Research Article

Partitioned Strong Coupling of Discrete Elements with Large Deformation Structural Finite Elements to Model Impact on Highly Flexible Tension Structures

Klaus Bernd Sautter ⓘ **,**[1] **Tobias Teschemacher,**[1] **Miguel Ángel Celigueta,**[2,3] **Philipp Bucher,**[1] **Kai-Uwe Bletzinger,**[1] **and Roland Wüchner**[1,2]

[1]*Chair of Structural Analysis, Technical University of Munich, Arcisstr. 21, Munich 80333, Germany*
[2]*International Centre for Numerical Methods in Engineering (CIMNE), Barcelona 08034, Spain*
[3]*Universitat Politècnica de Catalunya (UPC), Barcelona 08034, Spain*

Correspondence should be addressed to Klaus Bernd Sautter; klaus.sautter@tum.de

This article presents a staggered approach to couple the interaction of very flexible tension structures with large deformations, described with the finite element method (FEM), and objects undergoing large, complex, and arbitrary motions discretized with particle methods, in this case the discrete element method (DEM). The quantitative solution quality and convergence rate of this partitioned approach is highly time step dependent. Thus, the strong coupling approach is presented here, where the convergence is achieved in an iterative manner within each time step. This approach helps increase the time step size significantly, decreases the overall computational costs, and improves the numerical stability. Moreover, the proposed algorithm enables the application of two independent, standalone codes for simulating DEM and structural FEM as blackbox solvers. Systematic evaluations of the newly proposed iterative coupling scheme with respect to accuracy, robustness, and efficiency as well as cross comparisons between strong and weak FEM-DEM coupling approaches are performed. Additionally, the approach is validated against the rest position of an impacting object, and further examples with objects impacting highly flexible protection structures are presented. Here, the protection nets are described with nonlinear structural finite elements and the impacting objects as DEM elements. To allow the interested reader to independently reproduce the results, detailed code and algorithm descriptions are included in the appendix.

1. Introduction

The simulation of lightweight structures impacted by heavy objects can be a challenging problem if numerical methods are used, as both sides have highly different masses. Some of the applications can be as follows: rockfall in protection net structures [1–3], racecourse protection structures to secure both spectators and drivers at the same time, suicide protection nets on bridges or towers, and many more.

The examples in this work are mainly directed towards natural hazards, such as rockfall events which often cause destruction, especially in mountainous or populated areas.

As it is hard to prevent those events, protection structures are built along settlements and roads. In particular, lightweight flexible structures come into operation, as they are able to undergo large deformations and thus are capable of absorbing large amounts of energy with a smooth and comparably slow load transmission, reducing peak loads and maximum stresses. In principle, flexible structures are built to exploit the possibility to reduce stress peaks by their ability to tolerate large deformations. The interaction between the impacting object and very flexible structures calls for an advanced computational approach, as real scale tests are expensive, complex, and not suitable for a quick assessment of such structures.

The abovementioned problems can be divided into two separate problems: the highly deforming fixed protection structures and the freely moving impact objects. Based upon both solutions, an interaction and equilibrium between both need to be found. The net structures can typically show high deflections. However, the topology usually does not change and thus the net can be described with finite element formulations having a meshed discretization. Furthermore, multiple kinematic formulations can be applied within the analyses. Approaches range, e.g., from shell or membrane structures [2, 3] up to cable or beam models [4] or special formulations, such as ring elements [1]. In the following, the decision was made to use cable net models, discretized with cable element formulations (follow Section 2.2), as those provide accurate results for the given problems with comparatively small computational costs. Additionally, this approach allows contact at the correct position and thus allows small objects to pass through the structure.

Furthermore, the impact objects can also be discretized with different methodologies. The current state-of-the-art rockfall impact simulations is to use rigid bodies to simulate impacting rocks on highly flexible structures [1] where damage and deformation of the impacting objects are neglected. The approaches include finite element methods (FEMs [1]), discrete element methods- (DEM-) described structures, or more flexible structures with material point methods (MPMs [5]), and others. Using the FEM makes it possible to accurately describe the continuum of impacting objects. However, the approach can be very costly, as the contact detection can become very complex. DEM provides much optimized contact detection and thus an efficient multiphysics simulation. The drawback is that the continuous expression is complex and dependent upon difficult parameter calibration. Furthermore, the use of MPM might show similar properties as the FEM with fewer contact difficulties. This method can be quite complex in terms of the setup and smears the contact due to its numerical properties. In the following, DEM is chosen to discretize the impact objects used herein, as an efficient algorithm is needed, and no further attention is focused on its continuum description.

Coupling the DEM and the FEM is a common way to simulate various multiphysics problems [6, 7]. In particular, in problems where the interaction between granular materials or rigid objects with large motions and continuous systems is of interest, this combination of discretization methods is frequently used. Various applications, such as the thermomechanical behaviour [8] of contact between frictional bodies [9, 10], assessment of strains in the simulation of shot peening [11, 12], races and balls in ball bearings [13–15], general tribological systems [16, 17] such as the simulation of rail tracks [18], and more advanced investigations, including fracture due to blast loads [19], are studied using DEM-FEM coupling. The DEM is also used to simulate production processes such as rotating machinery [10] and particulate flows [20].

The coupling between DEM and FEM is done in a partitioned manner, which allows the combination of the respective best-suited solution strategies for each subproblem and the transfer of information in-between.

Accordingly, the user is not restricted to a code which includes both participants but instead can couple any existing DEM and FEM software by creating a suitable interface. This publication concentrates on the discussion of spatial mapping with cable structures. The coupling ultimately also allows the use of blackbox solvers for each participant (e.g., symplectic Euler [21] for the DEM and generalized alpha method [22] for the FEM). In order to advance the state of the art, a strong coupling algorithm is developed for the DEM-FEM impact problems.

To investigate its potential and superior performance in the underlying application case, a weak coupling algorithm is presented and used for comparison. It is known from the field of fluid-structure interaction [23, 24] that a strong coupling algorithm typically allows larger time steps compared to a weak coupling approach. The aim of this work is to investigate the properties of a strong coupling algorithm for DEM-FEM coupling and assess its usability in this setup.

From a formal point of view, the structure of the paper is as follows:

   (i) Section 2 describes the FEM notation including an introduction to the applied cable formulation

  (ii) Section 3 gives an overview of DEM

 (iii) Section 4 introduces the equilibrium required for the coupling and depicts the spatial mapping

 (iv) Section 5 explains the staggered weak coupling approach

  (v) Section 6 depicts the strong coupling approach, which adds additional complexity

 (vi) Section 7 demonstrates the advantages of the proposed coupling algorithm and investigates the influence of a variety of different input parameters

(vii) Sections 7.1 and 7.2 show and investigate the novel coupling approaches

(viii) Sections 7.3 and 7.4 present sensitivity analyses of the important parameters

 (ix) Sections 7.5 and 7.7 demonstrate large-scale problems

  (x) Section 8 gives a conclusion and outlook on future research

 (xi) Appendix A provides a more detailed discussion of the calculation of contact forces for the DEM

(xii) Appendix B provides references to the software used in this study and provides scripts for reproduction of results

2. The Finite Element Method

The finite element method (FEM) [25] is used to numerically solve partial differential equations. For this purpose, a domain is discretized into finite elements for which an approximated solution is known.

As described in [22, 26–28], kinematic relations describe the possible movement of such elements. The current configuration $\mathbf{x}$ can be described with the help of a time-

dependent map (noted as t) from the reference configuration $\mathbf{X}$, which can be obtained from the displacement vector $\mathbf{u}$ (additionally see Figure 1):

$$\mathbf{x}(t) = \mathbf{X} + \mathbf{u}(t), \tag{1}$$

the bold letters indicate tensors and vectors.

2.1. Virtual Work.

The momentum equation can be described by applying the divergence theorem [22, 27, 29, 30]:

$$di\,v(\boldsymbol{\sigma}) + \mathbf{b} = \rho\ddot{\mathbf{u}}, \tag{2}$$

with $\boldsymbol{\sigma}$ being the first Cauchy stress, the body forces $\mathbf{b}$, the acceleration $\ddot{\mathbf{u}}$, and the density ρ. Multiplying the momentum equation (2) with the virtual displacement field $\delta\mathbf{u}$ and integrating over the physical domain Ω leads to the weak form, respectively:

$$\delta W_{\text{int}} - \delta W_{\text{ext}} + \delta W_{\text{kin}} = 0. \tag{3}$$

The three virtual work contributions: the internal work (equations (4), respectively (5)), the external work (equation (6)), and the kinetic work (equation (7)), are obtained by the partial integration:

$$\delta W_{\text{int}} = \int_{\Omega} \boldsymbol{\sigma}: \delta\mathbf{e}\,d\Omega, \tag{4}$$

$$\delta W_{\text{int}} = \int_{\Omega_0} \mathbf{S}: \delta\mathbf{E}\,d\Omega_0, \tag{5}$$

$$\delta W_{\text{ext}} = \int_{\Omega} \mathbf{b} \cdot \delta\mathbf{u}\,d\Omega + \int_{\Gamma_{\sigma}} \boldsymbol{\sigma}\mathbf{n} \cdot \delta\mathbf{u}\,d\Gamma_{\sigma}, \tag{6}$$

$$\delta W_{\text{kin}} = \int_{\Omega} \rho\ddot{\mathbf{u}} \cdot \delta\mathbf{u}\,d\Omega, \tag{7}$$

with the Almansi strain $\mathbf{e}$, the normal direction of Ω, $\mathbf{n}$, and the Neumann boundary Γ_{σ}. From equation (4), one can obtain equation (5) (for detailed information, see [22, 27]), which describes the internal virtual work with the use of the Piola–Kirchhoff 2 (PK2) stress tensor $\mathbf{S}$ and the Green-–Lagrange strain tensor $\mathbf{E}$ within the initial domain Ω_0.

2.2. Cable Element.

With regard to the cable element, a cable or a "truss element formulation" having two nodes ($\mathbf{n}_k$ and $\mathbf{n}_l$) considering geometrical nonlinear behaviour is used. The kinematics is outlined in Figure 1. Each node requires three (or for 2D, two) displacement degrees of freedom (dofs), but no further dofs are required.

As the bases for the structural analysis and to introduce relevant notations, the internal forces and the tangent stiffness matrix are briefly noted.

Due to the one-dimensional nature of the element, only one parametric coordinate ξ [$-1 \le \xi \le +1$] is used to describe the geometry [25] with linear shape functions.

By virtue of equation (5), the internal forces $\mathbf{F}_{\text{int}}$ read for each degree of freedom r as follows:

$$F_{r,\text{int}} = A \int_0^L \left(\mathbf{S} + \mathbf{S}_{\text{pre}}\right): \frac{\partial\mathbf{E}}{\partial u_r}\underbrace{dX}_{(L/2)\cdot d\xi}, \tag{8}$$

for the constant crosssection A, the reference length L, and a given prestress $\mathbf{S}_{\text{pre}}$. The tangent stiffness matrix $\mathbf{K}$ can be expressed as

$$K_{\text{rs}} = A \cdot \frac{L}{2} \int_{-1}^{+1} \left(\frac{\partial\left(\mathbf{S} + \mathbf{S}_{\text{pre}}\right)}{\partial u_s}: \frac{\partial\mathbf{E}}{\partial u_r} + \left(\mathbf{S} + \mathbf{S}_{\text{pre}}\right): \frac{\partial^2\mathbf{E}}{\partial u_r\partial u_s}\right)d\xi,$$

$$K_{\text{rs}} = A \cdot \frac{L}{2} \int_{-1}^{+1} \left(\mathrm{D}: \frac{\partial\mathbf{E}}{\partial u_r}: \frac{\partial\mathbf{E}}{\partial u_s} + \left(\mathbf{S} + \mathbf{S}_{\text{pre}}\right): \frac{\partial^2\mathbf{E}}{\partial u_r\partial u_s}\right)d\xi. \tag{9}$$

The constitutive law used in this work is the St. Venant–Kirchhoff hyperelastic material model [29, 30] described by the strain-energy functional Ψ_{sv}. Its first and second derivatives w.r.t. $\mathbf{E}$ describe the PK2-stress $\mathbf{S}$ and the material tangent modulus D, respectively [30]:

$$\mathbf{S} = \frac{\partial\Psi_{\text{sv}}}{\partial\mathbf{E}},$$

$$\mathrm{D} = \frac{\partial^2\Psi_{\text{sv}}}{\partial\mathbf{E}\partial\mathbf{E}}. \tag{10}$$

The prestress $\mathbf{S}_{\text{pre}}$ is a stress, which is initially applied and added to the internal forces of the structure. It is a second-order tensor; however, for the truss kinematics, only a scalar is required. If a stress state, in equilibrium to the initial prestress, is required within a complex structure, form-finding procedures need to be applied beforehand. In contrast to standard structural analysis, form-finding solves an inverse engineering problem. This inverse problem is to find a geometry which corresponds to an equilibrium configuration, whose stress state is identical to the given prestress. The force density method is one approach used to realize this [24, 31]. The described kinematic relation is capable to model either a cable or a truss formulation. To model a realistic behaviour of cable structures, it needs to be ensured that no compression stresses are carried by the structure. With an additional check if a cable is under compression, stiffness contributions and internal forces are avoided. In the following, using the applied naming, trusses would allow compression stresses, whereas cables do not.

The element mass matrix $\mathbf{M}$ for a dynamic simulation results from the virtual kinetic work as depicted in equation (7). With $\mathbf{N}(\xi)$ being the shape functions and the reference density ρ_0, $\mathbf{M}$ is defined as follows:

$$\mathbf{M} = A \cdot \frac{L}{2} \cdot \rho_0 \int_{-1}^{+1} \mathbf{N}(\xi)^T \mathbf{N}(\xi)d\xi. \tag{11}$$

The damping matrix $\mathbf{C}$ is expressed via the Rayleigh damping as a linear combination of the mass matrix $\mathbf{M}$ and the stiffness matrix $\mathbf{K}$ (for more information, see [32]):

$$\mathbf{C} = \tau\mathbf{M} + \kappa\mathbf{K}, \tag{12}$$
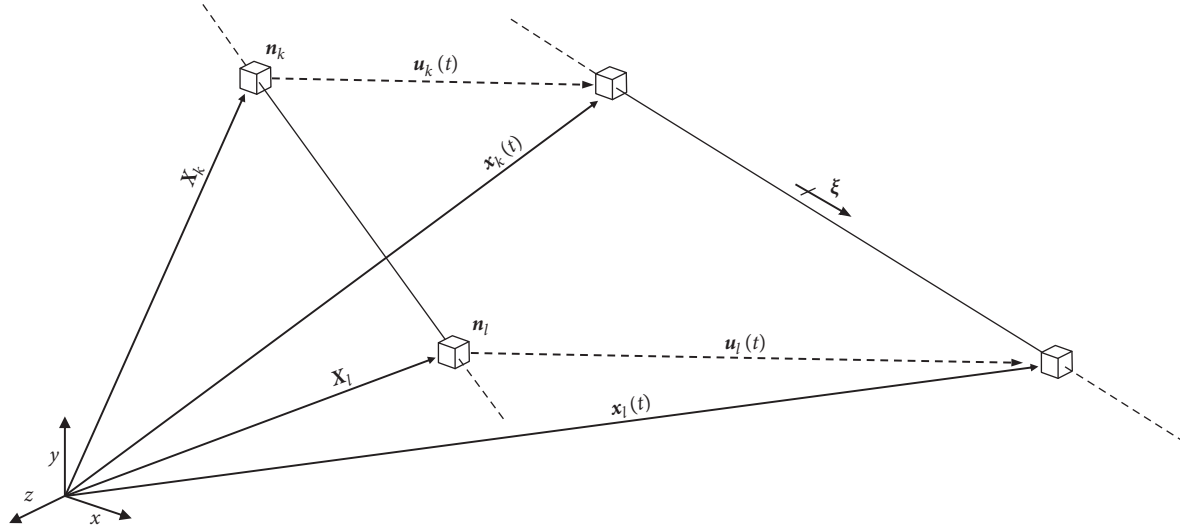
with τ and κ as the scaling factors.

FIGURE 1: Spatial mapping of cable element kinematics between the nodes $\mathbf{n}_k$ and $\mathbf{n}_l$.

3. The Discrete Element Method

This section describes the fundamentals of the discrete element method (DEM), with a focus towards the DEM-FEM coupling (following Sections 4–7).

The DEM used within this scope models the dynamics of spherical particles, considering external forces such as gravity and contact with other particles or contact with differently described objects. DEM can be used for many different particle shapes such as rectangles, cones, spheres, and more. However, only spherical particles are considered in this publication. To describe more complex shapes, a set of spheres is connected within clusters. A cluster consists of multiple particles which are used for contact detection and force evaluation. The mass and centre of gravity are described within the cluster shape and independent of the masses of the particles (for further information, see [33]).

The basic steps in a DEM simulation are as follows (also [21]):

(i) Contact detection

(ii) Evaluation of contact forces

(iii) Integration of motion

These steps will be briefly described in the following.

3.1. Contact Detection.
Within the scope of this publication, certain contact scenarios are of interest. The centre of spheres is described with $\mathbf{C}_i$, and their respective radius is R_i. The distance to contacted edges is deciphered with d_i (Figure 2).

