf{u}}$ is computed on the half time step, with displacement $\mathbf{u}$ found each full time step. Next we summarise the temporal coupling, enabled by multi-time step (MTS) integration.

3. Multi-Time Step Integration

Multi-time stepping enables partitioned subdomains Ω_L and Ω_s to integrate with Δt_L and Δt_s, respectively. However, to allow for this difference in time steps, special attention must be given to the solution of the interface Γ. Crucially, the conditional stability of explicit methods require an element's time step to obey the CFL condition for a linear undamped system:

$$\Delta t_C = \frac{2}{\omega_C} \le \min_e \left(\frac{h_e}{c_e} \right) \tag{14}$$

Here we represent Δt_C as the critical time step, ω_C as the maximum eigenfrequency, h_e as the characteristic length of an element, and c_e the dilatational (longitudinal) wave speed.

3.1. Salient Multi-Time Stepping Features

The asynchronous integration is enabled with three important groups of computations. The first being the explicit computation of the acceleration on the interface Γ:

$$\mathbf{M}_\Gamma = (\mathbf{C}_s^T \mathbf{M}_s \mathbf{C}_s) + (\mathbf{C}_L^T \mathbf{M}_L \mathbf{C}_L) \tag{15}$$

$$\mathbf{f}_\Gamma^{\text{int}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{int}} + \mathbf{C}_L^T \mathbf{f}_L^{\text{int}}; \quad \mathbf{f}_\Gamma^{\text{ext}} = \mathbf{C}_s^T \mathbf{f}_s^{\text{ext}} + \mathbf{C}_L^T \mathbf{f}_L^{\text{ext}} \tag{16}$$

We define indicator matrices (vectors in 1-D) for each subdomain $\mathbf{C}$ to identify the degrees of freedom on the interface of subdomains. These summations provide the ingredients for computing the interface acceleration as:

$$\ddot{\mathbf{u}}_\Gamma = \mathbf{M}_\Gamma^{-1}(\mathbf{f}_\Gamma^{\text{ext}} - \mathbf{f}_\Gamma^{\text{int}}) \tag{17}$$

where we compute $\ddot{\mathbf{u}}_\Gamma$ at each large time step Δt_L. The integration of the subdomains is controlled by the definition of the time step ratios. Suppose the two subdomains begin at a similar point in time $t_L^N = t_s^n$ where N and n are the small and large steps respectively. The time after the maximum stable integration step on each subdomain is referred to as the *trial time* t_T, such that:

$$t_{TL}^{N+1} = t_L + \Delta t_{CL}; \quad t_{Ts}^{n+k} = t_s + \Delta t_{Cs} \tag{18}$$

where for every Δt_L, the number of small time steps k elapse since the last point in time where subdomains are equal in time. Now we can define the *current time step ratio*, t_{ratio}^{n+k}, and *next time step ratio*, t_{ratio}^{n+k+1} for the advancement of Ω_s with:

$$t_{ratio}^{n+k} = \frac{t_s^{n+k} - t_L^N}{t_{TL}^{N+1} - t_L^N}; \quad t_{ratio}^{n+k+1} = \frac{(t_s^{n+k} + \Delta t_s^{n+k}) - t_L^N}{t_{TL}^{N+1} - t_L^N} \tag{19}$$

Starting from each common time step with the integration of Ω_s, the number of small time steps Δt_s is determined by the evaluation of time step ratios t_{ratio}^{n+k} and t_{ratio}^{n+k+1}. If the condition of $t_{ratio}^{n+k+1} \leq 1$ or ($t_{ratio}^{n+k} \leq 1$ and $t_{ratio}^{n+k+1} \leq 1$) is satisfied, further steps on Δt_s can proceed before integrating Ω_L over Δt_L. As a consequence of subdomains integrating over their own respective time step, we ensure that the proposed method still finds a common time between all subdomains. Following the small trial time t_{Ts} exceeding the large trial time t_{TL}, the method proceeds to compute two additional ratios α_L and α_s. These denote the *reduction factors* required to maintain the subdomains in synchronisation where $0 \leq \alpha \leq 1$. We compute:

$$\alpha_L = 1 - \frac{(t_{TL}^N - t_s^{n+k})}{(t_{TL}^{N+1} - t_L^N)}; \quad \alpha_s = 1 - \frac{(t_{Ts}^{n+k} - t_{TL}^{N+1})}{(t_{Ts}^{n+k+1} - t_s^{n+k})} \tag{20}$$

Through computing α on both subdomains, we can compare reduction factors, such that:

$$t_L^{N+1} = t_s^{n+k} = \begin{cases} t_s^n + \sum_{k=0}^{k-1} \Delta t_{Cs}^{n+k} + \alpha_s \cdot \Delta t_{Cs}^{n+k}, & \alpha_s > \alpha_L \\ t_s^n + \alpha_L \cdot \Delta t_{CL}^N, & \alpha_L \geq \alpha_s \end{cases} \tag{21}$$

where we look to reduce the time step of the subdomain closest to CFL condition. In the following section we summarise the multi-time stepping algorithm, with each of these key features, as well as its implementation.

3.2. Summary of Temporal Algorithm

We provide an overview of the method required to integrate Ω_L and Ω_s with Δt_L and Δt_s. It shows a single large step N, exemplifying each of the features mentioned above. A note that $TOL = 1e\text{-}6$ is used for when $\Delta t_L \approx \Delta t_s$, and computation of $\mathbf{M}_\Gamma$ in Equation 15 is only required at $t = 0$ for a mass-conserving problem.

Algorithm 1 Summary of Algorithm for Coupling in Time from N to $N+1$

1: **procedure** A TWO-SUBDOMAIN MULTI TIME INTEGRATION STEP
2:     **while** $t_{ratio}^{n+k+1} \leq 1$ or ($t_{ratio}^{n+k} \leq 1$ and $t_{ratio}^{n+k+1} \leq 1+$TOL) **do**
3:         Integrate subdomain Ω_s with $\ddot{\mathbf{u}}_\Gamma$ and compute force vectors $\mathbf{f}_s^{\text{int}}, \mathbf{f}_s^{\text{ext}}$ over Δt_s
4:         Compute trial times $t_{Ts}^{n+k}, t_{TL}^{N+1}$ and time step ratios $t_{ratio}^{n+k}, t_{ratio}^{n+k+1}$
5:     Compute time step reduction factors α_L, α_s
6:     **if** $\alpha_L \geq \alpha_s$ **then**
7:         $\Delta t_L = \alpha_L \cdot \Delta t_L$
8:     **else**
9:         $\Delta t_s = \alpha_s \cdot \Delta t_s$
10:         Integrate subdomain Ω_s with $\ddot{\mathbf{u}}_\Gamma$ and recompute force vectors $\mathbf{f}_s^{\text{int}}, \mathbf{f}_s^{\text{ext}}$ over Δt_s
11:         Recompute trial times $t_{Ts}^{n+k}, t_{TL}^{N+1}$ and time step ratios $t_{ratio}^{n+k}, t_{ratio}^{n+k+1}$
12:     Integrate subdomain Ω_L with $\ddot{\mathbf{u}}_\Gamma$ and compute force vectors $\mathbf{f}_L^{\text{int}}, \mathbf{f}_L^{\text{ext}}$ over Δt_L
13:     Compute interface acceleration $\ddot{\mathbf{u}}_\Gamma$ with Eqs. 15 - 17 for next time step
14:     Recompute trial times $t_{Ts}^{n+k}, t_{TL}^{N+1}$ and time step ratios $t_{ratio}^{n+k}, t_{ratio}^{n+k+1}$

