May, 2017

Co-rotated and reduced order finite element time integrators for multibody contact dynamics

Summary

In this work we present two linearly implicit time integration methods, suitable for deformable multibody dynamic contact problems. The first approach co-rotates the initial linearized dynamic system which is therefore factorized only once. The second approach employs a reduced order base and produces a smaller, dense algebraic system; a modal base can also be used rendering the reduces system diagonal. We cross-compare the proposed approaches with the unmodified Total Lagrangian finite element formulation and with the rigid body approach. An implementation of presented methods is available at https://github.com/tkoziara/solfec/

Contributors

Tomasz Koziara, t.koziara@gmail.com
Steve Brasier, steve.brasier@atkinsglobal.com
Łukasz Kaczmarczyk, lukasz.kaczmarczyk@glasgow.ac.uk

This report is dedicated to Nenad:)

1. Introduction

Rigid kinematics is commonly used in multibody simulations (e.g. in robotics or to simulate granular flow). This level of kinematic description is often sufficient when dynamic effects corresponding to the deformable behavior of bodies can be neglected. Whenever this is not the case though, one runs into the computational bottleneck of processing many interacting bodies, discretized with fine finite element meshes. For robustness reasons explicit codes are usually picked to process such problems, although typically the dynamics on the time step scale is not of interest. On the other hand, general implicit FE codes usually struggle to process multibody contact/impact problems with many interacting bodies. For problems involving 100s, 1000s and more bodies, there is still little choice left (but explicit codes) in the realm of the off the shelf approaches.

In this paper we are in the context of multibody contact/impact problems, solved by means of implicit formulations. We address one of the building blocks of such frameworks, namely time integration of deformable kinematics. We assume that the bodies are stiff, but undergo large rotations, as it is common for machines, blocky structures or granular flows. We reuse ideas from [7] (cf. Appendix A) and formulate two time integration methods, loosely based upon the scheme suggested by Zhang and Skeel [13]. The first approach only co-rotates the initial linearized dynamic system. The computational

gain stems from the fact that the system is factorized only once. The second approach additionally employs a reduced order base and hence it produces a smaller, dense system. This approach is efficient in the context of implicit multibody dynamics involving impacts, where its small/dense algebra facilitates fast updates of the evolving and often rapidly changing system of constraint equations. An implementation of the presented approaches is include into an MPI-parallel code, Solfec, available at https://github.com/tkoziara/solfec/.

The paper is organized as follows. Section 2 introduces some basic ides. Section 3 briefly outlines the Total Lagrangian time integrator. Two simplifications follow: Section 4 describes the co-rotated time stepping approach without order reduction, while Section 5 details the reduced order approach. Section 6 contains some remarks about the treatment of constraints. Section 7 briefly introduces the impact/friction model and comments on energetic consistency. The examples follow in Section 8. The paper is concluded in Section 9, followed by appendices. Appendix A outlines the method of updating rotation; Appendix B shows how to replace the reduced order base with a modal one; Appendix C provides links to source code and examples files.

2. Basic ideas

In this paper we work with bodies capable of large rigid motion and small elastic deformations (e.g. blocks of wood, metal or rock). Bodies are discretized by means of finite elements, with $\mathbf{q}$ denoting nodal displacements and $\mathbf{u}$ denoting nodal velocities, $\mathbf{u} = \dot{\mathbf{q}}$. Taking a single body, we parametrize its motion as

$$\mathbf{x}\left(\mathbf{X}, t\right) = \mathbf{X} + \mathbf{S}\left(\mathbf{X}\right)\mathbf{q}\left(t\right) \tag{1}$$

where $\mathbf{x}$ is a spatial point, $\mathbf{X}$ is a referential point, t is time, and $\mathbf{S}$ stores finite element shape functions. Further derivations are motivated by a simple reasoning. We would like to model bodies interacting through frictional contact in dynamic situations. We decide to use the implicit in time Contact Dynamics method [6, 11], where nonlinearity is located both at the level of the per-body momentum balance (e.g. finite elasticity) and on the level of individual constraints (e.g. contact and friction). We wish to simplify computations and only deal with the nonlinearity at the constraints. Hence, we are interested in a linearly implicit time integration scheme. We start from the Total Lagrangian formulation [4], which produces a constant mass matrix $\mathbf{M}$ and a stiffness matrix $\mathbf{K}$ that depends on the nodal displacements $\mathbf{q}$. For simplicity we diagonalize the mass matrix $\mathbf{M}$ by summing up its rows. For the Total Lagrangian formulation, in Section 3, we employ a linearly implicit integration scheme proposed by Zhang and Skeel [13], where an operator employed in the discrete momentum balance has the following form