Two spheres $\mathbf{C}_i$ and $\mathbf{C}_j$:

$$\left\| \mathbf{C}_i - \mathbf{C}_j \right\| < R_i + R_j. \tag{13}$$

Sphere $\mathbf{C}_i$ and vertex $\mathbf{n}_k$:

$$\left\| \mathbf{C}_i - \mathbf{n}_k \right\| < R_i. \tag{14}$$

Sphere $\mathbf{C}_i$ and d_i to edge:

$$d_i < R_i. \tag{15}$$

3.2. Evaluation of Contact Forces.
The contact forces can be evaluated using different contact laws and rheological models. For a detailed description of contact models, see [6, 21, 34, 35]. A more detailed description of the evaluation of contact forces is included in Appendix A. The coefficient of restitution (COR), $\varepsilon_\perp$ (see equation (A.7)), is an essential part in the contact and is further investigated in Section 7.4.

3.3. Integration of Motion.
The integration of motion is described by Newton's second law of motion. The translational acceleration $\ddot{\mathbf{u}}$ is described via the force $\mathbf{F}$ and the mass of the particle m. The angular acceleration $\dot{\boldsymbol{\omega}}$ is expressed by the moments $\mathbf{T}$ and the tensor of rotational inertia $\mathbf{I}$ [21]:

$$\begin{aligned} m\ddot{\mathbf{u}} &= \mathbf{F}, \\ \mathbf{I}\dot{\boldsymbol{\omega}} &= \mathbf{T}. \end{aligned} \tag{16}$$

The forces and torques at each particle i are described as follows [6, 19, 21]:

$$\begin{aligned} \mathbf{F}_i &= \mathbf{F}_i^{\text{ext}} + \sum_{j=1}^{n} \overbrace{\mathbf{F}^{ij}}^{\mathbf{F}^{ij}=F_n\mathbf{n}^{ij}+F_t\mathbf{t}^{ij}} + \mathbf{F}_i^{\text{damp}}, \\ \mathbf{T}_i &= \mathbf{T}_i^{\text{ext}} + \sum_{j=1}^{n} \mathbf{r}_c^{ij} \times \mathbf{F}^{ij} + \mathbf{T}_i^{\text{damp}}. \end{aligned} \tag{17}$$

The resulting forces and torques depend on the following components:

$\mathbf{F}_i^{\text{ext}}, \mathbf{T}_i^{\text{ext}}$: external loads (e.g., gravity)

$\mathbf{F}^{ij}$: contact interactions between neighbouring particles or boundaries as the contact $\mathbf{F}_{\text{contact}}$ (see Appendix A for description of the components F_n and F_t)
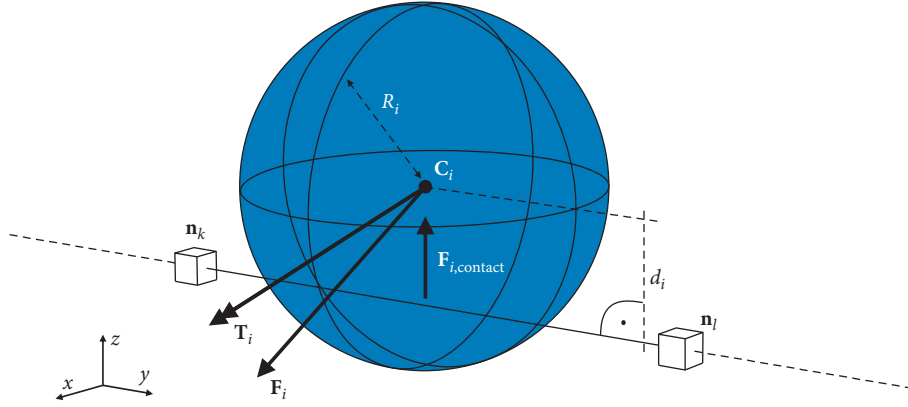
FIGURE 2: Spherical DEM particle i in contact with cable structure between the nodes $\mathbf{n}_k$ and $\mathbf{n}_l$.

$\mathbf{F}_i^{\mathrm{damp}}, \mathbf{T}_i^{\mathrm{damp}}$: external damping boundaries

$\mathbf{n}^{ij}, \mathbf{t}^{ij}$: normal and tangential vectors at the respective contact point

$\mathbf{r}_c^{ij}$: vector connecting the particle centre and the respective contact point

4. Staggered Coupling of DEM and FEM

To couple two standalone physically independent interacting numerical problems such as the simulation of particles and structures, a suitable interface condition to achieve equilibrium is needed to be found.

4.1. Structure-Particle Equilibrium. To put the two independent physics, the DEM and the FEM, into an equilibrium state, the following force equilibrium at the structure needs to be achieved:

$$
\begin{aligned}
&\mathbf{F}_{\mathrm{contact}}\left(\mathbf{u}^{\Omega_{S,\Gamma}}(t),\ \dot{\mathbf{u}}^{\Omega_{S,\Gamma}}(t),\ \mathbf{u}^{P}(t),\ \dot{\mathbf{u}}^{P}(t)\right) \\
&\quad - \mathbf{F}_{int}^{\Omega_{S,\Gamma}}\left(\mathbf{u}^{\Omega_{S,\Gamma}}(t),\ \dot{\mathbf{u}}^{\Omega_{S,\Gamma}}(t),\ \ddot{\mathbf{u}}^{\Omega_{S,\Gamma}}(t)\right) = 0,
\end{aligned}
\tag{18}
$$

where $\mathbf{u}$ describes the displacements, $\dot{\mathbf{u}}$ represents the velocities, Ω_S represents the structural domain, while Γ of Ω_S includes all nodes which are in interaction with the DEM particles, P.

With respect to equation (18), the contact forces $\mathbf{F}_{\mathrm{contact}}$ of the particles, which are dependent on its displacements and the velocities, need to be in equilibrium with the internal forces $\mathbf{F}_{int}^{\Omega_{S,\Gamma}}$ of the structure, which are dependent on its displacements, velocities, and accelerations (equation (8) and additional including inertia effects). The equilibrium of both DEM and FEM simulation is graphically depicted in Figure 3.

The basic idea of the proposed partitioned coupling simulation is the interchange of primary (such as the displacement) and secondary (e.g., forces) interface variables which are obtained as the solution of the respective components of the simulation.

4.2. Spatial Mapping. In the following, the DEM problem is solved independently from the structural problem. To do

this, the displacements and velocities of the structure at the given time step are transferred to DEM, and this structure is further seen as the DEM wall, described by the domain Ω_D. The wall is used to calculate contact forces with the DEM particles P, which are depending on its displacements and velocities (see Appendix A for contact laws and force calculations). After solving the DEM problem, the resulting contact forces are transferred to the structural analysis problem. With the contact forces, seen as external forces, the dynamic structural problem is solved, resulting in new displacements and velocities on the domain Ω_S. This procedure is outlined in Figure 4.

Following this, the contact forces $\mathbf{F}_{\mathrm{contact}}$ are now dependent on the displacements and velocities of Ω_D and not directly on Ω_S and are defined as the external forces coming from the DEM:

$$
\mathbf{F}_{\mathrm{ext}}^{\Omega_{S,\Gamma}} = \mathbf{F}_{\mathrm{contact}}\left(\mathbf{u}^{\Omega_{D,\Gamma}}(t),\ \dot{\mathbf{u}}^{\Omega_{D,\Gamma}}(t),\ \mathbf{u}^{P}(t),\ \dot{\mathbf{u}}^{P}(t)\right).
\tag{19}
$$

The equilibrium within the structural mechanics problem is given as follows:

$$
\mathbf{F}_{\mathrm{ext}}^{\Omega_{S,\Gamma}} - \mathbf{F}_{int}^{\Omega_{S,\Gamma}}\left(\mathbf{u}^{\Omega_{S,\Gamma}}(t),\ \dot{\mathbf{u}}^{\Omega_{S,\Gamma}}(t),\ \ddot{\mathbf{u}}^{\Omega_{S,\Gamma}}(t)\right) = 0.
\tag{20}
$$

After solving both domains, the two interface conditions, for the displacements and the velocities between both fields, are not fulfilled anymore:

$$
\mathbf{u}^{\Omega_{D,\Gamma}}(t) - \mathbf{u}^{\Omega_{S,\Gamma}}(t) = 0,
\tag{21}
$$

$$
\dot{\mathbf{u}}^{\Omega_{D,\Gamma}}(t) - \dot{\mathbf{u}}^{\Omega_{S,\Gamma}}(t) = 0.
\tag{22}
$$

Resulting from this, the contact forces $\mathbf{F}_{\mathrm{contact}}$ computed within Ω_D and the contact forces which would be computed within Ω_S are not the same anymore, and thus, the equilibrium expression is not fulfilled:

$$
\begin{aligned}
&\mathbf{F}_{\mathrm{contact}}\left(\mathbf{u}^{\Omega_{D,\Gamma}}(t),\ \dot{\mathbf{u}}^{\Omega_{D,\Gamma}}(t),\ \mathbf{u}^{P}(t),\ \dot{\mathbf{u}}^{P}(t)\right) \\
&\quad - \mathbf{F}_{\mathrm{contact}}\left(\mathbf{u}^{\Omega_{S,\Gamma}}(t),\ \dot{\mathbf{u}}^{\Omega_{S,\Gamma}}(t),\ \mathbf{u}^{P}(t),\ \dot{\mathbf{u}}^{P}(t)\right) = 0.
\end{aligned}
\tag{23}
$$

For small time steps, resulting into smaller contact forces, the tracking of the interface equilibrium can be negligible. However, for ill-conditioned systems and large time steps, the resulting difference will lead to inaccuracies
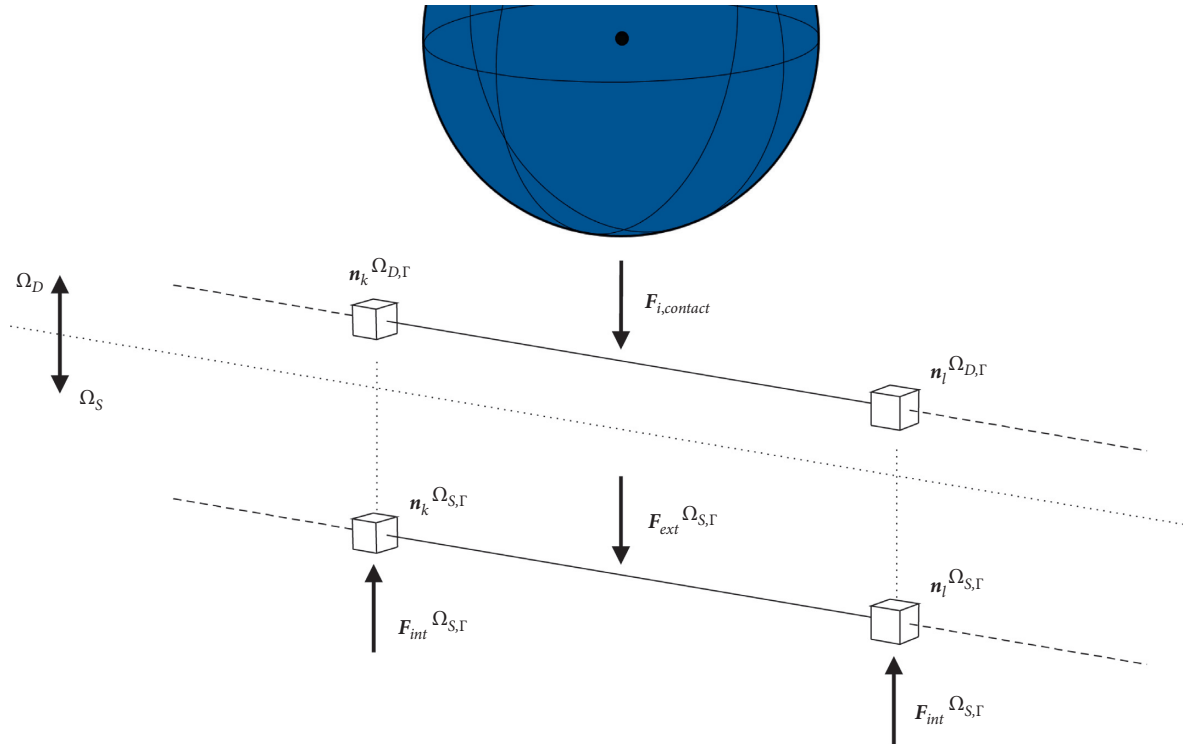
FIGURE 3: Domain definition and force equilibrium in DEM and FEM simulation.
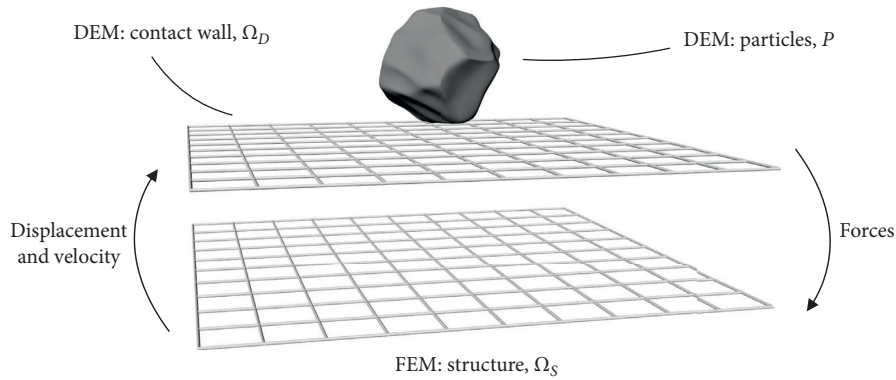

FIGURE 4: Transfer of forces and displacements and velocities between different applications.

and makes the solution unstable. To solve this problem, a possible approach is presented in Section 6.

4.3. Influence of Coefficient of Restitution (COR). Large contact forces will result in difficult fulfilment of interface conditions (equations (21)–(23)). Section 6 proposes a remedy for that problem. One major factor influencing the magnitude of the contact forces is the DEM particle property COR. This value must be defined by the user and heavily influences the stability of the coupled simulation (see example in Section 7.4).

The coefficient represents the ratio of initial speed and final speed after impact [21] (equation (A.6)) and is further discussed in Appendix A. Since this coefficient is determined

manually for each simulation, it is important to be careful when doing the calibration.

4.4. Mesh Dependency for Cable Structures. For the specific application of highly flexible cable structures in this study, such as rockfall protection nets or any other kind of cable-like structures, the DEM wall condition Ω_D discretization and the FEM Ω_S discretization on Γ must exactly coincide (conforming meshes). The respective meshes represent a physical mesh which must be correctly described in order to model the exact contact positions. To demonstrate this behaviour, Figure 5(a) visualizes the DEM part of the simulation. A cable net is modelled and impacted by two spheres. A large sphere finds contact and deforms the boundary, while a smaller sphere penetrates an opening. In
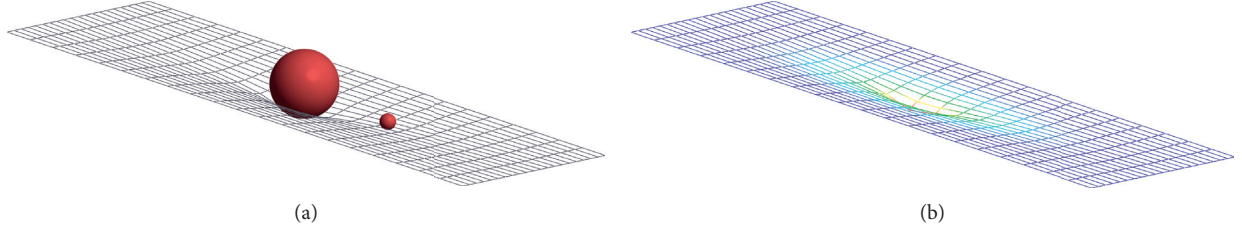
(a)   (b)

FIGURE 5: Modelling of impact with cable-structured boundaries. (a) Discretized Ω_D. (b) Discretized Ω_S.

addition, Figure 5(b) represents the respective FEM structure which is used to calculate the adequate structural response.

If surface elements such as shells or membranes, which do not possess physically the predetermined discrete contact positions, are used within a coupled simulation, arbitrary meshes can be used. In that case, a mapper [36] will be responsible for the correct data transfer.

5. Staggered Weak Coupling

The fundamental idea of the weak coupling (sometimes also called explicit coupling [24]) follows a single exchange of coupling data in each time step. The communication pattern is depicted in Figure 6. The important steps at each time, including this communication pattern, can be summarized as follows:

(1) Solve DEM (results: $\mathbf{u}^P$, $\dot{\mathbf{u}}^P$, and $\mathbf{F}_{contact}$)

(2) Map $\mathbf{F}_{contact}$ from DEM to Structure

(3) Solve Structure (results: $\mathbf{u}^{\Omega_{S,\Gamma}}$, $\dot{\mathbf{u}}^{\Omega_{S,\Gamma}}$, and $\ddot{\mathbf{u}}^{\Omega_{S,\Gamma}}$)

(4) Map *displacements* and *velocities* from Structure to DEM

(5) Advance in time (not explicitly shown)

The interface variables are accordingly updated (see Steps (2) and (4)):

Displacement:

$$^{k+1}\mathbf{u}^{\Omega_{D,\Gamma}} = {}^{k}\mathbf{u}^{\Omega_{S,\Gamma}}. \tag{24}$$

Velocity:

$$^{k+1}\dot{\mathbf{u}}^{\Omega_{D,\Gamma}} = {}^{k}\dot{\mathbf{u}}^{\Omega_{S,\Gamma}}. \tag{25}$$

Contact force:

$$^{k+1}\mathbf{F}_{ext}^{\Omega_{S,\Gamma}} = {}^{k}\mathbf{F}_{contact.} \tag{26}$$

This algorithm is comparatively easy to implement and typically does not require any deep interaction. Standard DEM and FEM simulation environments provide the exchange data as an output. Therefore, different software can

also be efficiently applied here. Furthermore, it was shown that the algorithm can be applied if the time steps do not become too large (see examples in Sections 7.1 and 7.2). However, the behaviour of this procedure can become unstable as soon as the differences in stiffness, mass, and velocity between the two physics become very high. The procedure is then very prone to the time step size used. Decreasing the time step size will lead to inefficient and numerically costly simulations.