The proposed method is implemented in an open-source python code, found in the following repository https://github.com/kinfungchan/multi-time-step-integration. It contains re-implementations of methods from literature for the two-subdomain case [9,10]. Whilst we depict the case of just two subdomains, the algorithm can be extended to multiple by processing subdomains as pairs as shown in [35].

3.3. Numerical Examples in Time

We present a numerical example in 1-D, with the elastic wave propagation through a heterogeneous bar. Suppose the domain Ω is split into two subdomains Ω_L and Ω_s of similar discretisation, with isotropic elastic properties of $E_L = 207$ GPa, $E_s = 1000$ GPa and $\rho_L = \rho_s = 7.83 \times 10^{-6}$ kgmm^{-3}. These material properties result in a non-integer $m = 2.19$, where the time step ratio is solely driven by the dissimilar material properties. Figure 2 depicts the bar configuration. The velocity boundary condition is applied to Ω_L at $x = 0$ with $\dot{u}(t) = 0.01\sin(2\pi\omega_L t)$ ms^{-1} where we define a half sine wave with a frequency of $\omega_L = (125((7.83 \times 10^{-6})/207))^{-1}$ rads^{-1}. The difference in material impedance results in a portion of the incident wave being transmitted and the remainder reflected in the opposite direction.
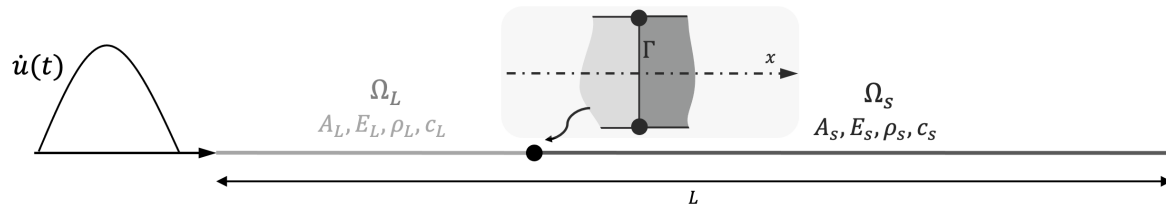

Figure 2. A one-dimensional heterogeneous domain Ω split into large subdomain Ω_L and small subdomain Ω_s, solved with Δt_L and Δt_s respectively, of length $L_L = 125$ mm and $L_s - 250$ mm The problem assuming uni-axial motion with $\nu = 0$. A compressive half sine velocity boundary condition is applied from Ω_L.

In Figure 3, a comparison between the coupled solution (Ω_L with Ω_s) and the monolithic (single time step) solution is presented at four separate time stamps. The multi-time step solution solves $\dot{\mathbf{u}}_L$ and $\dot{\mathbf{u}}_s$ over Δt_L and Δt_s, whereas $\dot{\mathbf{u}}_{mono}$ is limited by Δt_s. Consequently, for $m = 2.19$, our method reduces the number of integration steps on Ω_L by half. From prescription of the full wave at $t1$, through to the transmission and reflection of the wave at $t4$, the MTS solution aligns very well with the single time step solution, despite the halving the number of large time steps. This reduction in computational effort is even more prominent for highly heterogeneous configurations, as well as variance in spatial discretisation.
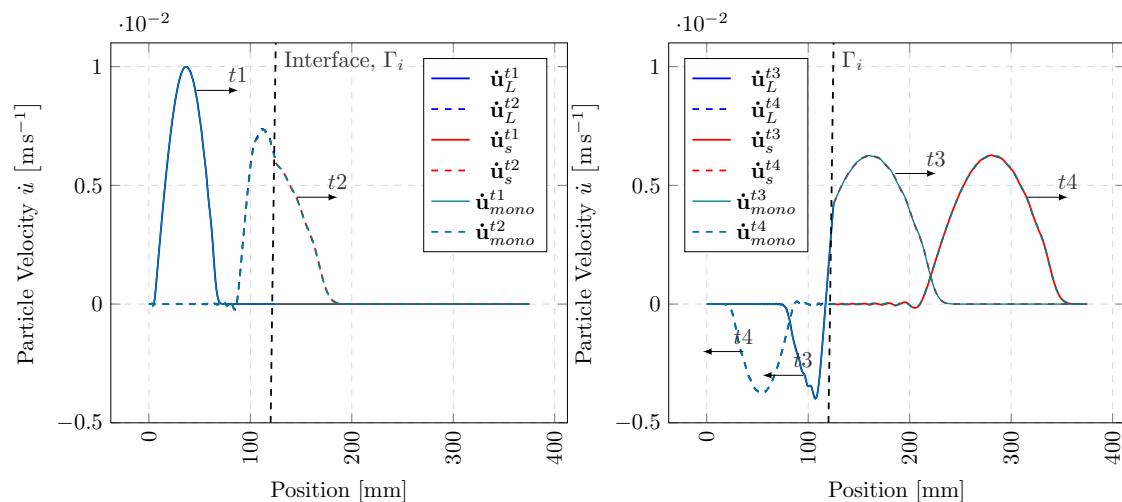

Figure 3. Axial wave propagation in a heterogeneous bar: (a) - boundary condition at $t1 = 0.03363$ ms and initial transmission at $t2 = 0.04424$ ms of the stress wave; (b) - transmission and reflection of the stress wave at $t3 = 0.01323$ ms and $t4 = 0.02920$ ms.