$$\mathbf{A}_1\left(t\right) = \mathbf{M} + \gamma\mathbf{K}\left(\mathbf{q}\left(t\right)\right) \tag{2}$$

with $\gamma > 0$. Having to deal with this operator is expensive: it needs to be recomputed and inverted at every time step. Realizing that our bodies are roughly rigid we then ask: what if the above operator could be approximated by

$$\mathbf{A}_2\left(t\right) = \mathbf{M} + \gamma\underline{\mathbf{\Lambda}}\left(t\right)\mathbf{K}_0\underline{\mathbf{\Lambda}}\left(t\right)^T \tag{3}$$

where $\mathbf{\Lambda}$ is a 3×3 rotation operator and $\underline{\mathbf{\Lambda}}$ denotes a block-diagonal square matrix composed of $\mathbf{\Lambda}$ and matching the dimension of $\mathbf{K}_0 = \mathbf{K}\left(\mathbf{q}\left(0\right)\right)$. Because $\mathbf{M}$ is diagonal and for each node has the same x, y, z components, we can write

$$\mathbf{A}_2\left(t\right) = \underline{\mathbf{\Lambda}}\left(t\right)\left(\mathbf{M} + \gamma\mathbf{K}_0\right)\underline{\mathbf{\Lambda}}\left(t\right)^T = \underline{\mathbf{\Lambda}}\left(t\right)\mathbf{A}_0\underline{\mathbf{\Lambda}}\left(t\right)^T \tag{4}$$

where $\mathbf{A}_0 = \mathbf{M} + \gamma\mathbf{K}_0$ needs to be computed and inverted only once. In Section 4 we simply rewrite the Total Lagrangian time stepping with this idea in mind. Finally, in Section 5, we investigate whether a further saving would be possible by assuming that $\mathbf{A}_0 = \mathbf{E}^T\left(\mathbf{M} + \gamma\mathbf{K}_0\right)\mathbf{E}$, where $\mathbf{E}$ collects selected "eigenmodes". In all three cases the linearly implicit scheme is essentially the same:

1. Compute the first explicit forward half-step.

2. At this half-step write down a discrete form of the momentum balance.

3. Compute the final implicit backward half-step.

To a large degree, the algorithmic details of what follows are a consequence of the fact that we are able to compute $\mathbf{\Lambda}(t)$ in a robust way (Appendix A). Because we consider small deformations about a possibly translated and rotated shape, it makes sense to talk about the gross rigid motion

$$\mathbf{x}_r(\mathbf{X}, t) = \mathbf{\Lambda}(t)(\mathbf{X} - \mathbf{A}) + \mathbf{a}(t) \tag{5}$$

where $\mathbf{\Lambda}$ is the rotation operator, $\mathbf{A}$ is a referential point, and $\mathbf{a}$ is a spatial image of $\mathbf{A}$. These two motions differ by the amount of deformation

$$\mathbf{d}(\mathbf{X}, t) = \mathbf{x}(\mathbf{X}, t) - \mathbf{x}_r(\mathbf{X}, t). \tag{6}$$

Assuming that $\mathbf{Z}_i$ are coordinates of mesh nodes, we can collect $\mathbf{d}(\mathbf{Z}_i, t)$ into a vector

$$\mathbf{d} = \begin{bmatrix} \dots \\ \mathbf{Z}_i + \mathbf{q}_i - \mathbf{\Lambda}(\mathbf{Z}_i - \mathbf{A}) - \mathbf{a} \\ \dots \end{bmatrix} \tag{7}$$

where $\mathbf{q}_i$ represent displacements at node i. In the following sections $\mathbf{d}$ will be multiplied by a symmetric and semi-positive definite stiffness matrix $\mathbf{K}$ whose six zero eigenvalues correspond to the rigid modes (translations and infinitesimal rotations). Since any product of $\mathbf{K}$ with a displacement field corresponding to a rigid translation produces a zero vector, $\mathbf{A}$ and $\mathbf{a}$ can be skipped in the above formula. Hence, we can redefine $\mathbf{d}$ as