To gain a deeper understanding of the underlying procedure, this approach is further detailed in Algorithm 1. In order to facilitate the reproduction of the results, the *Python* script used, including all comments, is provided in Appendix B.

In this procedure, two additional features will be discussed. They are independent of the coupling approach used but improve the performance significantly. They are added in Algorithm 1 and highlighted as follows:

(i) *particle_near_wall* (line 3–6): if the respective particles are in the vicinity of the structural model to adjust the time step is checked. A particle moving freely in space can be simulated with a time step larger than it would be required for the simulation of the DEM-FEM interaction.

(ii) *forces* $\neq 0.0$ (line 12–15): this is an additional check used only to solve the structure when contact forces are present. This is only valid if the self-weight of the structure or any other loads (except impact loads) are neglected. Otherwise, a preliminary simulation or a form finding [24, 31] of the structure is needed.

6. Staggered Strong Coupling

As known from other coupled multiphysics problems, such as fluid-structure interaction (FSI) [24, 37], the direct explicit transfer of the interface data (forces, velocities, and displacements) can lead to divergence problems in the staggered simulation. This problem is caused by large contact forces due to differences in velocities, acceleration, and highly different masses on both sides. In contrast to the weak coupling approach, the strong coupling (in the literature also called implicit coupling [24] or a conventional serial staggered approach within the context of loose coupling [38, 39]) adds an additional iteration loop in each time step, which solves for the equilibrium between both numerical physics. This requires a Gauss–Seidel loop between DEM and FEM, which might need to be solved multiple
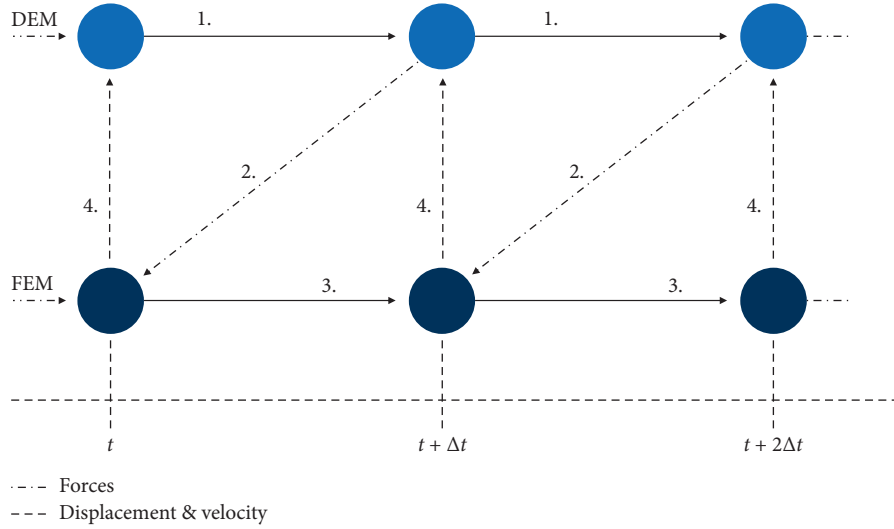
FIGURE 6: Staggered weak coupling procedure, between DEM and FEM.

```
(1)  Initialize
(2)  While time < end_time do
(3)      If particle _ near _ wall then
(4)          use predefined time step
(5)      Else
(6)          use increased time step
(7)      t = t + Δt
(8)      Search nearest neighbours and find contact ⟶ equations (13)–(15)
(9)      Calculate contact forces ⟶ equation (A.5)
(10)     Time integration of DEM part ⟶ equation (16)
(11)     Map forces on Γ from Ω_D to Ω_S
(12)     If forces ≠0.0 then
(13)         Solve structure (FEM)
(14)         Map velocity and displacement on Γ from Ω_S to Ω_D
(15)         Update position of Ω_D to Ω_S ⟶ equation (1)
(16) Finalize
```

ALGORITHM 1: Weak coupling.

times within one time step [23, 24, 38, 39]. This strategy enforces the coupling conditions (equations (4)–(6)) to be fulfilled. Convergence is considered to be achieved, as soon as the interface residual is below a user-defined tolerance ε. The residual formulation is defined by using equation (27)

The steps of this approach are shown in Figure 7 and summarized in the following, using the respective numbering in the abovementioned Figure 7:

(1) Solve DEM (results: $\mathbf{u}^P$, $\dot{\mathbf{u}}^P$, and $\mathbf{F}_{\text{contact}}$)

(2) Map $\mathbf{F}_{\text{contact}}$ from DEM to Structure

(3) Solve Structure (results: $\mathbf{u}^{\Omega_{S,\Gamma}}$, $\dot{\mathbf{u}}^{\Omega_{S,\Gamma}}$, and $\ddot{\mathbf{u}}^{\Omega_{S,\Gamma}}$)

(4) Map displacements and velocities from Structure to DEM

(5) Calculate interface residual ϵ (equation (27))

(6) Repeat Steps 1–5 until the interface residual reaches a given tolerance

(7) Advance in time

The weak coupling algorithm, described in the preceding Section 5 expresses single iteration in the strong coupling scheme (Steps (1)–(4)). The additional interface loop (Step (6), being controlled by the breaking criteria in Step (5)) which adds complexity to the solution procedure and significantly increases the computation costs as the system now needs to be solved multiple times within one time step. However, it allows more accurate results and higher simulation stability. It can be noted that the number of solving iterations is typically still lower than if the time step would be reduced to a value where the weak coupling approach would still be applicable. This is especially due to the property that
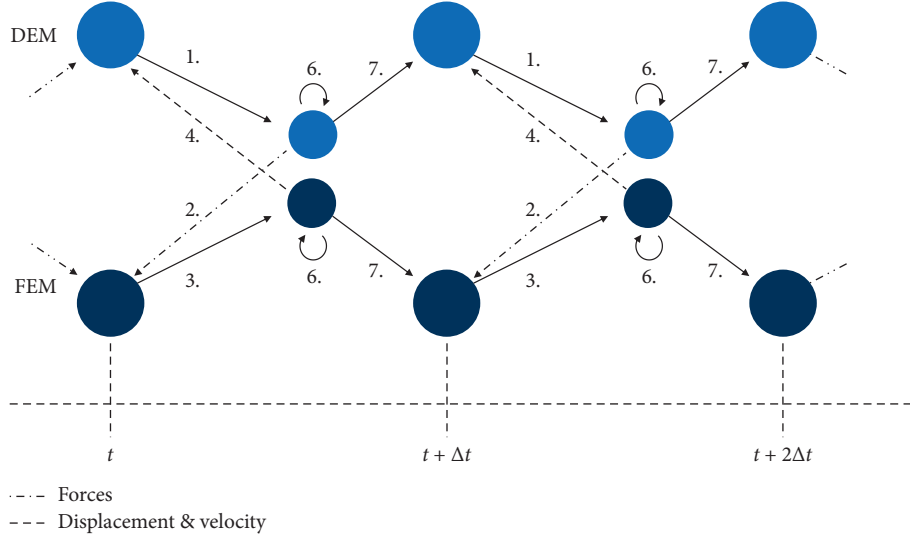
FIGURE 7: Strong coupling communication diagram.

many coupling iterations are typically not required throughout the simulation, but only at specific time steps. The comparison of the two procedures, including a view on the performance, is outlined in Section 7.

The residual criteria within the strong coupling loop are defined by

$$\epsilon < \frac{\|{}^k\mathbf{r}\|}{\sqrt{n_{\mathrm{eq}}}}, \tag{27}$$

where ϵ is the user-defined breaking tolerance. It is checked after each iteration k by scaling the norm of the residuum ${}^k\mathbf{r}$ with the square root of the number of degrees of freedom n_{eq} at the interface Γ [40]. It is important to note that the interface tolerance should be larger than the convergence tolerances of the respective individual solvers within the coupled system; otherwise the convergence criteria cannot be reached. The residuum can either be obtained by the displacements, the velocities, or the contact forces. By subtracting the current solutions on the boundary Γ from the previous solutions of Step $k-1$, the residuum of each variable can be noted as follows:

Displacement residuum:

$$^k\mathbf{r}_u = {}^k\mathbf{u}^{\Omega_{S,\Gamma}} - {}^{k-1}\mathbf{u}^{\Omega_{S,\Gamma}}. \tag{28}$$

Velocity residuum:

$$^k\mathbf{r}_{\dot u} = {}^k\dot{\mathbf{u}}^{\Omega_{S,\Gamma}} - {}^{k-1}\dot{\mathbf{u}}^{\Omega_{S,\Gamma}}. \tag{29}$$

Contact force residuum:

$$^k\mathbf{r}_F = {}^k\mathbf{F}_{\mathrm{contact}} - {}^{k-1}\mathbf{F}_{\mathrm{contact}}. \tag{30}$$

Furthermore, large time steps typically lead to large differences in the interface velocities and displacements, and thus the result can be nonphysical large contact forces. If those forces are too high, small time steps still can lead to unstable simulations, even with the use of the proposed strong coupling algorithm. As a remedy, the transferred data can be gradually applied, which is also called relaxation. The outcome is that this permits a faster interface convergence. The so-called *convergence acceleration* [24] can be achieved by numerous methods and is discussed in the following.

Two different strategies can be chosen for the relaxation: either the relaxation of the displacements and velocities or the relaxation of the contact forces. The relaxation is done w.r.t. the residual (equations (28)–(30)), respectively:

Relaxed displacements:

$$^k\mathbf{u}_{\mathrm{rel}}^{\Omega_{S,\Gamma}} = {}^{k-1}\mathbf{u}_{\mathrm{rel}}^{\Omega_{S,\Gamma}} + {}^k\alpha_u\,{}^k\mathbf{r}_u. \tag{31}$$

Relaxed velocities:

$$^k\dot{\mathbf{u}}_{\mathrm{rel}}^{\Omega_{S,\Gamma}} = {}^{k-1}\dot{\mathbf{u}}_{\mathrm{rel}}^{\Omega_{S,\Gamma}} + {}^k\alpha_{\dot u}\,{}^k\mathbf{r}_{\dot u}. \tag{32}$$

Relaxed contact forces:

$$^k\mathbf{F}_{\mathrm{contact,rel}} = {}^{k-1}\mathbf{F}_{\mathrm{contact,rel}} + {}^k\alpha_F\,{}^k\mathbf{r}_F. \tag{33}$$

Each variable is subsequently updated from the previous solution (Step $k-1$) using the respective interface residuum scaled by relaxation factor α.

There are different approaches to obtain the scaling factor α [39]. The relaxation factor can be set to a user-defined constant value, which is very simple and helps to

improve the quality of the simulation. Another approach is to use the *Aitken* method [41]. The *Aitken* method optimizes α in every iteration w.r.t. the current residuum $^k\mathbf{r}$ and the previous residuum $^{k-1}\mathbf{r}$:

$$^k\alpha(\mathbf{r}) = -{^{k-1}}\alpha \frac{{^{k-1}}\mathbf{r} \cdot ({^k}\mathbf{r} - {^{k-1}}\mathbf{r})}{\|{^k}\mathbf{r} - {^{k-1}}\mathbf{r}\|}, \qquad (34)$$

respectively, $\alpha_u = \alpha(\mathbf{r}_u)$, $\alpha_{\dot{u}} = \alpha(\mathbf{r}_{\dot{u}})$, and $\alpha_F = \alpha(\mathbf{r}_F)$. The influence of the relaxation factor α is studied in the example in Section 7.3.

In this study, either the displacement and the velocity field or the contact forces are independently relaxed and subsequently mapped. However, in the case of displacements and velocities, both residua have to be achieved to ensure that both solution fields still coincide on both sides. Thus, the resulting residuum for both relaxing procedures is given as follows:

Displacement and velocity residua:

$$^k\mathbf{r} = \max({^k}\mathbf{r}_u, {^k}\mathbf{r}_{\dot{u}}). \qquad (35)$$

Contact force residuum:

$$^k\mathbf{r} = {^k}\mathbf{r}_F. \qquad (36)$$

The interface variables are updated accordingly (see Steps 2 and 4 in Figure 7). The following variables are exchanged within the interface:

Without relaxation
Displacements:

$$^{k+1}\mathbf{u}^{\Omega_{D,\Gamma}} = {^k}\mathbf{u}^{\Omega_{S,\Gamma}}. \qquad (37)$$

Velocities:

$$^{k+1}\dot{\mathbf{u}}^{\Omega_{D,\Gamma}} = {^k}\dot{\mathbf{u}}^{\Omega_{S,\Gamma}}. \qquad (38)$$

Contact forces:

$$^{k+1}\mathbf{F}_{\text{ext}}^{\Omega_{S,\Gamma}} = {^k}\mathbf{F}_{\text{contact}}. \qquad (39)$$

With relaxation
Displacements:

$$^{k+1}\mathbf{u}^{\Omega_{D,\Gamma}} = {^k}\mathbf{u}_{\text{rel}}^{\Omega_{S,\Gamma}}. \qquad (40)$$

Velocities:

$$^{k+1}\dot{\mathbf{u}}^{\Omega_{D,\Gamma}} = {^k}\dot{\mathbf{u}}_{\text{rel}}^{\Omega_{S,\Gamma}}. \qquad (41)$$

Contact forces:

$$^{k+1}\mathbf{F}_{\text{ext}}^{\Omega_{S,\Gamma}} = {^k}\mathbf{F}_{\text{contact,rel}}. \qquad (42)$$

In summary, both solution strategies are described within Algorithms 2 and 3 in pseudocode. They are both further elaborated on in Appendix B.3.

7. Systematic Assessment of the DEM-FEM Coupling

This section presents some examples which systematically analyse the difference between the herein introduced coupling approaches and their application within the simulation of relevant industrial applications. The examples show problems of impacting objects on highly flexible lightweight cable structures, such as protection nets. These interaction problems typically have numerical stability issues within the simulations, as the net structures have a low mass, whereas the rocks are typically heavy. This instability leads to the problem that especially when the first impact occurs, the forces might become very large. Thus, due to the different masses, this may lead to convergence problems, especially if the chosen time step is large, which can lead to inaccuracies in the simulation.

In the first academic problem 7.1, a cable structure is modelled to evaluate the influence of different time step values. Section 7.2 subsequently uses a cable structure with a large prestress while also showing the influence of the COR in order to analyse the influence of larger contact forces on the required time step. Section 7.2 investigates the difference between relaxing forces (Algorithm 3) and relaxing displacements and velocities (Algorithm 2). The proper choice of a relaxation factor is further discussed in the example in 7.3. The influence of the COR, which scales the contact forces, is then analysed in Section 7.4. Finally, a practical application of a rockfall into a cable net, using the herein explained approaches, is presented in Section 7.5.

7.1. Impact on a Compliant Cable: Large Deformations. In this example, a single DEM particle with perfect spherical dimensions impacts on a prestressed cable, which is discretized with three finite elements. Here, the contact point on the structure is known, and thus it can be focused on the performance of the coupling algorithms. The setup of this academic example can be found in Figure 8(a), with E as Young's Modulus and Poisson's ratio ν. It demonstrates the necessity of a strong coupling algorithm; since for larger time steps, the phenomena of artificial contact loss, due to large initial contact forces, occur.