For the above simulation we also compute the energy components of each subdomain with the following:

$$W_{\text{ext}}^{n+1} = W_{\text{ext}}^n + \frac{\Delta t_i^{n+1/2}}{2}(\dot{\mathbf{u}}_i^{n+1/2})^{\text{T}}(\mathbf{f}_{\text{ext}}^n + \mathbf{f}_{\text{ext}}^{n+1}) = W_{\text{ext}}^n + \frac{1}{2}\Delta\mathbf{u}_i^{\text{T}}(\mathbf{f}_{\text{ext}}^n + \mathbf{f}_{\text{ext}}^{n+1}) \tag{22}$$

$$W_{\text{int}}^{n+1} = W_{\text{int}}^n + \frac{\Delta t_i^{n+1/2}}{2}(\dot{\mathbf{u}}_i^{n+1/2})^{\text{T}}(\mathbf{f}_{\text{int}}^n + \mathbf{f}_{\text{int}}^{n+1}) = W_{\text{int}}^n + \frac{1}{2}\Delta\mathbf{u}_i^{\text{T}}(\mathbf{f}_{\text{int}}^n + \mathbf{f}_{\text{int}}^{n+1}) \tag{23}$$

$$W_{\text{kin}}^n = \frac{1}{2}(\dot{\mathbf{u}}_i^{n+1/2})^{\text{T}}\mathbf{M}\dot{\mathbf{u}}_i^{n+1/2} \tag{24}$$

where n can be interchanged with N when evaluating Ω_L. The balance of energy can be evaluated in a similar way to the works of Neal and Belytschko [3], with the following:

$$|W_{\text{ext}} - W_{\text{int}} + W_{\text{kin}}| \leq ||W_{bal}|| \tag{25}$$

In Figure 4 we show each of the components of energy and its overall balance. For both monolithic and multi-time step solutions a smooth transition of energy is observed as the wave interacts with Γ. Remarkably, as the temporal coupling is enforced, the W_{bal} for both Ω_L and Ω_s is of the order 1×10^{-13} kNmm, significantly below each of the components at 1×10^{-8} kNmm. This numerical example captures the propagation of a smooth pulse, however severe loading cases, as well as a 3-D problems, remain ongoing work.
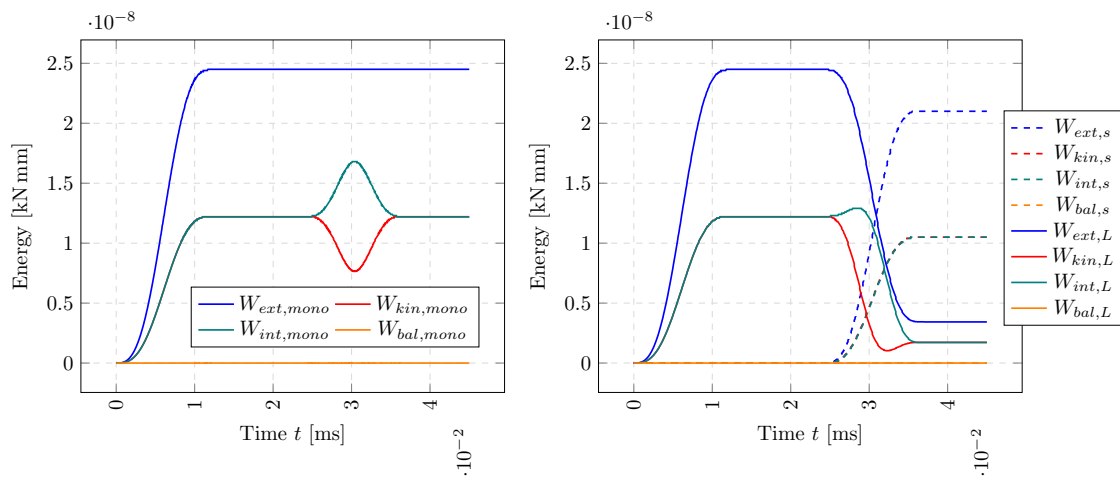

Figure 4. Energy balance for axial wave propagation problem: (a) - Monolithic (single Δt_{mono}) simulation system energy component history and balance with Equations 22 - 25, (b) - Multi-time stepping (Δt_L and Δt_s) simulation energy balance.

4. Solving Non-Matching Meshes

The problem of non-matching meshes is commonly found when simulating the dynamical behaviour of solids. We present an algorithm, combined with multi-time stepping, that relaxes the constraint of these conforming nodes, allowing for a coarser representation of a subdomain to be utilised; hence reducing the computational overhead.

4.1. Combined Spatial and Temporal Coupling

The following section follows on from governing equations defined in Section 2, however we now allow for the non-overlapping interface $\Gamma = \Gamma_L \cap \Gamma_s$ to consist of two separate spatial discretisations. Their compatibility is maintained such that:

$$\ddot{\mathbf{u}}_{\Gamma L}(\mathbf{C}_L\mathbf{x}_L) = \ddot{\mathbf{u}}_{\Gamma s}(\mathbf{C}_s\mathbf{x}_L) = \ddot{\mathbf{u}}_{\Gamma}(\mathbf{x}_{\Gamma}) \tag{26}$$

$$\mathbf{t}_{\Gamma L}(\mathbf{C}_L\mathbf{x}_L) = \mathbf{t}_{\Gamma s}(\mathbf{C}_s\mathbf{x}_s) = \mathbf{t}_{\Gamma}(\mathbf{x}_{\Gamma}) \tag{27}$$

where positions use incidence $\mathbf{C}$ matrix, such that $\mathbf{C}_L\mathbf{x}_L \in \Gamma_L$ and $\mathbf{C}_s\mathbf{x}_s \in \Gamma_s$ for each subdomain, and externally assembled interface contains $\mathbf{x}_\Gamma \in \Gamma$ to describe the common boundary between the subdomains. The total virtual power can be summated for two subdomains to give $\delta P = \delta P_L + \delta P_s + \delta P_\Gamma$, where a general form is obtained:

$$\delta P = \delta\dot{\mathbf{u}}_L^T\{\mathbf{f}_L^{\text{int}} - \mathbf{f}_L^{\text{ext}} + \mathbf{M}_L\ddot{\mathbf{u}}_L + \mathbf{N}_L^T\mathbf{f}_{\Gamma L}\} + \delta\dot{\mathbf{u}}_s^T\{\mathbf{f}_s^{\text{int}} - \mathbf{f}_s^{\text{ext}} + \mathbf{M}_s\ddot{\mathbf{u}}_s + \mathbf{N}_s^T\mathbf{f}_{\Gamma s}\}$$
$$+ \delta\mathbf{f}_{\Gamma L}^T\{\mathbf{N}_L^T\dot{\mathbf{u}}_{\Gamma L} - \mathbf{L}_L^T\dot{\mathbf{u}}_\Gamma\} + \delta\mathbf{f}_{\Gamma s}^T\{\mathbf{N}_s^T\dot{\mathbf{u}}_{\Gamma s} - \mathbf{L}_s^T\dot{\mathbf{u}}_\Gamma\} - \delta\dot{\mathbf{u}}_\Gamma^T\{\mathbf{N}_L^T\mathbf{f}_{\Gamma L} + \mathbf{N}_s^T\mathbf{f}_{\Gamma s}\} \quad (28)$$