$$\mathbf{d} = \begin{bmatrix} \dots \\ (\mathbf{I} - \mathbf{\Lambda})\mathbf{Z}_i + \mathbf{q}_i \\ \dots \end{bmatrix} = \underline{(\mathbf{I} - \mathbf{\Lambda})}\mathbf{Z} + \mathbf{q} \tag{8}$$

where $\mathbf{I}$ is the 3×3 identity and $\underline{\mathbf{I} - \mathbf{\Lambda}}$ denotes a block-diagonal square matrix composed of $(\mathbf{I} - \mathbf{\Lambda})$. Equation (8) corresponds to the below deformation evaluated at mesh nodes

$$\mathbf{d}(\mathbf{X}, t) = \mathbf{x}(\mathbf{X}, t) - \mathbf{\Lambda}(t)\mathbf{X} \tag{9}$$

which, as a difference between the current deformed position and the current rotated position (without translation), superposes the rigid translation and the small deformations of the body. As a result antisymmetric part of the gradient of (9) equals to the antisymmetric part of the gradient of (6). This fact is used in Appendix A. We should note, that simplification (8) is not necessary in general and it would make no difference if $\mathbf{A}$ and $\mathbf{a}$ were maintained throughout our derivations. The simplification stems from our implementation [9], where kinematic models are combined with Lagrange multipliers imposing boundary conditions, and hence all bodies are nominally unconstrained.

3. Total Lagrangian time stepping (TL)

Consider a multi-body domain with constraints. Let the constraint reactions be denoted by $\mathbf{R}$ and the relative constraints velocities be denoted by $\mathbf{U}$. Let also the constraints equations be denoted by $\mathbf{C}(\mathbf{U}, \mathbf{R}) = \mathbf{0}$. $\mathbf{R}$ and $\mathbf{U}$ can be pictured as column vectors comprising 3-component sub-vectors, denoting reactions and velocities at individual constraint points. When talking about $\mathbf{q}$, $\mathbf{u}$ etc. we mean suitable collections of per-body entities. In order to integrate in time, a half-step configuration is extrapolated first

$$\mathbf{q}^{t+h/2} = \mathbf{q}^t + \frac{h}{2}\mathbf{u}^t \tag{10}$$

A global $\mathbf{u}$ to local $\mathbf{U}$ velocity transformation operator is computed next

$$\mathbf{H} = \mathbf{H}\left(\mathbf{q}^{t+h/2}\right) \tag{11}$$

where $\mathbf{H}$ is a global to local coordinates transformation of the shape functions values nonzero at constraint points. The number of rows of $\mathbf{H}$ depends on the number of constraints, while its rank is related to their linear independence. For example if we think about a relative velocity between two material points of two bodies i and j being in contact at a spatial point $\mathbf{x}_\alpha$, we can express $\mathbf{H}\left(\mathbf{q}^{t+h/2}\right)$ as

$$\mathbf{H}\left(\mathbf{q}^{t+h/2}\right) = \{\mathbf{e}\}^T \left[\begin{array}{ccccc} \text{zeros} & \mathbf{S}_i\left(\mathbf{X}\left(\mathbf{q}_i^{t+h/2}, \mathbf{x}_\alpha\right)\right) & \text{zeros} & -\mathbf{S}_j\left(\mathbf{X}\left(\mathbf{q}_j^{t+h/2}, \mathbf{x}_\alpha\right)\right) & \text{zeros} \end{array} \right] \tag{12}$$

where in addition $\{\mathbf{e}\} = \{\mathbf{e}_{T1}, \mathbf{e}_{T2}, \mathbf{e}_N\}$ is a local orthonormal coordinate system with two tangential and one normal direction, that we may wish to use. In cases with many constraints (e.g. contact points) $\mathbf{H}$ will be composed from multiple rows similar to the three-row operator above. In the next step, the momentum balance

$$\mathbf{b} = h\mathbf{f}_{ext}^{t+h/2} - h\mathbf{f}_{int}\left(\mathbf{q}^{t+h/2}\right) - h\mathbf{K}\left(\mathbf{q}^{t+h/2}\right)\eta\mathbf{u}^t \tag{13}$$