Within empirical tests, the time step $\Delta t = 10^{-3}$ s is found to be the highest possible time step for which the weak coupling algorithm can resolve to an appropriate solution.

```
(1) Initialize
(2) While time < end_time do
(3)     While interface _ res ≥ tolerance _ interface do
(4)         Search nearest neighbours and find contact ⟶ equations (13)–(15)
(5)         Calculate contact forces ⟶ equation (A.5)
(6)         Time integration of DEM part ⟶ equation (16)
(7)         Map forces on Γ from Ω_D to Ω_S
(8)         Solve structure (FEM)
(9)         Map velocity and displacement on Γ from Ω_S to Ω_D
(10)        Calculate residuals for velocity and displacement ⟶ equations (28) and (29)
(11)        Relax velocity and displacement ⟶ equations (31) and (32)
(12)        Update position of Ω_D ⟶ equation (1)
(13)        interface_res = max(displacement_residual, velocity_residual)
(14)    Update position of Ω_S ⟶ equation (1)
(15) Finalize
```

ALGORITHM 2: Strong coupling: relaxed displacement and velocity.

```
(1) Initialize
(2) While time < end_time do
(3)     While interface _ res ≥ tolerance _ interface do
(4)         Search nearest neighbours and find contact ⟶ equations (13)–(15)
(5)         Calculate contact forces ⟶ equation (A.5)
(6)         Time integration of DEM part ⟶ equation (16)
(7)         Map forces on Γ from Ω_D to Ω_S
(8)         Calculate residuals for forces ⟶ equation (30)
(9)         Relax forces ⟶ equation (33)
(10)        Solve structure (FEM)
(11)        Map velocity and displacement on Γ from Ω_s to Ω_D
(12)        Update position of Ω_D ⟶ equation (1)
(13)        interface_res = force_residual
(14)    Update position of Ω_S ⟶ equation (1)
(15) Finalize
```

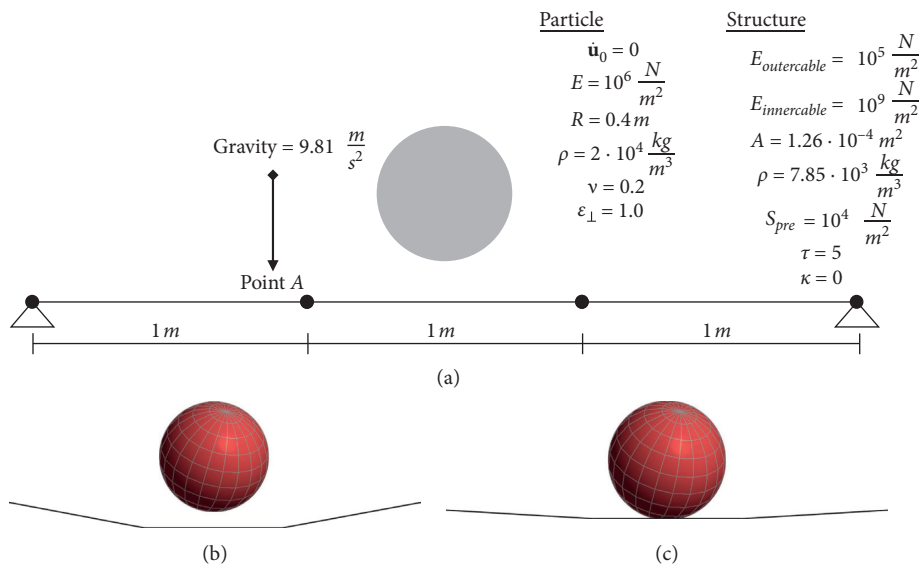ALGORITHM 3: Strong coupling: relaxed forces.

FIGURE 8: Problem setup, including all geometrical and physical input parameters for the simulation. Results after impact with time step of $\Delta t = 10^{-2}$ s. (a) Problem setup. (b) Weak coupling after impact. (c) Strong coupling after impact.

Here, the coefficient of restitution (COR) is set to be 1.0. Implicit time integration is used for the structure, as the chosen time step is too large for an appropriate solution with an explicit time integration scheme. Figures 8(b) and 8(c) show the behaviour of the cable after impact, for the time step size of $\Delta t = 10^{-2}$ s. Using the weak coupling approach, a "jump" can be outlined as shown in Figure 8(b). Due to the large time step, a greater indentation and higher velocities occur. Consequently, the interacting force is too large, so that the sphere and cable do not continuously stay in contact during the entire time. This leads to a nonphysical behaviour of the coupled problem, as shown in Figure 9.

By adding the additional interface loop to solve for the contact force, the convergence of the problem can be achieved for a larger time step of $\Delta t = 10^{-2}$ s.

In the following, the time step of the first contact is discussed in detail. It can be seen (Figure 10(d)) that the contact force is relatively large in the first inner iteration (coming from the relatively large time step) and decreases within the interface iteration to a converged solution, due to the application of the *Aitken* relaxation, introduced in equation (34). This exemplarily demonstrates the advantages of the strong coupling scheme, presented in this article. The large discrepancy in the contact force would lead to an unstable coupled simulation when using a standard weak coupling algorithm. The same accounts for the deflections of the impacting sphere as shown in Figures 10(a)–10(c) presenting a visual description of the interface condition in equations (21) and (22) It can be seen that the positions of Ω_D and Ω_S do converge to a common value to fulfil the interface displacement/velocity equilibrium (equations (21) and (22)).

As an example, the converging contact force for each iteration in time step $t = 1.2$ s and the total number of inner iterations for each time step are shown in Figures 11(a) and 11(b). It shows that the force at the first iteration is almost 4 times higher than it is after the converged solution. Figure 11(b) shows that the number of inner iterations can vary greatly (between 1, if there is no contact, and 9 iterations) within each time step. However, it can be noted that the number of contact simulations is still lower than if the time step would be decreased to $\Delta t = 10^{-3}$ s, which is the limit for which the weak coupling approach still converges.

7.2. Comparison to Position of Rest with Different Time Steps. In this section, a setup similar to the previous example (Section 7.1 and Figure 8(a)) with an increased prestress ($S_{pre} = 10^6$ N/m^2) and a homogeneous Young's Modulus ($E = 10^9$ N/m^2) in the cable structure is used with the following changes for the impacting sphere: $R = 0.12$ m and $\rho = 3.5 \cdot 10^4$ kg/m^3. The result of the transient analysis will be compared to the static solution, considering the particle as an external force. This static force is defined as follows:

$$F_{static} = \overbrace{\frac{4}{3}\pi R_{sphere}^3}^{m_{sphere}} \cdot \rho_{sphere} \cdot \underbrace{9.81\frac{m}{s^2}}_{gravity}. \quad (43)$$

The resulting static deflection of Point A (Figure 8(a)) is shown in Figure 12. This comparison proves that the transient analysis approaches the static solution after a certain time.

Furthermore, the sensitivity of the time step within each coupling algorithm is also studied in this example. The results of all solutions are presented and compared in Figure 12. It shows that the weak coupling approach provides an accurate performance for a time step of $\Delta t = 10^{-3}$ s, whereas the solution for $\Delta t = 10^{-2}$ s is very unstable. It turns out that for large time steps, the result oscillates around the expected solutions.

The measured solutions for time steps of $\Delta t = 10^{-2}$ s and $\Delta t = 3 \cdot 10^{-2}$ s show that the strong coupling algorithm still allows for good convergence for rather large time steps. However, by increasing the time steps, the number of interface iterations subsequently increase, which is shown in Figures 13(a) and 13(b). Especially when the impact occurs, the large difference in interface velocities leads to an increased number of interface iterations (Figure 13(b)). Within the scope of coupled simulations, this difference is decreased by the proposed algorithm leading to a smaller number of iterations.

The influence if either displacements and velocities or forces are relaxed is examined in the following. Both options are described in Algorithms 2 and 3, respectively.

Comparing Figures 13(a) and 13(b), it can be noted that relaxing the forces facilitates slightly faster convergence than relaxing displacements and velocities.

In this specific case at hand, clear and marked off points of load application (impact position) do exist. In different cases, for example in the following Section 7.5, where a variety of possible impact nodes exist, relaxing displacements and velocities are shown to be the better choice. In those cases, which appear more frequently, the impacting spheres can rapidly change the impacting position and thus lead to a slow converging force residual.

7.3. Influence of the Relaxation Factor. In this example, the influence of the relaxation factor α in the case of relaxed displacements and velocities is investigated by comparing the *Aitken* relaxation (equation (34)) and a set of constant relaxation factors. For this purpose, the COR is set to 1.0 (which physically describes a perfectly elastic impact on a rigid obstacle) to neglect the influence of the wall velocity and thus concentrate solely on the relaxation factor. The same problem setup as in Section 7.2 is used.

As Figure 14 shows, a constant relaxation factor can be used as long as it is smaller than 1.0. $\alpha = 1.0$ describes a nonrelaxed system and does not find a proper solution for
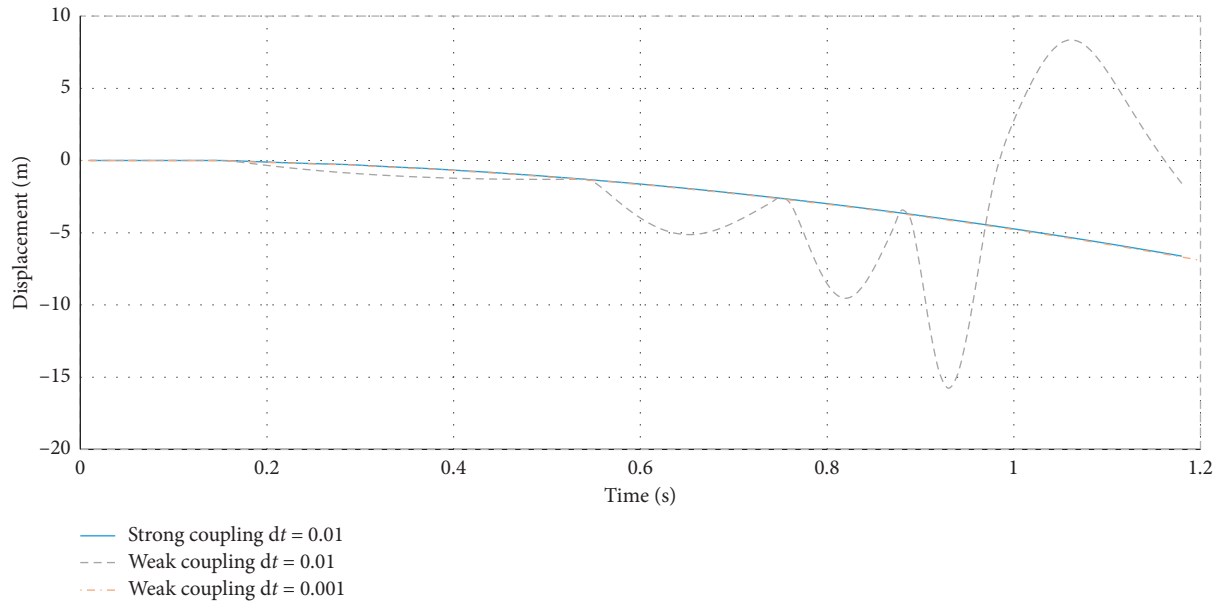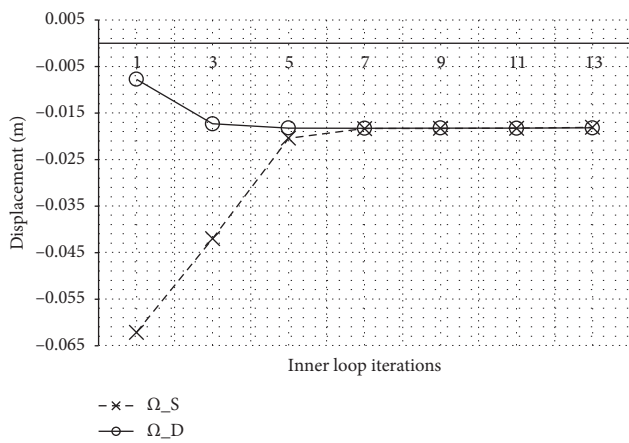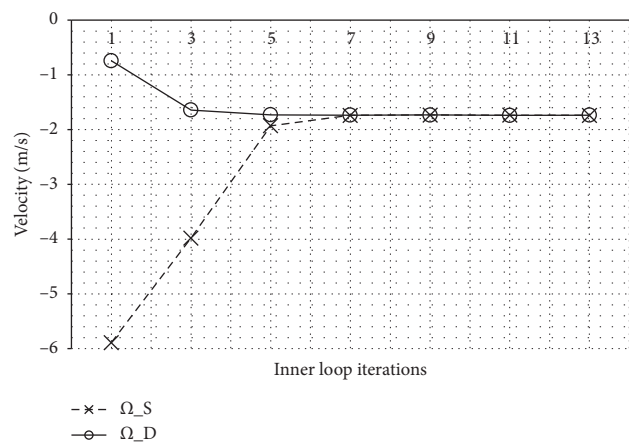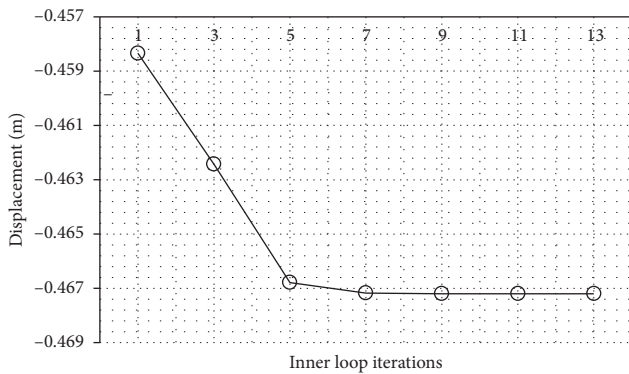
FIGURE 9: Comparison of weak coupling approach with $\Delta t = 10^{-2}$ s and $\Delta t = 10^{-3}$ s against the strong coupling approach with $\Delta t = 10^{-2}$ s for the vertical displacements of node (A) (Figure 8(a)).
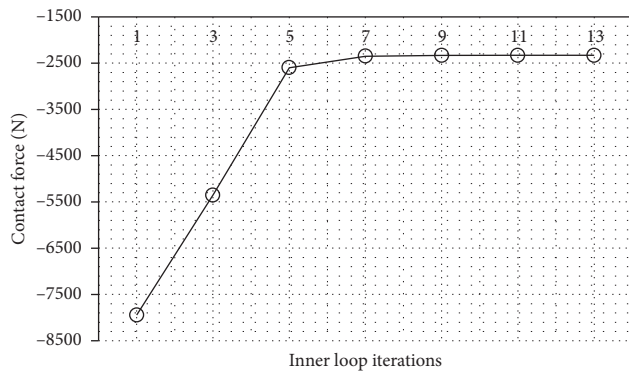

FIGURE 10: Simulation results within one time step over inner coupling iterations. (a) Interface displacement. (b) Interface velocity. (c) Displacement of impacting sphere. (d) Contact forces on Ω_D.
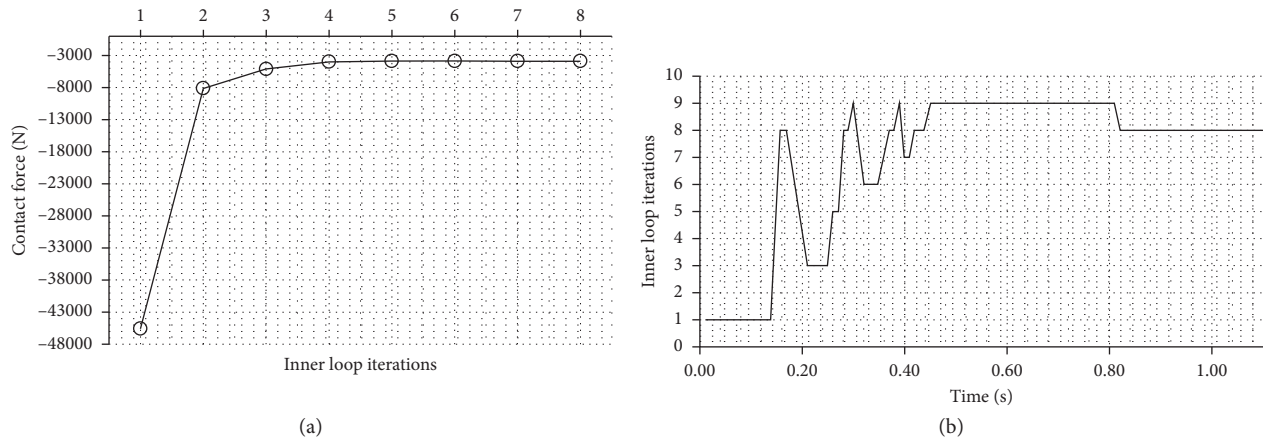
(a)

(b)

FIGURE 11: Interface simulation results. (a) Contact force at $t = 1.2$ s. (b) Number of interface iterations per time step.

—— Weak coupling d$t = 0.001$ - - - Strong coupling d$t = 0.03$
- - - Weak coupling d$t = 0.01$ ·-·-· Static displacement
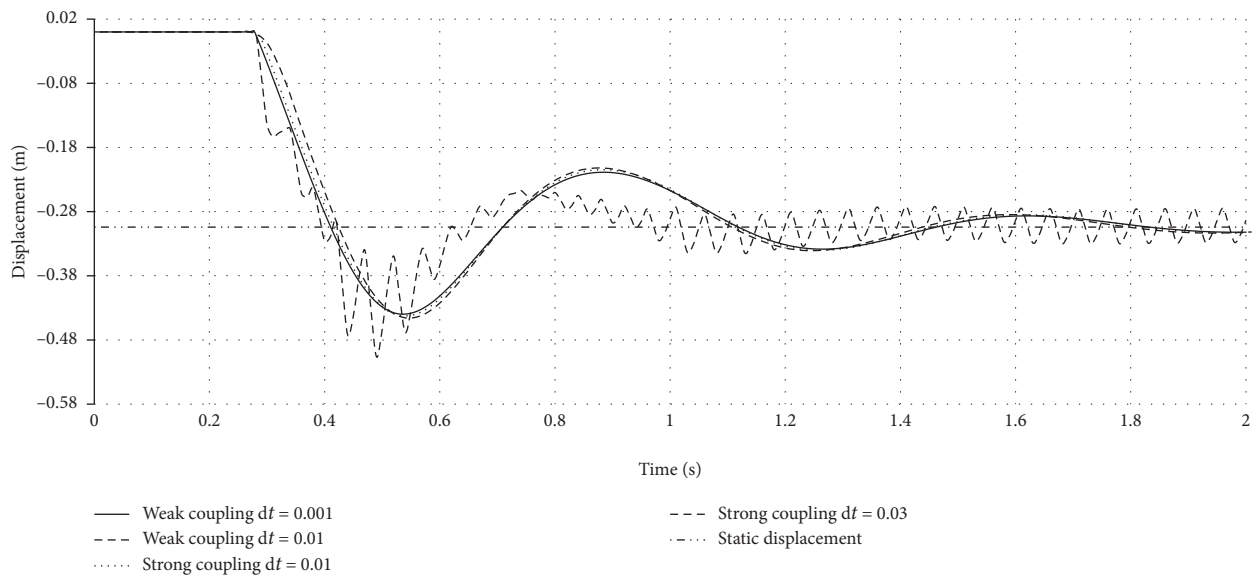········ Strong coupling d$t = 0.01$

FIGURE 12: Comparison of weak coupling approach with $\Delta t = 10^{-2}$ s and $\Delta t = 10^{-3}$ s against the strong coupling approach with $\Delta t = 10^{-2}$ s and $\Delta t = 3 \cdot 10^{-2}$ s for the vertical displacements of node A (Figure 8(a)). Furthermore, the static solution of the problem is provided to show which value the results should approach.

this given example. Manually finding a suitable constant relaxation factor is cumbersome and is dependent on the system setup. In addition, it heavily influences the solving time, as Table 1 demonstrates. For a constant α and the *Aitken* α (equation (34)), the comparison is performed with respect to computation time. It can be noted that although constant relaxation factors provide good results, the optimized *Aitken* relaxation factor facilitates faster convergence to the residual.