where variation in velocity coupling force on the interface $\dot{\mathbf{u}}$ and $\mathbf{f}_{\Gamma i}$ are accounted for. $\mathbf{N}_i^T$ and $\mathbf{L}_i$ are interpolation (or prolongation), and incidence operators (Γ to Ω_i), respectively. To map to the interface we describe this spatial coupling operator $\mathbf{N}_i$ in more detail:

$$\mathbf{N}_i \in \mathbb{R}^{\mathcal{N}_{\Gamma i} \times \mathcal{N}_\Gamma} \quad (29)$$

with dimensions determined by $\mathcal{N}_{\Gamma i}$ and $\mathcal{N}_\Gamma$; as the number of nodes on the interface of the subdomain Ω_i, and the number of nodes on the interface Γ, respectively. Therefore, $\mathbf{N}_\Gamma$ interpolates using Lagrangian shape functions for the two subdomains Ω_L and Ω_s as:

$$\mathbf{N}_\Gamma(\mathbf{x}) = \delta(\mathbf{x}_\Gamma - \mathbf{x}_{\Gamma i}), \text{ for } i = L, s \quad (30)$$

where δ is viewed as a dirac Delta function for coincident nodes. It is convenient to define a restriction operator $\mathbf{R}_i \in \mathbb{R}^{\mathcal{N}_\Gamma \times \mathcal{N}_{\Gamma i}}$ as the transpose of $\mathbf{N}_i$, to map both forces and mass from subdomain interfaces Γ_L and Γ_s onto Γ. The summation on the interfaces now become:

$$\mathbf{M}_\Gamma = \mathbf{R}_L\mathbf{C}_L^T\mathbf{M}_L + \mathbf{R}_s\mathbf{C}_s^T\mathbf{M}_s = \mathbf{R}_L\mathbf{M}_{\Gamma L} + \mathbf{R}_s\mathbf{M}_{\Gamma s} \quad (31)$$

$$\mathbf{f}_\Gamma^{\text{int}} = \mathbf{R}_L\mathbf{C}_L^T\mathbf{f}_L^{\text{int}} + \mathbf{R}_s\mathbf{C}_s^T\mathbf{f}_s^{\text{int}} = \mathbf{R}_L\mathbf{f}_{\Gamma L}^{\text{int}} + \mathbf{R}_s\mathbf{f}_{\Gamma s}^{\text{int}} \quad (32)$$

$$\mathbf{f}_\Gamma^{\text{ext}} = \mathbf{R}_L\mathbf{C}_L^T\mathbf{f}_L^{\text{ext}} + \mathbf{R}_s\mathbf{C}_s^T\mathbf{f}_s^{\text{ext}} = \mathbf{R}_L\mathbf{f}_{\Gamma L}^{\text{ext}} + \mathbf{R}_s\mathbf{f}_{\Gamma s}^{\text{ext}} \quad (33)$$

where we compute mass $\mathbf{M}_\Gamma$, internal force $\mathbf{f}_\Gamma^{\text{int}}$ and external force $\mathbf{f}_\Gamma^{\text{ext}}$ to allow for the explicit computation of $\ddot{\mathbf{u}}_\Gamma$ in Equation 17 to be recalled. These operators are analogous to concepts in multigrid methods and localised Lagrange multipliers (LLMs) [36,37]. Subsequently, we map $\ddot{\mathbf{u}}_\Gamma$ from Γ back to the subdomains:

$$\ddot{\mathbf{u}}_{\Gamma L} = \mathbf{N}_L\ddot{\mathbf{u}}_\Gamma; \quad \ddot{\mathbf{u}}_{\Gamma s} = \mathbf{N}_s\ddot{\mathbf{u}}_\Gamma \quad (34)$$

We illustrate a non-matching mesh in Figure 5 and compute its operators through exemplifying a linear isoparametric mapping in 2-D, where Γ is discretised with line elements to depict the simplicity of this coupling.
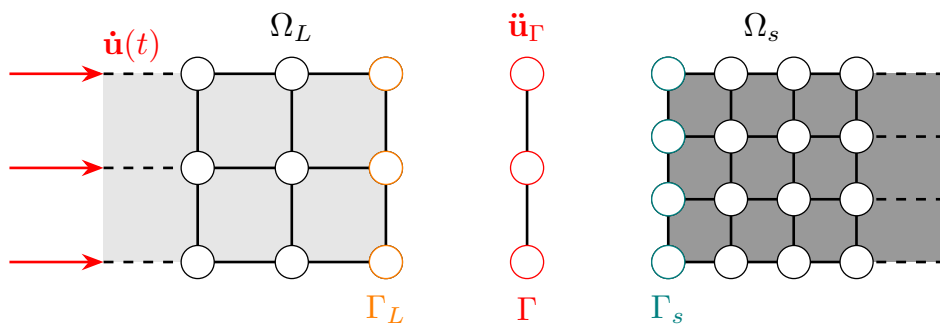

Figure 5. A non-matching benchmark highlighting the interface with differently discretised subdomains partitioned with the Large Γ_L, small Γ_s and externally meshed interface Γ.

For the interpolation matrix, we elucidate that from Γ to Γ_L the mapping is simply one-to-one and $\mathbf{N}_L$ will always take the form of an identity matrix Ω_L, whereas $\mathbf{N}_s$ requires computation of the shape functions:

$$\mathbf{N}_L = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad \mathbf{N}_s = \begin{bmatrix} 1 & 0 & 0 \\ 1/3 & 2/3 & 0 \\ 0 & 2/3 & 1/3 \\ 0 & 0 & 1 \end{bmatrix}; \quad \mathbf{R}_L = \mathbf{N}_L^{\mathrm{T}}; \quad \mathbf{R}_s = \mathbf{N}_s^{\mathrm{T}} \tag{35}$$

where, for this simple case, it can also be shown that the restriction is the transpose of the interpolation. The interface Γ is assumed to share the same geometrical description on both subdomain interfaces Γ_L and Γ_s, without overlap or separation. The spatial and temporal methods are combined to give the following algorithm:

Algorithm 2 Summary of Non-Matching Mesh Algorithm with Multi-Time Stepping

---

1: **procedure** INTEGRATE A TWO-DOMAIN NON-MATCHING MESH WITH MTS
2:    **while** $t^{n+k+1}_{ratio} \leq 1$ or $(t^{n+k}_{ratio} \leq 1$ and $t^{n+k+1}_{ratio} \leq 1 + TOL)$ **do**
3:       Compute $\ddot{\mathbf{u}}_{\Gamma_s}$ with operator $\mathbf{N}_s$ in Eq. 34 on Γ_s
4:       Integrate small domain Ω_s and compute vectors $\mathbf{f}^{\text{int}}_s, \mathbf{f}^{\text{ext}}_s$
5:       Compute trial times $t^{n+k}_{Ts}, t^{N+1}_{TL}$ and time step ratios $t^{n+k}_{ratio}, t^{n+k+1}_{ratio}$
6:    Compute time step reduction factors α_L, α_s
7:    **if** $\alpha_L \geq \alpha_s$ **then**
8:       $\Delta t_L = \alpha_L \cdot \Delta t_L$
9:    **else**
10:       $\Delta t_s = \alpha_s \cdot \Delta t_s$
11:       Compute $\ddot{\mathbf{u}}_{\Gamma_s}$ with operator $\mathbf{N}_s$ in Eq. 34 on Γ_s
12:       Integrate small domain Ω_s and compute vectors $\mathbf{f}^{\text{int}}_s, \mathbf{f}^{\text{ext}}_s$
13:       Recompute trial times $t^{n+k}_{Ts}, t^{N+1}_{TL}$ and time step ratios $t^{n+k}_{ratio}, t^{n+k+1}_{ratio}$
14:    Compute $\ddot{\mathbf{u}}_{\Gamma_L}$ with operator $\mathbf{N}_L$ in Eq. 34 on Γ_L
15:    Integrate large domain Ω_L and compute vectors $\mathbf{f}^{\text{int}}_L, \mathbf{f}^{\text{ext}}_L$
16:    Summate kinetics with $\mathbf{R}_L, \mathbf{R}_s, \mathbf{C}_L, \mathbf{C}_s$ to find $\mathbf{M}_\Gamma, \mathbf{f}^{\text{int}}_\Gamma$ and $\mathbf{f}^{\text{ext}}_\Gamma$ with Eq. 31 - 33
17:    Compute trial times $t^{n+k}_{Ts}, t^{N+1}_{TL}$ and time step ratios $t^{n+k}_{ratio}, t^{n+k+1}_{ratio}$

4.2. Numerical Examples in Space and Time

The following numerical study looks to represent the stress gradients prior to fracture in a compact-tension (CT) specimen test, utilising a similar geometry from literature [38,39]. Figure 6 (a) portrays the geometry modelled in the following example. As the specimen is loaded, stress concentrates about the specimen's crack tip. We apply a ramped velocity $\dot{\mathbf{u}}(t)$ boundary condition on nodes that create a semicircle for upper and lower pins, as shown in Figure 6 (b), with a maximum magnitude of 0.2 ms^{-1}. Whilst equivalent velocities are applied to each of the nodes in the pins, to replicate the contact pressure on the pins, methods such as those applied by Triclot et al. should be considered [40]. Material properties are similar to alumina with $E = 370$ GPa, $\rho = 3.9 \times 10^{-6}$ kgmm^{-3} and Poisson's ratio $\nu = 0.22$. We model the CT specimen with three simulations; one using a fine mesh throughout the entire domain, one coupling a coarse Ω_L and fine Ω_s mesh with a single Δt, and another with Ω_L and Ω_s integrating with multiple time steps Δt_L and Δt_s, respectively. Structured meshes are used in all cases where an element size of 0.58 mm for fine and 1 mm for coarse. All simulations use $Co = 0.5$, running for a maximum $t_{final} = 0.02$ ms.
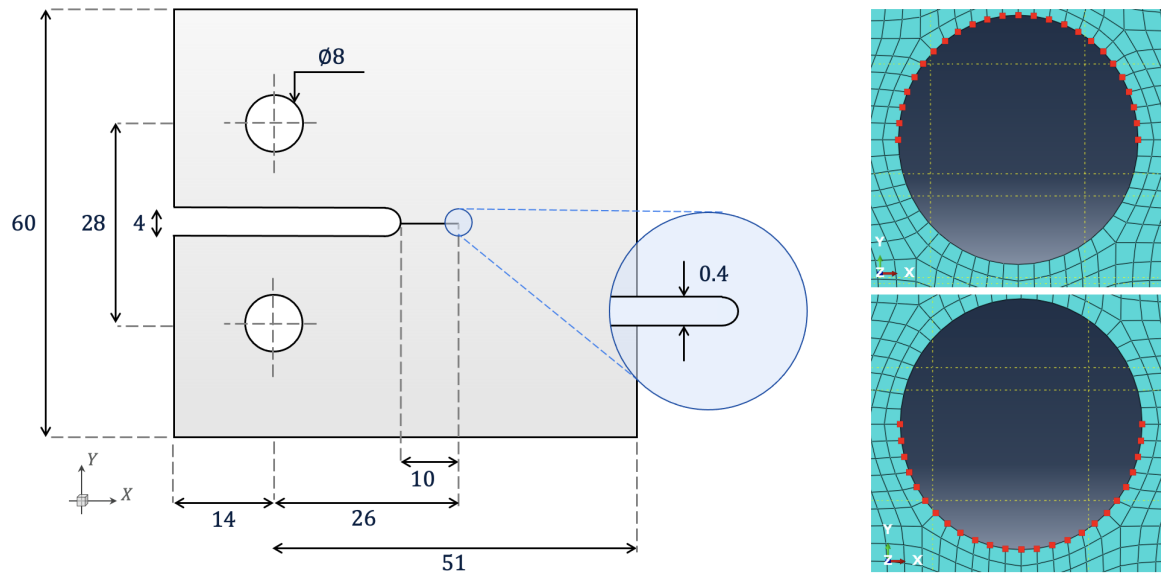
Figure 6. (a) - Diagram of compact-tension specimen with dimensions in [mm], as seen in Sommer et al. [39]; (b) - Nodal sets on the meshed subdomain Ω_L for prescribed velocity boundary conditions on top (+ve loading) and bottom (-ve loading) pins.