$$\left(\mathbf{M} + \left(\frac{\eta h}{2} + \frac{h^2}{4}\right)\mathbf{K}\left(\mathbf{q}^{t+h/2}\right)\right)\left(\mathbf{u}^{t+h} - \mathbf{u}^t\right) = \mathbf{b} + \mathbf{H}^T\mathbf{R} \tag{14}$$

together with the transformation

$$\mathbf{U} = \mathbf{H}\mathbf{u}^{t+h} \tag{15}$$

and the constraints equations

$$\mathbf{C}\left(\mathbf{U}, \mathbf{R}\right) = \mathbf{0} \tag{16}$$

are used to evaluate the velocity $\mathbf{u}^{t+h}$ and the constraint reactions $\mathbf{R}$. This is further commented on in Section 6. The stiffness-proportional damping in the form of $\eta\mathbf{K}\left(\mathbf{q}^{t+h/2}\right)\left(\mathbf{u}^{t+h} + \mathbf{u}^t\right)/2$ contributes on the left and right hand side of (14). This is a pragmatic device: it allows to avoid uninteresting oscillations as well as, to some extent, drive a post-impact behavior. While more will be said about this in Section 8, for the time being let us assume that this term is going to be useful and preserve it in the following two sections. Finally, the end-step configuration is updated

$$\mathbf{q}^{t+h} = \mathbf{q}^{t+h/2} + \frac{h}{2}\mathbf{u}^{t+h}. \tag{17}$$

The above scheme is linearly implicit: as far as unconstrained motion is concerned only one linear system needs to be solved per time step. Without damping and constraints it amounts to the time-reversible scheme by Zhang and Skeel [13], with the choice of $\alpha = 1/4$ in their algorithm. The most general, fully implicit scheme in [13] can be rewritten as

$$\mathbf{q}^{t+h/2} = \mathbf{q}^t + \frac{h}{2}\mathbf{u}^t \tag{18}$$

$$\mathbf{f}^{t+h/2} = \mathbf{M}^{-1}\left(\mathbf{f}_{ext}^{t+h/2} - \mathbf{f}_{int}\left(\mathbf{q}^{t+h/2} + \alpha h^2\mathbf{f}^{t+h/2}\right)\right) \tag{19}$$

$$\mathbf{u}^{t+h} = \mathbf{u}^t + h\mathbf{f}^{t+h/2} \tag{20}$$

$$\mathbf{q}^{t+h} = \mathbf{q}^{t+h/2} + \frac{h}{2}\mathbf{u}^{t+h} \tag{21}$$

where $\alpha = 1/4$ produces the implicit mid-point rule, exactly conserving energy for linear problems [13]. Nonlinear systems integrated with (18-21) experience a positive energy drift, visible in the long term.

This however is of lesser concern in short simulations, where in addition frictional and viscous effects are purposely introduced. We should note, that exact energy conservation in finite elastodynamics has been achieved algorithmically on the stress integration level [12, 10], something which is not possible if one aims at the kind of simplifications pursued here. As our scheme is akin to the mid-point rule, second order accuracy should be expected for smooth, unconstrained problems. This is indeed the case, as demonstrated numerically in Section 8. The linearly implicit scheme (10-17) follows from (18-21) by starting from $\mathbf{f}^{t+h/2} = \mathbf{0}$ and taking one Newton step

$$\mathbf{M}\left(\mathbf{0} + \triangle\mathbf{f}^{t+h/2}\right) = \mathbf{f}_{ext}^{t+h/2} - \mathbf{f}_{int}\left(\mathbf{q}^{t+h/2} + \alpha h^2\mathbf{0}\right) - \alpha h^2\mathbf{K}\left(\mathbf{q}^{t+h/2} + \alpha h^2\mathbf{0}\right)\triangle\mathbf{f}^{t+h/2} \qquad (22)$$

which for $\alpha = 1/4$ yields

$$\left(\mathbf{M} + \frac{h^2}{4}\mathbf{K}\left(\mathbf{q}^{t+h/2}\right)\right)\triangle\mathbf{f}^{t+h/2} = \mathbf{f}_{ext}^{t+h/2} - \mathbf{f}_{int}\left(\mathbf{q}^{t+h/2}\right) \qquad (23)$$

and combined with (20) results in (10-17), without the constraints and viscous damping.