7.4. Influence of the Coefficient of Restitution. Another important entity within the multiphysics problem is the COR $\varepsilon_{\perp}$ (Section 3). As Figure 15(b) shows, the COR directly influences the contribution of the DEM rigid wall velocity to the contact force. Current state-of-the-art publications such as [42–44] express the importance of the COR value for

impact simulations. For a case study, different COR values are used, while the time step is kept constant.

As can be seen in Figure 16(a) (zoomed in Figure 16(b)), the interface coupling becomes unstable as soon as the COR reaches a small value. This instability can be overcome by using the strong coupling algorithm presented in Section 6 and is a result of the increased contact force in the system [34]. Additionally, Figure 16(a) describes another important feature: the choice of COR does not influence the final damped solution of the structure (see "static" in the graph in Figure 12) but only the maximum transient solution. Figure 17 visualizes the progression of the maximum interface iterations over the simulation time and indicates the advantages of the proposed coupling algorithm. The large number of iterations at the time of first contact ($t \approx 0.25$ s) calls for a small time step due to the increased difference in interface velocities. This can be overcome with the help of
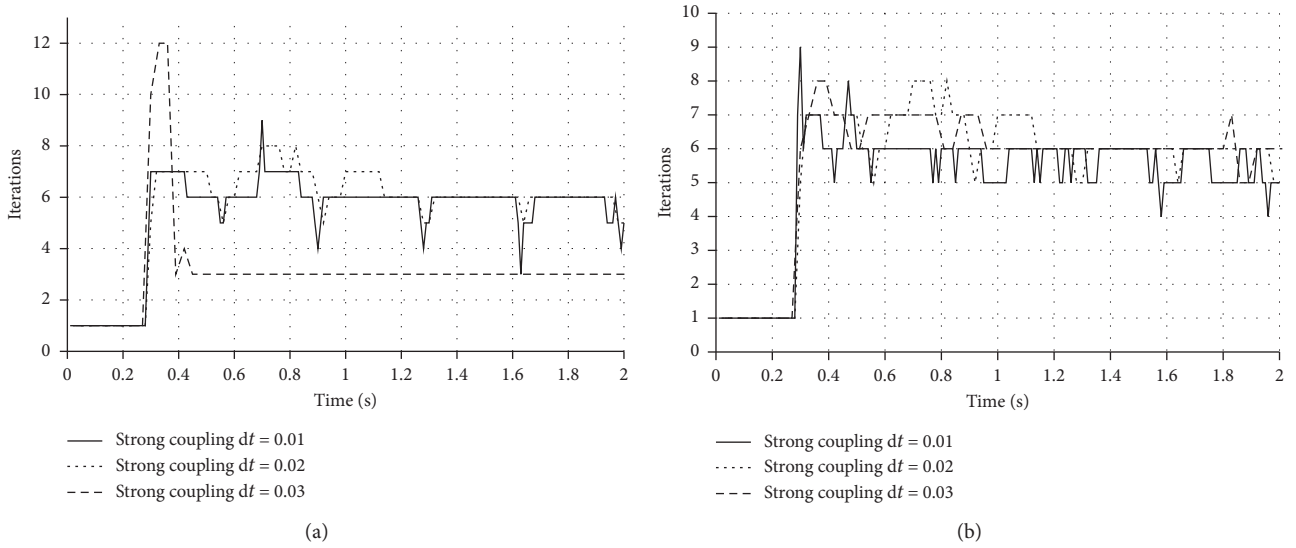
(a)

(b)

FIGURE 13: Number of interface iterations per time step, for either (a) relaxed displacements and velocities or (b) relaxed forces.
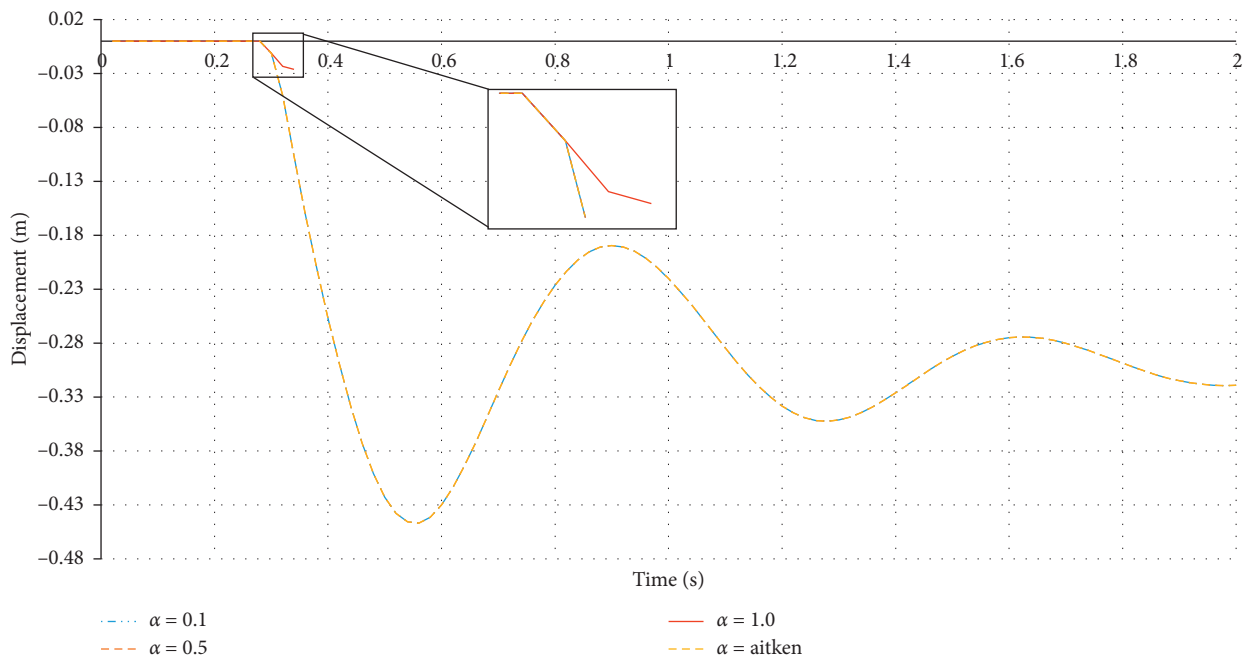

FIGURE 14: Comparison of displacement over time for different relaxation factors α. With $\alpha = 1.0$, the simulation becomes unstable shortly after the first contact is detected. Underrelaxation of $\alpha < 1.0$ leads to stable simulations too. However, this is more computationally costly than using the optimized Aitken relaxation factor.

TABLE 1: Comparison of computation time with different relaxations and the *Aitken* relaxation.

Relaxation factor α	Relative computation time (%)
0.1	100.0
0.2	54.1
0.5	20.4
0.7	28.5
Aitken	18.7

additionally introduced interface iterations. As the simulation proceeds and the initial velocity difference is properly controlled, fewer iterations are needed to enforce the interface conditions.

7.5. Practical Application: Angled Protection Net. One prominent practical application case of highly flexible structures can be found in mountainous regions. As an

(a)

(b)
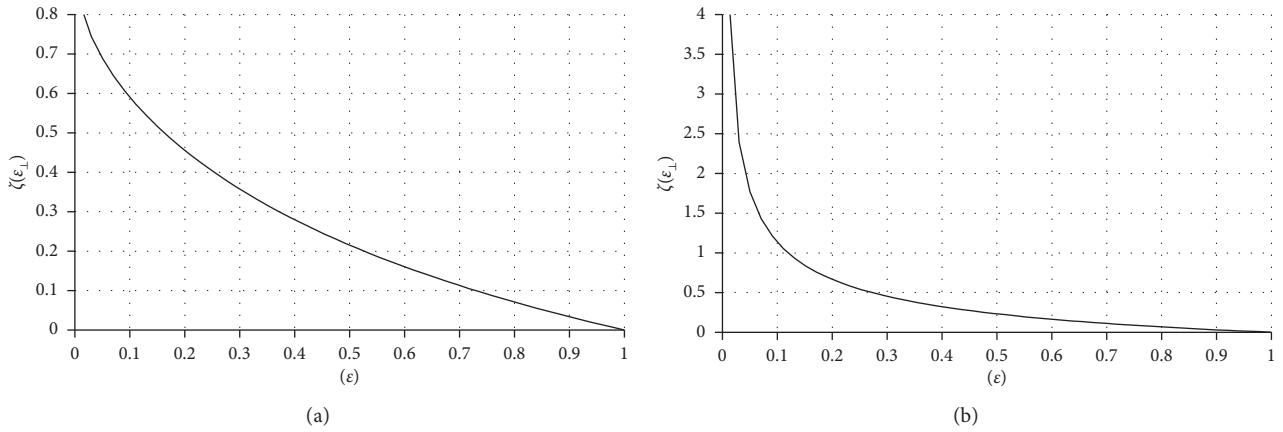
FIGURE 15: Damping coefficient $\zeta(\varepsilon_\perp)$, see [34, 35] for more information. (a) Influence of COR w.r.t. equation (A.7). (b) Influence of COR w.r.t. equation (A.8).

—— Weak coupling $\varepsilon = 0.01$	—— Strong coupling $\varepsilon = 0.01$
—— Weak coupling $\varepsilon = 0.1$	—·— Strong coupling $\varepsilon = 0.1$
—·— Weak coupling $\varepsilon = 0.2$	—··— Strong coupling $\varepsilon = 0.2$
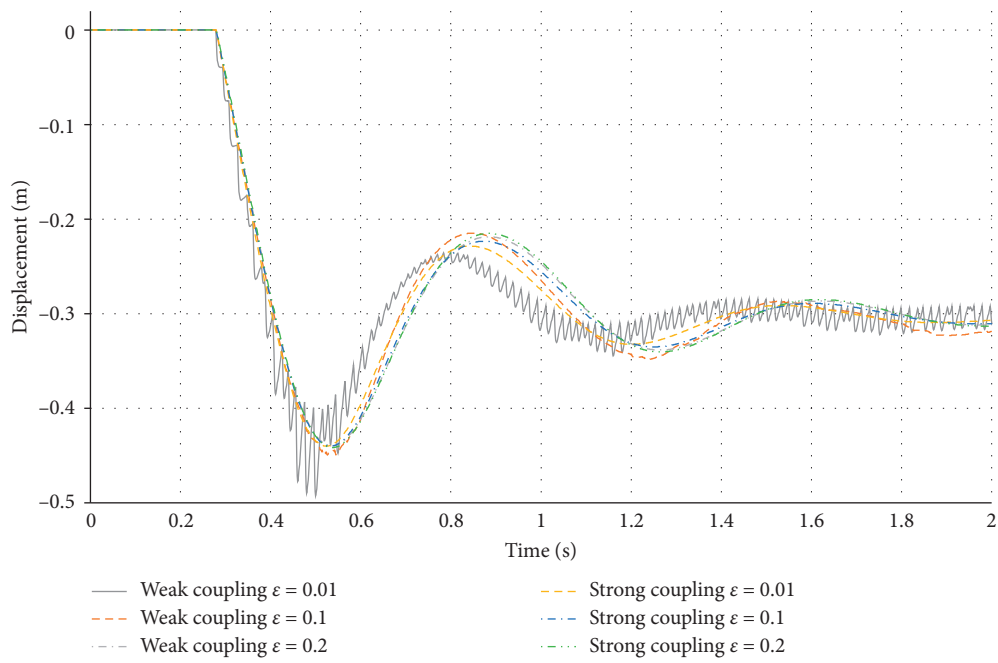
(a)

FIGURE 16: Continued.

FIGURE 16: Comparison of different COR values. (a) $t_0 = 0.0 - t_1 = 2.0$. (b) Scaling to $t_0 = 0.4 - t_1 = 1.4$.

FIGURE 17: Strong coupling interface iterations for $\varepsilon_{\perp} = 0.2$.

alternative to protection nets used to catch rocks, angled nets can also be spanned over roads to direct impacting objects to a safe spot, as shown in Figure 18(a).

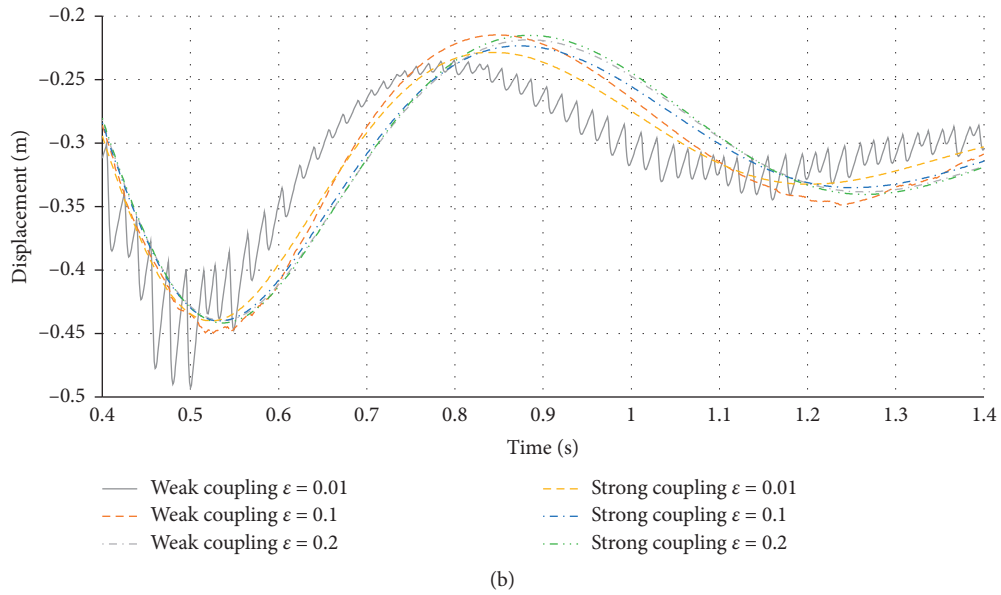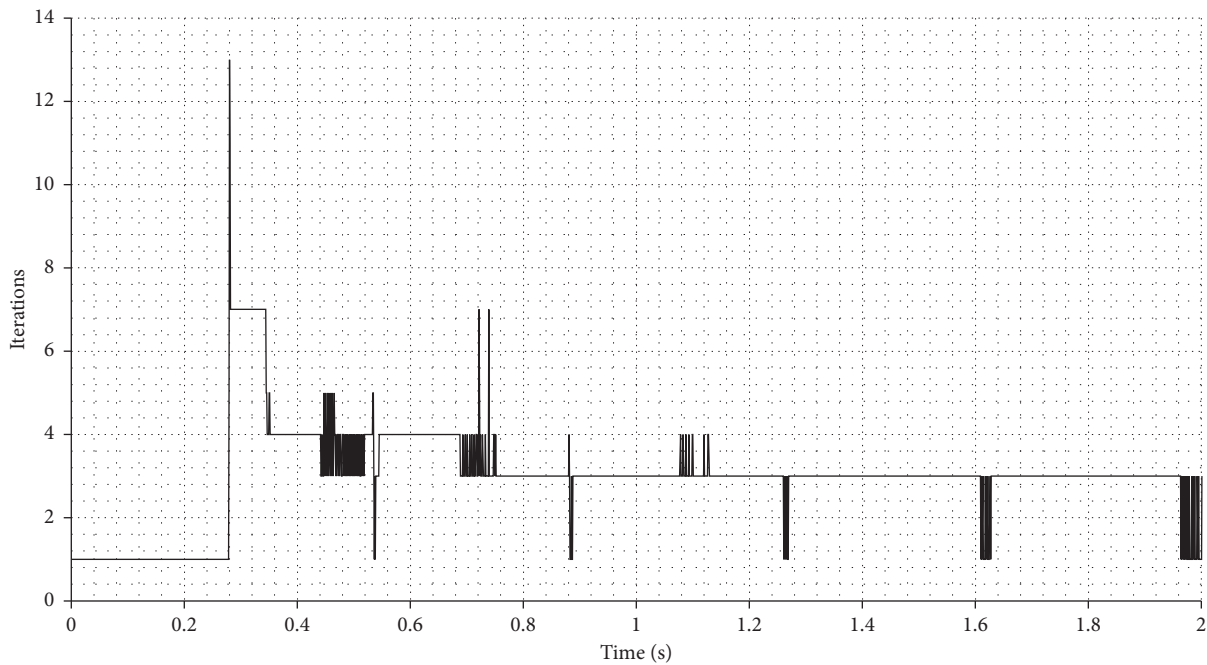To test the limits of the presented algorithms, in this study, the same system as shown in Figure 18(b) is modelled without prestressing the cable structure, leading to a very compliant structure (compare Table 2). Additionally, a small COR of $\varepsilon_{\perp} = 0.2$ and a high impact velocity are chosen to introduce even more difficulties due to an increased contact force.