In Figure 7 and 8 we plot the stress contours σ_{yy} at $t = 0.01408$ ms and $t = 0.01966$ ms, respectively where the stress concentration can be visualised at the crack tip of the specimen. Figures 7 (a) and 8 (a) capture the reference mesh, whilst Figures 7 (b) and 8 (b) capture the spatially coupled mesh on a single time step Δt_s. Figure 8 accurately predicts maximum stresses of 0.0396 GPa vs 0.0410 GPa for reference against non-matching mesh.
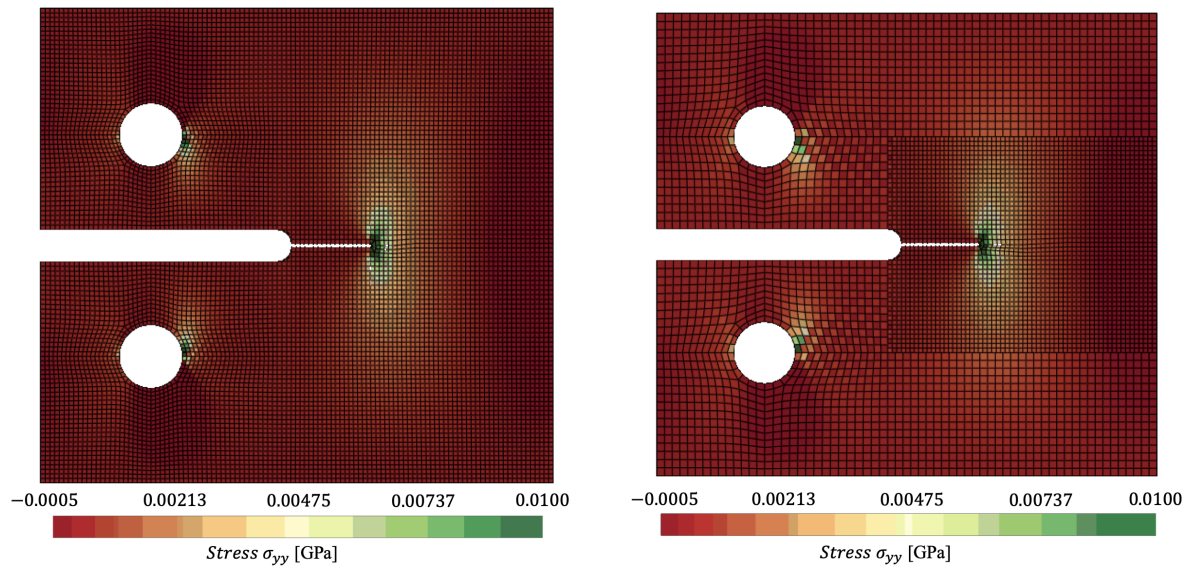

Figure 7. Comparison of σ_{yy} for: (a) - reference (monolithic) versus (b) - spatially coupled dynamically loaded compact-tension specimen, clipping from -0.0005 to 0.01 GPa at $t = 0.01408$ ms.

Figure 8. Comparison of σ_{yy} for: (a) - reference (monolithic) versus (b) - spatially coupled dynamically loaded compact-tension specimen, clipping from -0.0005 to 0.01 GPa at $t = 0.01966$ ms.
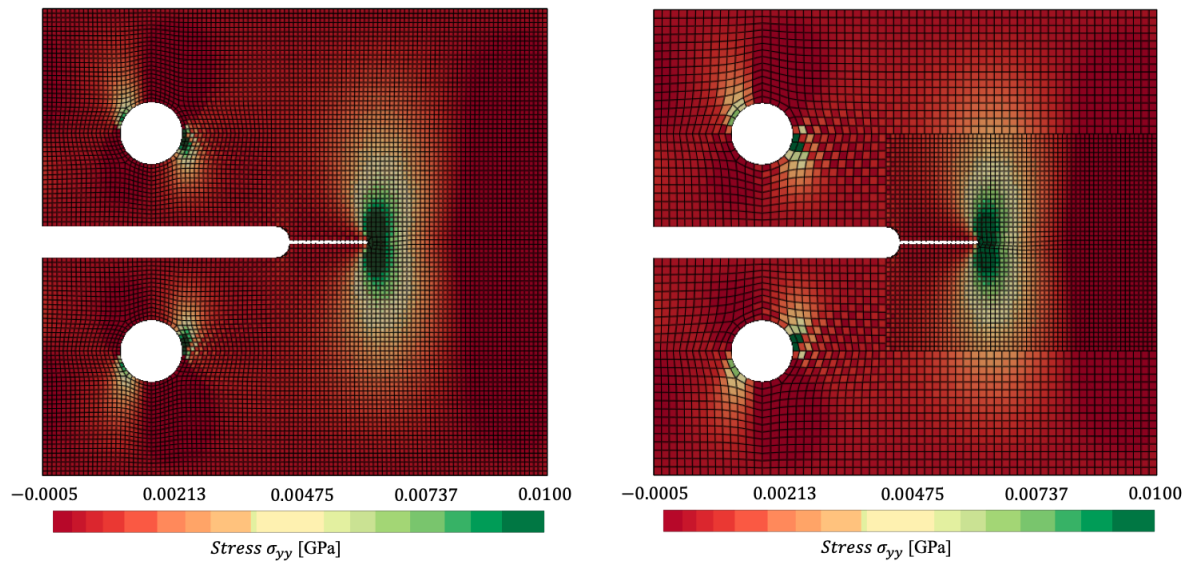
Through combining spatial and temporal coupling, we observe similar ($< 1 \times 10^{-6}$ GPa) results in σ_{yy} distribution, as seen in (a) of Figure 9. The difference in the coupled simulations is captured via the (b) of Figure 9, where the root mean square error (RMSE) is plotted. The stress σ_{yy} is linear interpolated for the non-matching mesh, to allow for comparison of stress on the coarse mesh's Gauss points. The couplings capture nearly the entire specimen within a $< 5\%$ error. Considerable error is located around the circumference of the pins, and the far right of the specimen, however these Gauss points reside far from the area of interest at the crack tip. To avoid such errors, a smaller element size would be required in Ω_L, however this raises the trade-off between accuracy and computational efficiency. Considerable speedup is achieved, with both couplings, as presented in Table 9.
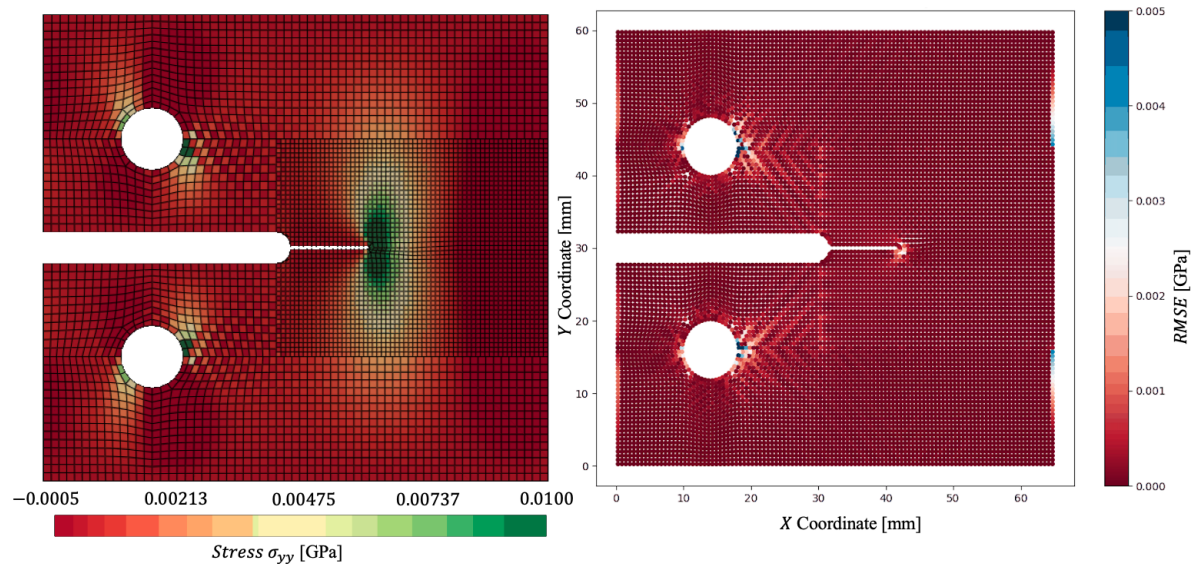

Figure 9. Comparison of σ_{yy} for: (a) - combined spatial and temporal coupling (multi-time stepping); (b) - RMSE Error of σ_{yy} for dynamically loaded compact-tension specimen at $t = 0.01966$ ms.

Table 1. Computational runtime in seconds and speedup vs reference, of the dynamically loaded compact-tension specimen.

Simulation	Runtime [s]	Speedup
Reference (monolithic)	7428	-
Spatially Coupled	2267	$3.27\times$
Spatially and Temporally Coupled	572	$12.98\times$

5. Conclusions

We present coupling methods for the dynamic modelling of solids with explicit finite elements; temporally and spatially. When modelling composites, constituents have varying dilatational wave speeds, hence different time steps. Integrating over the smallest time step can prove highly inefficient, hence the need for multi-time stepping. Our method allows partitions of a domain to solve with their own respective time step, hence reducing computational overhead. Stability of the method is assessed through evaluation of the subdomains' energies. The addition of the coupling in space solves the issue of non-matching meshes, so that small element sizes are only required in regions of interest. Coupling operators are easily implemented, without increasing the degrees of freedom on the interface. Ongoing work addresses the spatial coupling with a quadrilateral non-matching interfaces in 3-D domains [41]. Numerical examples capture an increase in efficiency with stress wave propagation in a heterogeneous bar, and the modelling of a compact-tension specimen. Both couplings in time and space reduce computational runtimes, when compared to their monolithic simulation, especially when combined. Future work looks at the coupling between macro and meso- scale meshes [42–44], with adaptivity a clear necessity for these multi-scale couplings [45]. Other coupling opportunities are plentiful when considering dynamic applications; the modelling of contact [14,16], composite fracture [38,39], fluid-structure interaction [13,27], and other impact engineering scenarios are just a few to mention.

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.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Courant, R., Friedrichs, K. and Lewy, H., On the partial difference equations of mathematical physics. *IBM journal of Research and Development* **1967**, *11(2)*, 215–234.
2. Hughes, T.J. and Liu, W.K., Implicit-explicit finite elements in transient analysis: Implementation and numerical examples. *Journal of Applied Mechanics, Transactions ASME* **1978**, *45(2)*, 375–378.
3. Neal, M.O. and Belytschko, T., Explicit-explicit subcycling with non-integer time step ratios for structural dynamic systems. *Computers & Structures* **1989**, *31(6)*, 871–880.
4. Daniel, W.J.T., Analysis and implementation of a new constant acceleration subcycling algorithm. *International Journal for Numerical Methods in Engineering* **1997**, *40(15)*, 2841–2855.
5. Daniel, W.J.T., A partial velocity approach to subcycling structural dynamics. *Computer methods in applied mechanics and engineering* **2003**, *192(3-4)*, 375–394.
6. Lew, A., Marsden, J.E., Ortiz, M. and West, M., Variational time integrators. *International Journal for Numerical Methods in Engineering* **2004**, *60(1)*, 153–212.
7. Combescure, A. and Gravouil, A., A numerical scheme to couple subdomains with different time-steps for predominantly linear transient analysis. *Computer methods in applied mechanics and engineering* **2002**, *191(11-12)*, 1129–1157.

8. Gravouil, A., Combescure, A. and Brun, M., Heterogeneous asynchronous time integrators for computational structural dynamics. *International Journal for Numerical Methods in Engineering* **2015**, *102(3-4)*, 202–232.

9. Cho, S.S., Kolman, R., González, J.A. and Park, K.C., Explicit multistep time integration for discontinuous elastic stress wave propagation in heterogeneous solids. *International Journal for Numerical Methods in Engineering* **2019**, *118(5)*, 276–302.

10. Dvořák, R., Kolman, R., Mračko, M., Kopačka, J., Fíla, T., Jiroušek, O., Falta, J., Neuhäuserová, M., Rada, V., Adámek, V. and González, J.A., Energy-conserving interface dynamics with asynchronous direct time integration employing arbitrary time steps. *Computer Methods in Applied Mechanics and Engineering* **2023**, *413*, 116110.

11. de Boer, A., van Zuijlen, A.H. and Bijl, H., Review of coupling methods for non-matching meshes. *Computer methods in applied mechanics and engineering* **2007**, *196(8)*, 1515–1525.

12. Hansbo, A. and Hansbo, P., An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer methods in applied mechanics and engineering* **2002**, *191(47-48)*, 5537–5552.

13. Hansbo, A. and Hansbo, P., Nitsche's method for coupling non-matching meshes in fluid-structure vibration problems. *Computational Mechanics* **2002**, *32*, 134–139.

14. Wriggers, P. and Zavarise, G., A formulation for frictionless contact problems using a weak form introduced by Nitsche. *Computational Mechanics* **2017**, *41*, 407–420.

15. Sanders, J.D., Laursen, T.A. and Puso, M.A., A Nitsche embedded mesh method. *Computational Mechanics* **2010**, *49*, 243–257.

16. Puso, M.A. and Laursen, T.A., A mortar segment-to-segment contact method for large deformation solid mechanics. *Computer methods in applied mechanics and engineering* **2004**, *193(6-8)*, 601–629.