Using a time step of $2 \cdot 10^{-4}$ s, the different behaviours after impact are presented in the incidental Figures 19(a) and 19(b).

Similar to the example from Section 7.1, the weakly coupled problem experiences too large contact forces and loses contact between the impacting object and the structure, whereas the strong coupling algorithm manages to keep the contact for the given time step (Figure 20(a)).

The considerably large number of interface iterations (Figure 20(b)) is a result of the system setup. This example

(a)   (b)

FIGURE 18: Picture of protection net and corresponding numerical model, including the net as cable structure and two impacting rocks modelled as single DEM particles. The one particle is chosen to be so small that it falls through the net, to verify the correct coupling detection. (a) Route Chalais-Vercorin, Valais [45]. (b) Numerical DEM-FEM coupled model.

TABLE 2: Properties of FEM and DEM parts of example in Section 7.5.

FEM		DEM	
E [N/m^2]	$7 \cdot 10^5$	E [N/m^2]	$1 \cdot 10^5$
A [m^2]	$1.26 \cdot 10^{-5}$	$\dot{u}_0$ [m/s]	$[0.0, -5.54, -5.54]$
ρ [kg/m^3]	$7.85 \cdot 10^3$	ρ [kg/m^3]	$2.5 \cdot 10^3$
S_{pre}	0	R_i [m]	$[0.21, 0.05]$
τ	20	ν	$2 \cdot 10^{-1}$
κ	0	$\varepsilon_\perp$	$2 \cdot 10^{-1}$

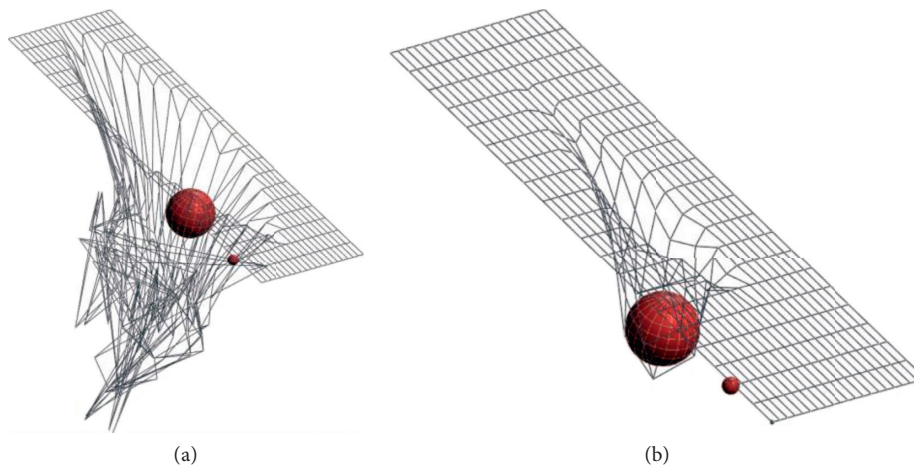(a)   (b)

FIGURE 19: Comparison of the (a) weak coupling to the (b) strong coupling approach with the same time step $\Delta t = 2 \cdot 10^{-4}$ s.

tries to push the time step to a maximum and represents the largest possible time step, which cannot even be used for weak coupling anymore, describing a complex problem.

7.6. *Arbitrary Boundary Conditions.* Another advantage of this procedure is the possibility to use arbitrary boundary conditions for the problem. As can be seen in Figures 21(a)–22(e), the arbitrary triangular and quadrilateral meshes can be used to simulate any boundary condition, for instance, a mountainous region. As it is the state of the art in industrial applications, ∗.*stl* files can be used. If only point clouds are available, standard tools can be used to create a triangulated mesh. The structural part can be subsequently put into this domain to easily capture the interaction of different terrain models and loading scenarios.

As an example, Figures 21(a)–21(i) demonstrate a plane boundary with a curtain-like structure in the middle and a cable net protection net at the end of the slope.

In contrast to the plane boundary, Figures 22(a)–22(e) show the use of an arbitrarily shaped boundary, obtained from a ∗.*stl* mesh. The effortless integration of a deformable FEM structure into the arbitrary boundary is indicated in this example.
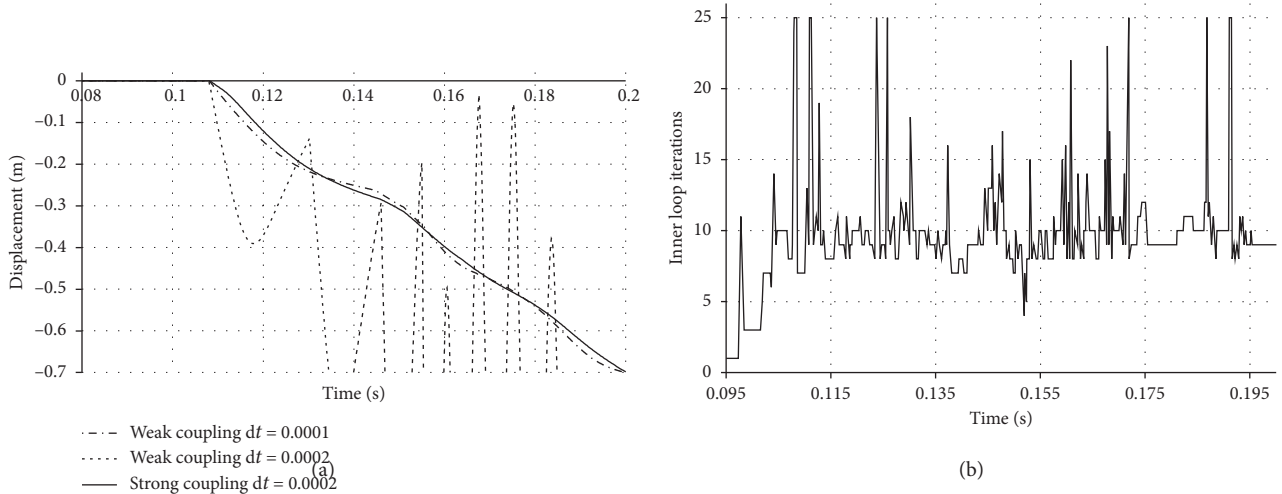
FIGURE 20: Simulation results. (a) Weak vs. strong coupling-displacement of centre node. (b) Interface iterations over simulation time.
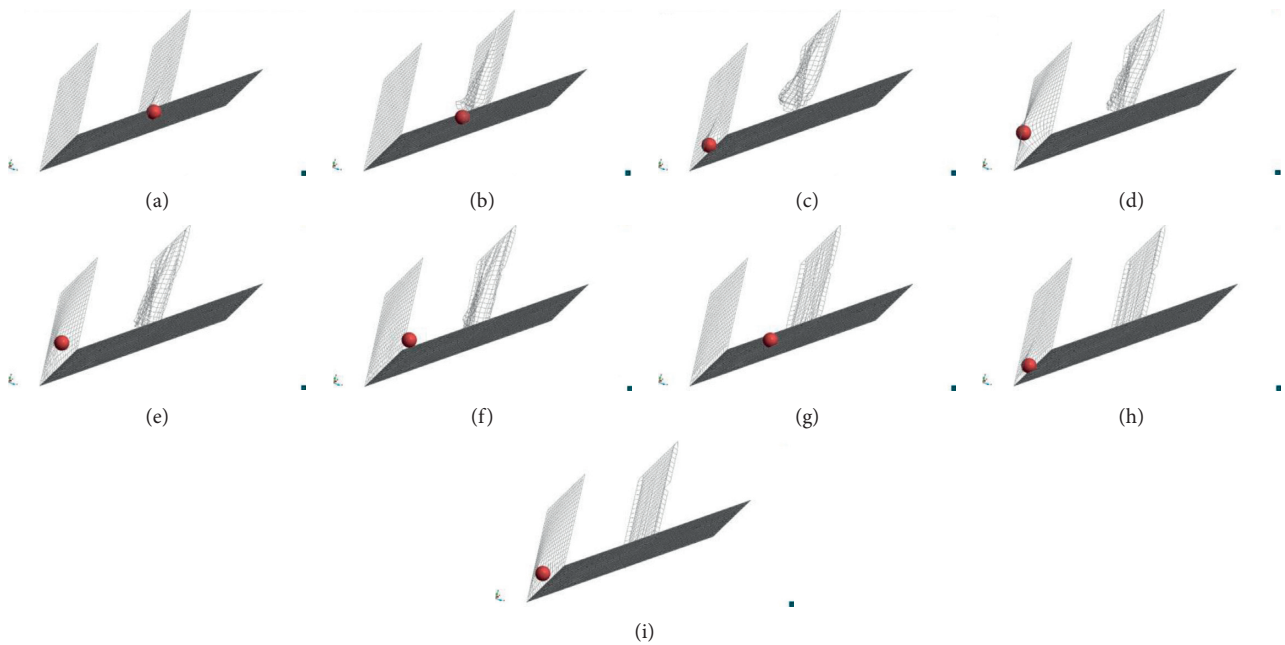
FIGURE 21: Plane boundary rigid with one impacting sphere and two protection nets. The first net (right) is not connected on the lower edge, and the other is coupled to the ground. (a) $t = 1.3$ s. (b) $t = 1.5$ s. (c) $t = 1.9$ s. (d) $t = 2.2$ s. (e) $t = 2.4$ s. (f) $t = 2.5$ s. (g) $t = 3.6$ s. (h) $t = 4.4$ s. (i) $t = 4.7$ s.
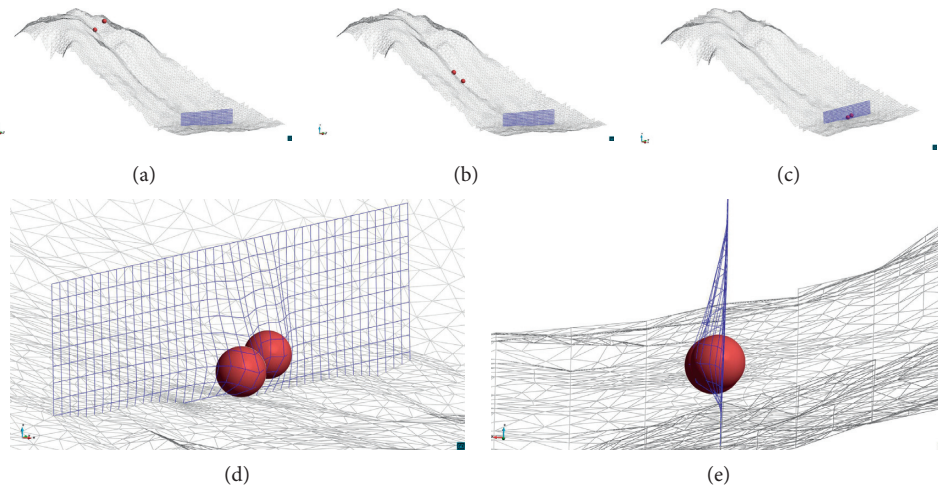
FIGURE 22: Triangulated terrain model, showing the downward rolling of the rocks, which are being slowed and stopped by the protection net. (a) $t = 1.2$ s. (b) $t = 4.5$ s. (c) $t = 6.8$ s. (d) Net impact $t = 6.8$ s. (e) Net impact: side view $t = 6.8$ s.
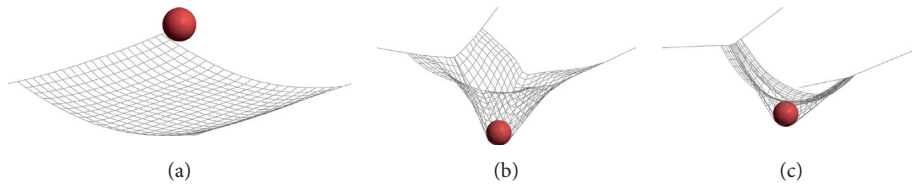
FIGURE 23: Sliding node within the FEM model. It allows us to model a more accurate behaviour of the cable structure nets. (a) $t = 0.0$ s. (b) $t = 2.7$ s. (c) $t = 4.5$ s.
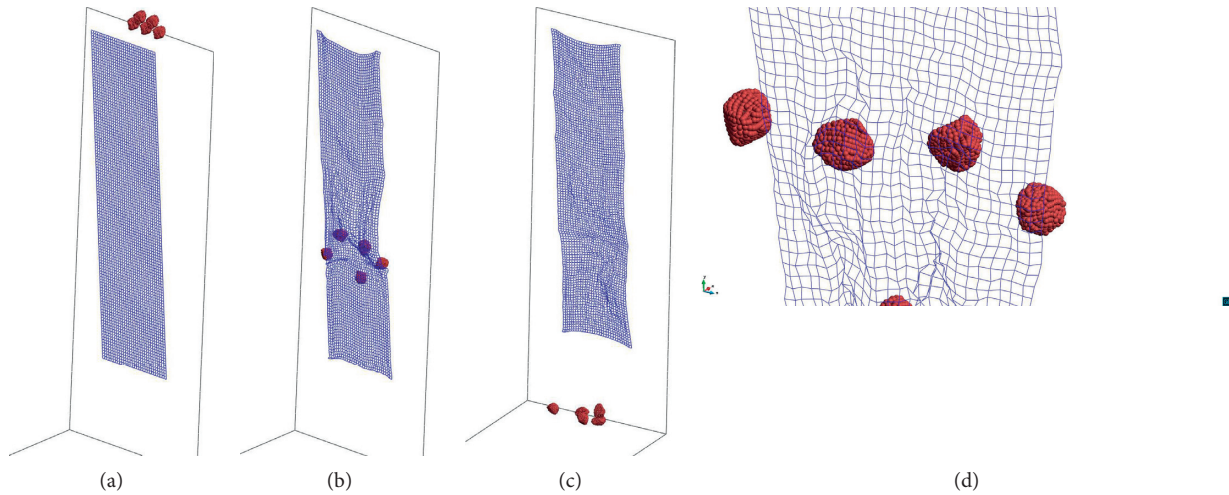
FIGURE 24: Impact problem with more complex shape of impacting objects. Moving impact rocks are discretized as particles which are clustered to allow any arbitrary shapes. (a) $t = 0.0$ s. (b) $t = 1.3$ s. (c) $t = 3.0$ s. (d) Impact of DEM clusters, $t = 1.3$ s.
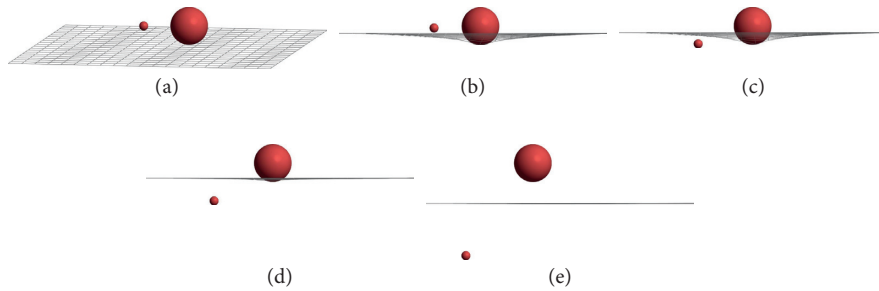
FIGURE 25: Small particle is penetrating and falling through the coarse cable net. Big particle collides with cable structure and is stopped and thrown back by the protection structure. (a) $t = 0.0$ s. (b) $t = 0.05$ s. (c) $t = 0.1$ s. (d) $t = 0.15$ s. (e) $t = 0.2$ s.
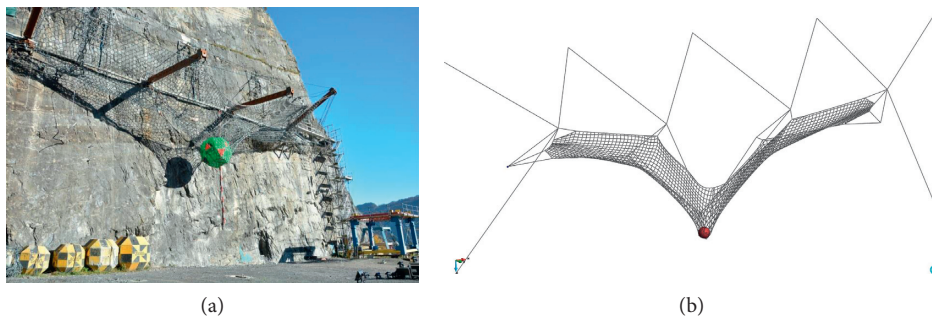
FIGURE 26: Protection net test setup in Walenstadt, Switzerland. (a) Testing area in Walenstadt, St. Gallen, Switzerland [45]. (b) FEM mesh.

7.7. Special Modelling Possibilities. Using two standalone solution techniques, such as the DEM and structural mechanics FEM, enables the user to benefit from the full range of capabilities and strengths of both participants, such as sliding nodes on cable elements (including friction) [1, 4] (Figures 23(a)–23(c)), custom ring elements [1, 4, 46], plasticity laws to model energy dissipation elements, choice of multiple time integration schemes, and clusters of particles (Figures 24(a)–24(d)) to model arbitrarily shaped objects such as rocks, which is an advantage over state-of-the-art rockfall protection simulations as discussed in [1, 47].

Using rigid bodies to model impacting objects has the disadvantage of neglecting damage and deformation of the object itself and adding additional complexity when handling arbitrarily shaped objects. The DEM offers the possibility to simulate breakup of impacting objects [21]; however, the simulation of the continuum with DEM particles can be very costly, and calibration can be elaborate.

The possibility to model Ω_D with line wall conditions suitable to the FEM mesh additionally allows for small rocks to penetrate the protection net (Figures 25(a)–25(e)). In summary, the combination of the DEM and the FEM allows the user to model any possible object impacting in a highly flexible structure modelled by any suitable structural finite element types. Any conceivable combination (Figure 26) is possible as long as a suitable algorithm, as presented in this paper, is available.

8. Conclusions and Outlook

The numerical analysis of lightweight structures coupled with impacting heavy objects proves to be a complex problem and leads to instabilities within the simulation, especially due to the different masses of the participants. To overcome this problem, this publication presents several staggered coupling approaches and presents a sensitivity study with respect to certain crucial parameters.

The procedure suggested herein uses FEM with cable element formulations for flexible lightweight structures (Section 2) and DEM for the interacting objects (Section 3). Furthermore, Section 4 shows the procedure for reaching the equilibrium between both physics. First, the procedure is explained with a single interface calculation within each time step (Section 5). In many examples, this approach proved to be unstable, specifically at initial contacts (indicating large velocity differences). Additionally, the simulation needs small time steps, which might be required only at certain steps. Thus, to overcome this problem, an additional iteration between the physics was explained in Section 6. This allows time steps to be increased significantly and improves the efficiency of the simulation (see examples in 7.1, 7.2, and 7.3). Additionally, the sensitivity of the quality of the simulation is tested by varying the relaxation factor (equation (34) in combination with example in 7.3) and the coefficient of restitution (COR) (see Appendix A in combination with example in 7.4). While the underlying algorithms are abstractly presented in the preceding sections, more detailed versions can be found in the following appendix to allow the interested reader to independently reproduce the results.

The novel approaches make it possible to efficiently simulate the correct behaviour of complex existing structures. The example in 7.5 shows net structures interacted with rocks which are based on existing structures in the Austrian and Swiss Alps. The stability is heavily influenced by a restricting time step (Figures 19(a) and 20(a)) if the interface is not controlled by a suitable algorithm as presented in this study.

In addition, the use of two standalone applications in this study, the so-called "blackbox solvers" allows for a variety of advantageous features. As described in Section 7.6, any given terrain model can be integrated into the simulation process to efficiently capture environmental influences on the results (Figures 22(a)–22(e)). Furthermore, Section 7.7 demonstrates the advantages of an independent FEM application which is capable of modelling numerous structural details, such as energy dissipation elements or sliding nodes on cable elements. Accordingly, DEM can be used to model arbitrarily shaped impacting objects (Figures 24(a)–24(d)). This allows for independent work in the respective application without changing the coupling strategy, which especially proves beneficial in an open-source software environment such as *KRATOS* [48].

In future research, different FEM formulations [2, 3] can be tested for the simulation of the protection nets. Furthermore, if rocks cannot be explicitly described, other particle approaches such as the material point method (MPM [5]) could be applied with the proposed coupling approach. By the way of example, conceivable application cases include the simulation of mud-flow barriers as well as avalanche barriers. Furthermore, the influence of the time integration scheme is additionally a significant factor which will require deeper investigations.

Abbreviations

FEM: Finite element method
DEM: Discrete element method
COR: Coefficient of restitution
MPM: Material point method
FSI: Fluid-structure interaction.

Appendix

A. DEM Force Derivation

A detailed description of the evaluation of forces described in Section 3 is provided in order to further discuss the necessary quantities in the underlying coupling scheme. As soon as a contact is detected, the forces can be evaluated using various contact laws and rheological models in which the normal indentation δ_n [21] is as follows:

$$\delta_n = R_i + R_j - \left(\mathbf{C}_j - \mathbf{C}_i\right) \cdot \mathbf{n}^{ij}, \quad \text{with } \mathbf{n}^{ij} = \frac{\mathbf{C}_j - \mathbf{C}_i}{\left\|\mathbf{C}_j - \mathbf{C}_i\right\|}, \tag{A.1}$$

and its time derivative $\dot{\delta}_n$, the normal vector $\mathbf{n}^{ij}$, and the increment of tangential displacement Δs are used.

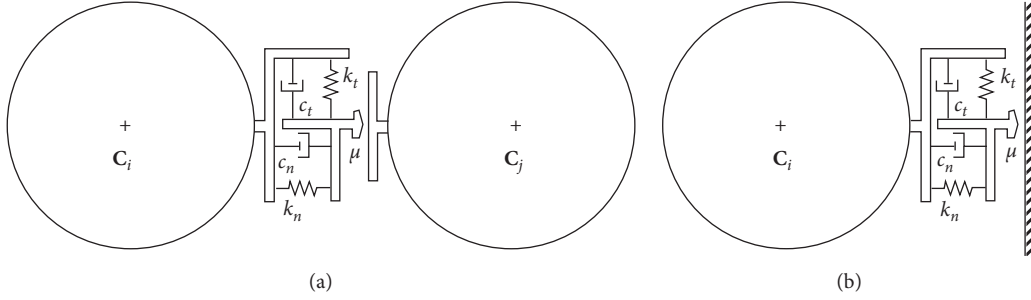(a)   (b)

Figure 27: DEM-DEM and DEM-FEM rheological models. (a) DEM-DEM [21]. (b) DEM-FEM [21].

Table 3: Stiffness values for sphere-sphere and sphere-wall contact [21].

	Sphere i-Sphere j	Sphere i-Wall j
k_n	$(4/(3(((1-\nu_i^2)/E_i)+((1-\nu_j^2)/E_j))))\sqrt{(R_iR_j/(R_i+R_j))\delta_n}$	$(4/(3(((1-\nu_i^2)/E_i)+((1-\nu_j^2)/E_j))))\sqrt{R_i\delta_n}$
k_t	$(8/(((2-\nu_i)/G_i)+((2-\nu_j)/G_j)))\sqrt{(R_iR_j/(R_i+R_j))\delta_n}$	$(8G_i/(2-\nu_i))\sqrt{R_i\delta_n}$
c_n	$(2(\zeta_i+\zeta_j)/2)\sqrt{((m_im_j)/(m_i+m_j))k_n}$	$2\zeta_i\sqrt{m_ik_n}$
c_t	$(2(\zeta_i+\zeta_j)/2)\sqrt{((m_im_j)/(m_i+m_j))k_t}$	$2\zeta_i\sqrt{m_ik_t}$

In order to obtain Δs, the tangent unit vector $\mathbf{t}^{ij}$ must be derived by splitting the velocity at the contact point $\mathbf{v}_c^{ij}$ into normal $\mathbf{v}_{c,n}^{ij}$ and tangential $\mathbf{v}_{c,t}^{ij}$ components. First, the contact point velocity is expressed with the aid of the respective element velocity $\mathbf{v}_i$, element angular velocity $\boldsymbol{\omega}_i$, and the vector $\mathbf{r}_c^{ij}$ connecting the particle centre and the contact point:

$$\mathbf{v}_c^{ij}=\left(\boldsymbol{\omega}_j\times\mathbf{r}_c^{ji}+\mathbf{v}_j\right)-\left(\boldsymbol{\omega}_i\times\mathbf{r}_c^{ij}+\mathbf{v}_i\right), \quad (A.2)$$

and then, split up

$$\begin{aligned}\mathbf{v}_{c,n}^{ij}&=\left(\mathbf{v}_c^{ij}\cdot\mathbf{n}^{ij}\right)\mathbf{n}^{ij},\\ \mathbf{v}_{c,t}^{ij}&=\mathbf{v}_c^{ij}-\mathbf{v}_{c,n}^{ij},\end{aligned} \quad (A.3)$$

which allows us to express Δs using the tangent unit vector, the element displacement $\mathbf{u}_i$, and the element rotation $\boldsymbol{\Theta}_i$:

$$\begin{aligned}\Delta s&=\left\|\mathbf{u}^{ij}\cdot\mathbf{t}^{ij}\right\|,\\ \mathbf{t}^{ij}&=\frac{\mathbf{v}_{c,t}^{ij}}{\left\|\mathbf{v}_{c,t}^{ij}\right\|},\\ \mathbf{u}^{ij}&=\left(\boldsymbol{\Theta}_j\times\mathbf{r}_c^{ji}+\mathbf{u}_j\right)-\left(\boldsymbol{\Theta}_i\times\mathbf{r}_c^{ij}+\mathbf{u}_i\right).\end{aligned} \quad (A.4)$$

For a Hertz–Mindlin spring-dashpot contact model (denominated as HM + D in [34]), as shown in Figures 27(a) and 27(b), the normal F_n and tangential F_t contact forces [21, 34] for the case of two spheres colliding,

$$\begin{aligned}F_n&=k_n\delta_n+c_n\dot{\delta}_n,\\ F_{te}^n&=F_{te}^{n-1}+k_t^n\Delta s^n, \quad if\ \Delta F_n\geq0,\\ F_{te}^n&=F_{te}^{n-1}\frac{k_t^n}{k_t^{n-1}}+k_t^n\Delta s^n, \quad if\ \Delta F_n<0,\\ F_t^n&=\min\left[F_{te}^n+c_tv_{c,t}^{ij},\mu F_n\right].\end{aligned} \quad (A.5)$$

are calculated with the aid of the normal and tangential stiffness k_n and k_t, respectively, and the damping coefficients c_n and c_t [21], considering the maximum tangential force restricted to the Coulomb's friction limit [34] with the coefficient of friction μ. The tangential forces are consequently updated from the last step, indicated by the superscript n.

The material parameters in Table 3 (Young's modulus E, particle mass m, shear modulus G, and Poisson's ratio ν) are typically obtained by calibration from experiments [49].

As a scaling factor in Table 3, the dashpot coefficient ζ [21, 34] is frequently expressed using the normal coefficient of restitution (COR) $\varepsilon_\perp$ as shown in the following equation:

$$\varepsilon_\perp=-\frac{\dot{\delta}_n^{\text{after}}}{\dot{\delta}_n^{\text{before}}}, \quad (A.6)$$

$$\zeta=-\frac{\ln\varepsilon_\perp}{\sqrt{\pi^2+\left(\ln\varepsilon_\perp\right)^2}}. \quad (A.7)$$

Based on our experience, which is as well as supported by [34, 35], a more realistic modelling of the dashpot coefficient ζ is given by the following equation [34, 35]:

$$\zeta=\sqrt{\frac{1.0}{1.0-\left(1.0+\varepsilon_\perp\right)^2\cdot e^\Lambda}-1.0}, \quad (A.8)$$

$$\begin{aligned}\Lambda=\varepsilon_\perp\cdot\big(h_1+\varepsilon_\perp\cdot\big(h_2+\varepsilon_\perp\cdot\big(h_3+\varepsilon_\perp\cdot\big(h_4+\varepsilon_\perp\cdot\big(h_5+\varepsilon_\perp\\ \cdot\big(h_6+\varepsilon_\perp\cdot\big(h_7+\varepsilon_\perp\cdot\big(h_8+\varepsilon_\perp\cdot\big(h_9+\varepsilon_\perp\cdot h_{10}\big)\big)\big)\big)\big)\big)\big)\big)\big),\end{aligned} \quad (A.9)$$

```
(1) ### Import Applications ####
(2) # Structural Mechanics Application is for FEM analysis, in this scope used for the cable structures.
(3) from KratosMultiphysics.StructuralMechanicsApplication import structural_mechanics_analysis as structural_analysis
(4) # DEM Application is for DEM analysis; in this scope, it holds certain expressions for walls and can also deal with clustered
    particles.
(5) from KratosMultiphysics.DEMApplication import KratosDEMAnalysis as dem_analysis
(6) # Mapping Application is to allow a mapping between certain spaces; it is used to handle certain interfaces to make the procedures
    more generic.
(7) import KratosMultiphysics.MappingApplication as KratosMapping
(8) ### Define Problem Setup ####
(9) # model part for all faces/boundary walls
(10) mp_dem = dem_analysis.rigid_face_model_part
(11) # model part for all DEM particles
(12) mp_dem_particle = dem_analysis.spheres_model_part
(13) # Analysis model and model part for structural elements
(14) model = KratosMultiphysics.Model()
(15) mp_struct = model["Structure.computing_domain"]
(16) # Create mapper and define relations. It relates the model parts of the walls in DEM to the cable structures in FEM
(17) mapper = KratosMapping.MapperFactory.CreateMapper(mp_dem, mp_struct, mapper_settings)
(18) # Create utility to optimize contact detection
(19) dem_mesh_moving_utility = DEMApplication.MoveMeshUtility()
(20) ### Initialize Application Setup ####
(21) # Initialize all necessary variables within the applications
(22) structural_analysis.Initialize()
(23) dem_analysis.Initialize()
```

ALGORITHM 4: Problem setup.

```
(1)  ### Start Time Loop ####
(2)  while dem_analysis.time < dem_analysis.end_time:
(3)      # increase time step if particles are not near to the interface
(4)      if not dem_mesh_moving_utility. CheckIsNearToWall(mp_dem_particle.Nodes):
(5)         dem_analysis.SetDeltaTime(multiply = 100.0)
(6)      # reset time step if particles are near to the interface
(7)      else:
(8)         dem_analysis.SetDeltaTime(multiply = 1.0)
(9)      ### Solve DEM Problem ####
(10)     # update time parameters
(11)     dem_analysis._UpdateTimeParameters()
(12)     # search and find neighbouring elements/particles which are in contact
(13)     dem_analysis.SearchOperations()
(14)     # calculate contact forces
(15)     dem_analysis.ForceOperations()
(16)     # integrate in time to obtain new position and velocity of DEM particles
(17)     dem_analysis.IntegrationOfMotion()
(18)     # finalize time step by updating state variables
(19)     dem_analysis.FinalizeSingleTimeStep()
(20)     # check for contact forces and solve FEM part if contact forces exist
(21)     if dem_mesh_moving_utility.CheckContact(mp_dem.Nodes):
(22)        ### Map Contact Forces ####
(23)        # DEM to Structure
(24)        mapper.Map(DEMApplication.CONTACT_FORCES, StructuralMechanicsApplication.POINT_LOAD)
(25)        ### Solve Structural Mechanics Problem ####
(26)        structural_analysis.AdvanceInTime()
(27)        # set the previous configuration as the current configuration
(28)        structural_analysis.InitializeSolutionStep()
(29)        # prediction step for solution scheme if necessary
```

ALGORITHM 5: Continued.

```
(30)     structural_analysis.Predict()
(31)     # solve the FEM system of equations or explicitly integrate in time
(32)     structural_analysis.SolveSolutionStep()
(33)     # finalize time step by updating state variables and spatial position
(34)     structural_analysis.FinalizeSolutionStep()
(35)     ### Map Velocity and Displacement ####
(36)     # Structure to DEM
(37)     mapper.InverseMap(VELOCITY)
(38)     mapper.InverseMap(DISPLACEMENT)
(39)     # update position of DEM wall condition
(40)     dem_analysis.MoveMesh()
(41) ### Finalize Applications ####
(42) # e.g., free memory, make output, ...
(43) dem_analysis.Finalize()
(44) structural_analysis.Finalize()
```

ALGORITHM 5: Weak coupling.

```
(1) ### Start Time Loop ####
(2) while dem_analysis.time < dem_analysis.end_time:
(3)    # update time parameters
(4)    dem_analysis.AdvanceInTime()
(5)    structural_analysis.AdvanceInTime()
(6)    # save the current position, forces, velocity, etc., of the DEM wall condition and particles
(7)    dem_analysis.SaveCurrentData()
(8)    # initialize time step
(9)    structural_analysis.InitializeSolutionStep()
(10)   # initial interface residuals
(11)   InitializeResiduals()
(12)   while interface_residual > interface_tolerance:
(13)       ### Solve DEM Problem ####
(14)       # reset the previous saved data of the particle to keep it at the same reference position in each inner loop step
(15)       dem_analysis.SetOldDataParticles()
(16)       # search and find neighbouring elements/particles which are in contact
(17) dem_analysis.SearchOperations()
(18) # calculate contact forces
(19)       dem_analysis.ForceOperations()
(20)       # integrate in time to obtain new position and velocity of DEM particles
(21)       dem_analysis.IntegrationOfMotion()
(22)       ### Map Contact Forces ####
(23)       # DEM to Structure
(24)       mapper.Map(DEMApplication.CONTACT_FORCES, StructuralMechanicsApplication.POINT_LOAD)
(25)       ### Solve Structural Mechanics Problem ####
(26)       # prediction step for solution scheme (if necessary)
(27)       structural_analysis.Predict()
(28)       # solve the FEM system of equations or explicitly integrate in time
(29)       structural_analysis.SolveSolutionStep()
(30)       ### Map Velocity and Displacement ####
(31)       # Structure to DEM
(32)       mapper.InverseMap(VELOCITY)
(33)       mapper.InverseMap(DISPLACEMENT)
(34)       ### Calculate Interface Residuals ####
(35)       calculate_displacement_residual()
(36)       calculate_velocity_residual()
(37)       # use the maximum residual for the convergence check
(38)       interface_residual = max(displacement_residual, velocity_residual)
(39)       ### Relax Exchange Data ####
(40)       dem_analysis.RelaxDisplacementAndVelocity()
```