17. Faucher, V. and Combescure, A., A time and space mortar method for coupling linear modal subdomains and non-linear subdomains in explicit structural dynamics. *Computer methods in applied mechanics and engineering* **2003**, *192(5-6)*, 509–533.

18. Steinbrecher, I., Mayr, M., Grill, M.J., Kremheller, J., Meier, C. and Popp, A., A mortar-type finite element approach for embedding 1D beams into 3D solid volumes. *Computational Mechanics* **2020**, *66*, 1377–1398.

19. Zhou, M., Zhang, B., Chen, T., Peng, C. and Fang, H., A three-field dual mortar method for elastic problems with nonconforming mesh. *Computer Methods in Applied Mechanics and Engineering* **2020**, *362*, 112870.

20. Wilson, P., Teschemacher, T., Bucher, P. and Wüchner, R., Non-conforming FEM-FEM coupling approaches and their application to dynamic structural analysis. *Engineering Structures* **2021**, *241*, 112342.

21. Singer, V., Teschemacher, T., Larese, A., Wüchner, R. and Bletzinger, K.U., Lagrange multiplier imposition of non-conforming essential boundary conditions in implicit material point method. *Computational Mechanics* **2024**, *73(6)*, 1311–1333.

22. Puso, M.A. and Laursen, T.A., A simple algorithm for localized construction of non-matching structural interfaces. *International Journal for Numerical Methods in Engineering* **2002**, *53(9)*, 2117–2142.

23. Herry, B., Di Valentin, L. and Combescure, A., An approach to the connection between subdomains with non-matching meshes for transient mechanical analysis. *International Journal for Numerical Methods in Engineering* **2002**, *55(8)*, 973–1003.

24. Subber, W. and Matouš, K., Asynchronous space–time algorithm based on a domain decomposition method for structural dynamics problems on non-matching meshes *Computational Mechanics* **2016**, *57*, 211–235.

25. González, J.A., Kolman, R., Cho, S.S., Felippa, C.A. and Park, K.C., Inverse mass matrix via the method of localized Lagrange multipliers. *International Journal for Numerical Methods in Engineering* **2018**, *113(2)*, 277–295.

26. Jeong, G.E., Song, Y.U., Youn, S.K. and Park, K.C., A new approach for nonmatching interface construction by the method of localized Lagrange multipliers. *Computer Methods in Applied Mechanics and Engineering* **2020**, *361*, 112728.

27. González, J.A. and Park, K.C., Three-field partitioned analysis of fluid–structure interaction problems with a consistent interface model. *Computer Methods in Applied Mechanics and Engineering* **2023**, *414*, 116134.

28. Cho, Y.S., Jun, S., Im, S. and Kim, H.G., An improved interface element with variable nodes for non-matching finite element meshes. *Computer methods in applied mechanics and engineering* **2005**, *194(27-29)*, 3022–3046.

29. Kim, H.G., Development of three-dimensional interface elements for coupling of non-matching hexahedral meshes. *Computer methods in applied mechanics and engineering* **2005**, *197(45-48)*, 3870–3882.

30. Bitencourt Jr, L.A., Manzoli, O.L., Prazeres, P.G., Rodrigues, E.A. and Bittencourt, T.N., A coupling technique for non-matching finite element meshes. *Computer Methods in Applied Mechanics and Engineering* **2015**, *290*, 19–44.

31. Rodrigues, E.A., Manzoli, O.L., Bitencourt Jr, L.A., Bittencourt, T.N. and Sánchez, M., An adaptive concurrent multiscale model for concrete based on coupling finite elements. *Computer methods in applied mechanics and engineering* **2018**, *328*, 26–46.

32. Dunne, F., and Petrinic, N., *Introduction to computational plasticity*, OUP Oxford, 2005.

33. de Souza Neto, E.A., Peric, D. and Owen, D.R., *Computational methods for plasticity: theory and applications*, John Wiley & Sons, 2011.

34. Belytschko, T., Liu, W.K., Moran, B. and Elkhodary, K., *Nonlinear finite elements for continua and structures*, John Wiley & Sons, 2014.

35. Chan, K.F., Bombace, N., Sap, D., Wason, D., Falco, S. and Petrinic, N., A Multi-Time Stepping Algorithm for the Modelling of Heterogeneous Structures With Explicit Time Integration. *International Journal for Numerical Methods in Engineering* **2025**, *126(1)*, e7638.

36. Biotteau, E., Gravouil, A., Lubrecht, A.A. and Combescure, A., Multigrid solver with automatic mesh refinement for transient elastoplastic dynamic problems. *International Journal for Numerical Methods in Engineering* **2010**, *84(8)*, 947–971.

37. Dvořák, R., Kolman, R. and González, J.A., On the automatic construction of interface coupling operators for non-matching meshes by optimization methods. *Computer Methods in Applied Mechanics and Engineering* **2024**, *432*, 117336.

38. Pinho, S.T., Robinson, P. and Iannucci, L., Fracture toughness of the tensile and compressive fibre failure modes in laminated composites *Composites Science and Technology* **2006**, *66(13)*, 2069–2079.

39. Sommer, D.E., Thomson, D., Hoffmann, J. and Petrinic, N., Numerical modelling of quasi-static and dynamic compact tension tests for obtaining the translaminar fracture toughness of CFRP. *Composites Science and Technology* **2023**, *237*, 109997.

40. Triclot, J., Corre, T., Gravouil, A. and Lazarus, V., Key role of boundary conditions for the 2D modeling of crack propagation in linear elastic Compact Tension tests. *Engineering Fracture Mechanics* **2023**, *277*, 109012.

41. Sahu, I., Bilinear-Inverse-Mapper: Analytical Solution and Algorithm for Inverse Mapping of Bilinear Interpolation of Quadrilaterals. **2024**, Available at SSRN 4790071.

42. Falco, S., Fogell, N., Iannucci, L., Petrinic, N. and Eakins, D., A method for the generation of 3D representative models of granular based materials. *International Journal for Numerical Methods in Engineering* **2017**, *112(4)*, 338–359.

43. Wason, D., A multi-scale approach to the development of high-rate-based microstructure-aware constitutive models for magnesium alloys. Doctoral dissertation, University of Oxford.

44. Falco, S., Fogell, N., Iannucci, L., Petrinic, N. and Eakins, D., Raster approach to modelling the failure of arbitrarily inclined interfaces with structured meshes. *Computational Mechanics* **2024**, *74*, 805-–818.

45. Bombace, N., Dynamic adaptive concurrent multi-scale simulation of wave propagation in 3D media. Doctoral dissertation, University of Oxford.