ALGORITHM 6: Continued.

```
(41)        dem_analysis.SetRelaxedDisplacementAndVelocity()
(42)        ### Update DEM ####
(43)        # update position of DEM wall
(44)        dem_analysis.MoveMesh()
(45)        # use the current position of the DEM wall condition and the last converged position to calculate the difference in
      displacement, which is used to calculate contact force
(46)        dem_analysis.CalculateDeltaDispFromIntermediatePos()
(47)     # finalize time step by updating state variables and spatial position
(48)     structural_analysis.FinalizeSolutionStep()
(49)     dem_analysis.FinalizeSingleTimeStep()
(50) ### Finalize Applications ####
(51) dem_analysis.Finalize()
(52) structural_analysis.Finalize()
```

ALGORITHM 6: Strong coupling: relax displacements and velocities.

```
(1) ### Start Time Loop ####
(2) while dem_analysis.time < dem_analysis.end_time:
(3)    # update time parameters
(4)    dem_analysis.AdvanceInTime()
(5)    structural_analysis.AdvanceInTime()
(6)    # save the current position, forces, velocity, etc., of the DEM wall condition and particles
(7)    dem_analysis.SaveCurrentData()
(8)    # initialize time step
(9)    structural_analysis.InitializeSolutionStep()
(10)    # initial interface residuals
(11)    InitializeResiduals()
(12)    while interface_residual > interface_tolerance:
(13)       ### Solve DEM Problem ####
(14)       # reset the previous saved data of the particle to keep it at the same reference position in each inner loop step
(15)       dem_analysis.SetOldDataParticles()
(16)       # search and find neighbouring elements/particles which are in contact
(17)       dem_analysis.SearchOperations()
(18)       # calculate contact forces
(19)       dem_analysis.ForceOperations()
(20)       # integrate in time to obtain new position and velocity of DEM particles
(21)       dem_analysis.IntegrationOfMotion()
(22)       ### Map Contact Forces ####
(23)       # DEM to Structure
(24)       mapper.Map(DEMApplication.CONTACT_FORCES, StructuralMechanicsApplication.POINT_LOAD)
(25)       ### Calculate Interface Residual ####
(26)       interface_residual = calculate_force_residual()
(27)       ### Relax Exchange Data ####
(28)       dem_analysis.RelaxForces()
(29)       dem_analysis.SetRelaxedForces()
(30)       # prediction step for solution scheme (if necessary)
(31)       structural_analysis.Predict()
(32)       # solve the FEM system of equations or explicitly integrate in time
(33)       structural_analysis.SolveSolutionStep()
(34)       ### Map Velocity and Displacement ####
(35)       # Structure to DEM
(36)       mapper.InverseMap(VELOCITY)
(37)       mapper.InverseMap(DISPLACEMENT)
(38)       ### Update DEM ####
(39)       # update position of DEM wall
(40)       dem_analysis.MoveMesh()
```

ALGORITHM 7: Continued.

```
(41)      # use the current position of the DEM wall condition and the last converged position to calculate the difference in
          displacement, which is used to calculate
(42)        dem_analysis.CalculateDeltaDispFromIntermediatePos()
(43)    # finalize time step by updating state variables and spatial position
(44)    structural_analysis.FinalizeSolutionStep()
(45)    dem_analysis.FinalizeSingleTimeStep()
(46) ### Finalize Applications ####
(47) dem_analysis.Finalize()
(48) structural_analysis.Finalize()
```

ALGORITHM 7: Strong coupling: relax forces.

$$\mathbf{h} = [-6.918798, -16.41105, 146.8049, -796.4559, 2928.711,$$
$$-7206.864, 11494.29, -11342.18, 6276.757, -1489.915].$$
$$(A.10)$$

The difference can be observed in Figures 15(a) and 15(b). For further information about this topic, the reader is redirected to [21, 34, 35].

With respect to [21, 35], the COR expresses the ratio between the velocity after $\dot{\delta}_n^{after}$ and the velocity before $\dot{\delta}_n^{before}$ impact. A maximum of $\varepsilon_\perp = 1.0$ will model a perfectly elastic impact, whereas $\varepsilon_\perp = 0.0$ models a perfectly plastic impact. A smaller COR increases the influence of the impact velocity in the contact force calculation [21, 34] (equation (A.5)) and is thus critical for the coupled simulation in this study, in which a Hertz–viscous–Coulomb contact law is used [21].

For frictional cohesion-less contact as used in this study, the normal force must be constrained to always be ≥ 0.0 [21], since no traction in normal direction is allowed. To correctly capture this behaviour, Cummins et al. [34] describe how to modify the contact duration calculation by using the dashpot coefficient as described by equation (A.8). For further discussion on the contact forces, additional contact laws, and specific DEM-related topics such as contact duration, etc., see [21, 34, 35, 50, 51].

B. Code Scripts and Development Environment

To give the interested reader a better understanding and the possibility to reproduce the results, the algorithms are presented with the notation used. The open-source multiphysics software *KRATOS* [48] was used for this study. It can be downloaded [52]. An installation guideline is provided there, too. *KRATOS* is designed in *C++* and includes a *Python* interface to facilitate the advanced development and simulation. Documentation for the *Python* scripts used in this study is provided in the following. To run the simulation, the structural mechanics application, the discrete element application, and the mapping application are required.

B.1. Problem Setup. The script to define the problem setup is shown in the following. This initialization script is for both the weak and the strong coupling approach, which are described in the following appendices (Algorithm 4).

B.2. Weak Coupling Algorithm. First, the *Python* script to run the weak coupling algorithm (Section 5) is provided. This code sequence describes two possibilities to improve the efficiency of the simulation: one by increasing the time step if particles are far away from the interface and the other by solving the FEM part only if contact forces have been detected. These two features are omitted in Appendix B.3 for simplicity purposes. However, they can be used to optimize computation time (Algorithm 5).

B.3. Strong Coupling Algorithm. The two strong coupling approaches described in Section 6 are depicted in the following algorithms. First, the procedure to relax displacements and velocities is explained, followed by the procedure to relax the transferred forces (Algorithms 6 and 7).

B.3.1. Relax Displacements and Velocities algorithm 7
B.3.2. Relax Forces

Data Availability

No data were used to support the findings of the study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

All the authors prepared the manuscript. All the authors read and approved the final manuscript.

Acknowledgments

This work was supported by the Technical University of Munich (TUM).

References

[1] A. Volkwein, *Numerische Simulation von flexiblen Steinschlagschutzsystemen*, ETH, Zurich, Switzerland, 2004.
[2] A. Mentani, L. Govoni, A. Giacomini, G. Gottardi, and O. Buzzi, "An equivalent continuum approach to efficiently model the response of steel wire meshes to rockfall impacts," *Rock Mechanics and Rock Engineering*, vol. 51, no. 9, pp. 2825–2838, 2018.

[3] S. Tahmasbi, A. Giacomini, C. Wendeler, and O. Buzzi, "On the computational efficiency of the hybrid approach in numerical simulation of rockall flexible chain-link mesh," *Rock Mechanics and Rock Engineering*, vol. 52, no. 10, pp. 3849–3866, 2019.

[4] R. Boulaud and C. Douthe, "A sliding cable model for rockfall barrier simulations using dynamic relaxation," in *Proceedings of the IASS Annual Symposium*, Hamburg, Germany, September 2017.

[5] X. Zhang, Z. Chen, and Y. Liu, *The Material Point Method: A Continuum-Based Particle Method for Extreme Loading Cases*, Academic Press, Cambridge, MA, USA, 2016.

[6] M. Santasusana, Miquel, J. Irazábal, E. Oñate, and J. M. Carbonell, "The Double Hierarchy Method. A parallel 3D contact method for the interaction of spherical particles with rigid FE boundaries using the DEM," *Computational Particle Mechanics*, vol. 3, no. 3, pp. 407–428, 2016.

[7] J. Stransky and M. Jirásek, "Open source DEM–FEM coupling," in *Proceedings of the Engineering Mechanics*, Stuttgart, Germany, May 2012.

[8] H. Haddad, M. Guessasma, and J. Fortin, "A DEM–FEM coupling based approach simulating thermomechanical behaviour of frictional bodies with interface layer," *International Journal of Solids and Structures*, vol. 81, pp. 203–218, 2016.

[9] J. Denape, "Third body concept and wear particle behavior in dry friction sliding conditions," *Tribological Aspects in Modern Aircraft Industry*, pp. 1–12, 2014.

[10] L. Cao, F. Sadeghi, and L.-E. Stacke, "A combined EFEM-DEM dynamic model of rotor-bearing-housing system," *Journal of Tribology*, vol. 139, no. 6, 2017.

[11] B. Baskaran, S. M. Srinivasan, B. Maffeo et al., "Shot peening simulation using discrete and finite element methods," *Advances in Engineering Software*, vol. 41, no. 12, pp. 1266–1276, 2010.

[12] K. Kovthaman, S. Utili, and N. Petrinic, "A combined DEM–FEM numerical method for Shot Peening parameter optimization," *Advances in Engineering Software*, vol. 79, pp. 13–26, 2015.

[13] A. Ashtekar and F. Sadeghi, "A new approach for including cage flexibility in dynamic bearing models by using combined explicit finite and discrete element methods," *Journal of Tribology*, vol. 134, no. 4, 2012.

[14] C. Bovet and L. Zamponi, "An approach for predicting the internal behaviour of ball bearings under high moment load," *Mechanism and Machine Theory*, vol. 101, pp. 1–22, 2016.

[15] J. Takabi and M. M. Khonsari, "On the dynamic performance of roller bearings operating under low rotational speeds with consideration of surface roughness," *Tribology International*, vol. 86, pp. 62–71, 2015.

[16] K. A. Kounoudji, M. Renouf, G. Mollon, and Y. Berthier, "Role of third body on bolted joints' self-loosening," *Tribology Letters*, vol. 61, no. 3, 2016.

[17] S. Liu, H. Li, S. Shen, and S. Wu, "Simulation of particle rebounding from the slider air bearing surface," *Microsystem Technologies*, vol. 22, no. 6, pp. 1475–1481, 2016.

[18] N. T. Ngo, B. Indraratna, and C. Rujikiatkamjorn, "Coupled DEM-FEM analysis for simulating ballasted rail tracks," in *Proceedings of the 15th IACMAG*, Wuhan, China, October 2017.

[19] E. Oñate, F. Zárate, M. A. Celigueta et al., "Advances in the DE and coupled DEM and FEM techniques in non linear solid mechanics," *Advances in Computational Plasticity: A Book in Honour of D. Roger J. Owen*, Scipedia, Barcelona, Spain, 2018.

[20] E. Oñate, M. A. Celigueta, S. Latorre et al., "Lagrangian analysis of multiscale particulate flows with the particle Finite Element Method," *Computational Particle Mechanics*, vol. 1, no. 1, pp. 85–102, 2014.

[21] M. Santasusana, *Numerical Techniques for Non-linear Analysis of Structures Combining Discrete Element and Finite Element Methods*, CIMNE, Barcelona, Spain, 2016.

[22] T. Belytschko, W. Liu, B. Moran, and K. Elkhodary, *Nonlinear Finite Elements for Continua and Structures*, Wiley, Haboken, NJ, USA, 2013.

[23] U. Küttler, *Effiziente Lösungsverfahren für Fluid-Struktur-Interaktions-Probleme*, TUM, Munich, Germany, 2009.

[24] R. Wüchner, *Mechanik und Numerik der Formfindung und Fluid-Struktur-Interaktion von Membrantragwerken*, TUM, Munich, Germany, 2006.

[25] G. Strang and G. Fix, *An Analysis of the Finite Element Method*, Wellesley-Cambridge Press, Wellesley, MA, USA, 2008.

[26] E. A. de Souza Neto, D. Peric, and D. R. J. Owen, *Computational Methods for Plasticity: Theory and Applications*, Wiley, Haboken, NJ, USA, 2011.

[27] J. Bonet and R. D. Wood, *Nonlinear Continuum Mechanics for Finite Element Analysis*, Cambridge University Press, Cambridge, UK, 2008.

[28] G. A. Holzapfel, *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*, Wiley, Haboken, NJ, USA, 2000.

[29] A. M. Widhammer, *Variation of Reference Strategy*, TUM, Munich, Germany, 2015.

[30] Y. Basar and D. Weichert, *Nonlinear Continuum Mechanics of Solids: Fundamental Mathematical and Physical Concepts*, Springer, Berlin Heidelberg, 2000.

[31] T. Drieseberg, *Ein Beitrag zur Formfindung von Tensegrity-Systemen mit der Kraftdichtemethode*, Kassel University Press, Kassel, Germany, 2007.

[32] A. Alipour and F. Zareian, "Study Rayleigh damping in structures; unceratinties and treatments," in *Proceedings of the the 14th World Conference on Earthquake Engineering*, Beijing, China, October 2008.

[33] B. Smeets, T. Odenthal, S. Vanmaercke, and H. Ramon, "Polygon-based contact description for modeling arbitrary polyhedra in the Discrete Element Method," *Computer Methods in Applied Mechanics and Engineering*, vol. 290, pp. 277–289, 2015.

[34] S. Cummins, C. Thornton, and C. Cleary, "Contact force models in inelastic collisions," in *Proceedings of the Ninth International Conference on CFD in the Minerals and Process*, Melbourne, Australia, December 2012.

[35] T. Schwager and T. Pöschel, "Coefficient of restitution and linear--dashpot model revisited," *Granular Matter*, vol. 9, no. 6, pp. 465–469, 2007.

[36] T. Wang, *Development of Co-simulation Environment and Mapping Algorithms*, TUM, Munich, Germany, 2016.

[37] A. Winterstein, C. Lerch, K.-U. Bletzinger, and R. Wüchner, "Partitioned simulation strategies for fluid-structure-control interaction problems by Gauss-Seidel formulations," *Advanced Modeling and Simulation in Engineering Sciences*, vol. 5, no. 1, 2018.

[38] C. Farhat, K. G. van der Zee, and P. Geuzaine, "Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity," *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 17-18, pp. 1973–2001, 2006.

[39] S. Sicklinger, *Stabilized Co-simulation of Coupled Problems Including Fields and Signals*, TUM, Munich, Germany, 2014.

[40] U. Küttler and W. Wall, "Fixed-point fluid-structure interaction solvers with dynamic relaxation," *Computational Mechanics*, vol. 43, no. 1, pp. 61–72, 2008.

[41] C. Aitken, "XXV.—on Bernoulli's numerical solution of algebraic equations," in *Proceedings of the Royal Society of Edinburgh*, Edinburgh, UK, 1927.

[42] N. Sabatakakis, N. Depountis, and N. Vagenas, "Evaluation of rockfall restitution coefficients," *Engineering Geology for Society and Territory* -, vol. 2, pp. 2023–2026, 2015.

[43] A. Pavlos, H. Saroglou, and G. Tsiambaos, "Rockfall: scaling factors for the coefficient of restitution," in *Proceedings of the Eurock 2013 Rock Mechanics for Resources, Energy & Environment*, Wrocław, Poland, Septmeber 2013.

[44] K. T. Chau, R. H. C. Wong, and J. J. Wu, "Coefficient of restitution and rotational motions of rockfall impacts," *International Journal of Rock Mechanics and Mining Sciences*, vol. 39, no. 1, pp. 69–77, 2002.

[45] Geobrugg, https://www.geobrugg.com/.

[46] J. P. E. Osorio, *Simulation of Flexible Steel Wire-Net Rock-Fall Barriers via Finite Element Model Updating*, ETH, Zurich, Switzerland, 2015.

[47] A. Volkwein, "Numerical simulation of flexible rockfall protection systems," *Computing in Civil Engineering*, 2005.

[48] KRATOS Multi-Physics, https://www.cimne.com/kratos/.

[49] C. J. Coetzee and D. N. J. Els, "Calibration of granular material parameters for DEM modelling and numerical verification by blade–granular material interaction," *Journal of Terramechanics*, vol. 46, no. 1, pp. 15–26, 2009.

[50] G. Casas, D. Mukherjee, M. A. Celigueta, T. I. Zohdi, and E. Oñate, "A modular, partitioned, Discrete Element framework for industrial grain distribution systems with rotating machinery," *Computational Particle Mechanics*, vol. 4, no. 2, pp. 181–198, 2017.

[51] H. A. Navarro and M. P. de Souza Braun, "Determination of the normal spring stiffness coefficient in the linear spring-–dashpot contact model of discrete element method," *Powder Technology*, vol. 246, pp. 707–722, 2013.

[52] KRATOS Multi-Physics - GitHub Repository, https://github.com/KratosMultiphysics/Kratos.