4. Body co-rotated time stepping (BC)

Once again, consider a multi-body domain with constraints. Let $\mathbf{R}$, $\mathbf{U}$ and $\mathbf{C}\left(\mathbf{U}, \mathbf{R}\right) = \mathbf{0}$ retain their meaning from the previous section. When talking about $\mathbf{q}$, $\mathbf{u}$ *and* $\mathbf{\Lambda}$, etc. we now again mean suitable collections of per-body entities. In order to integrate in time, a half-step configuration is extrapolated first

$$\mathbf{q}^{t+h/2} = \mathbf{q}^t + \frac{h}{2}\mathbf{u}^t \qquad (24)$$

$$\mathbf{\Lambda}_1 = \hat{\mathbf{\Lambda}}\left(\mathbf{q}^{t+h/2}, \mathbf{\Lambda}^t\right). \qquad (25)$$

Function $\hat{\mathbf{\Lambda}}\left(\cdot, \cdot\right)$ is detailed Appendix A; here we treat it as a "black-box". A global $\mathbf{u}$ to local $\mathbf{U}$ velocity transformation operator is computed next

$$\mathbf{H} = \mathbf{H}\left(\mathbf{q}^{t+h/2}\right) \qquad (26)$$

the same way as in the previous section. The momentum balance

$$\mathbf{b} = h\mathbf{f}^{t+h/2} - h\underline{\mathbf{\Lambda}_1}\mathbf{K}_0\underline{\mathbf{\Lambda}_1}^T\left[\left(\mathbf{I} - \mathbf{\Lambda}_1\right)\mathbf{Z} + \mathbf{q}^{t+h/2} + \eta\mathbf{u}^t\right] \qquad (27)$$

$$\underline{\mathbf{\Lambda}_1}\left(\mathbf{M} + \left(\frac{\eta h}{2} + \frac{h^2}{4}\right)\mathbf{K}_0\right)\underline{\mathbf{\Lambda}_1}^T\left(\mathbf{u}^{t+h} - \mathbf{u}^t\right) = \mathbf{b} + \mathbf{H}^T\mathbf{R} \qquad (28)$$

together with the transformation

$$\mathbf{U} = \mathbf{H}\mathbf{u}^{t+h} \qquad (29)$$

and the constraints equations

$$\mathbf{C}\left(\mathbf{U}, \mathbf{R}\right) = \mathbf{0} \qquad (30)$$

are now used to evaluate the velocity $\mathbf{u}^{t+h}$ and the constraint reactions $\mathbf{R}$. Here the stiffness-proportional damping in the form of $\eta\underline{\mathbf{\Lambda}_1}\mathbf{K}\underline{\mathbf{\Lambda}_1}^T\left(\mathbf{u}^{t+h} + \mathbf{u}^t\right)/2$ contributes on the left and right hand side of (28). Finally, the end-step configuration is updated

$$\mathbf{q}^{t+h} = \mathbf{q}^{t+h/2} + \frac{h}{2}\mathbf{u}^{t+h} \qquad (31)$$

$$\mathbf{\Lambda}^{t+h} = \hat{\mathbf{\Lambda}}\left(\mathbf{q}^{t+h}, \mathbf{\Lambda}_1\right). \qquad (32)$$

Without the constraints, the above scheme is again linearly implicit. Combined with a set of nonlinear constraints, it constitutes a fully implicit time-stepping, where the nonlinearity is pushed solely to the constraints. Of course, the whole point of using the co-rotated formulation is in the ease of inverting

$$\mathbf{A} = \underline{\mathbf{\Lambda}}_1 \left(\mathbf{M} + \left(\frac{\eta h}{2} + \frac{h^2}{4} \right) \mathbf{K} \right) \underline{\mathbf{\Lambda}}_1{}^T = \underline{\mathbf{\Lambda}}_1 \mathbf{A}_0 \underline{\mathbf{\Lambda}}_1{}^T \tag{33}$$

since it is enough to invert $\mathbf{A}_0$ only once

$$\mathbf{A}^{-1} = \left[\underline{\mathbf{\Lambda}}_1 \mathbf{A}_0 \underline{\mathbf{\Lambda}}_1{}^T \right]^{-1} = \underline{\mathbf{\Lambda}}_1 \left[\underline{\mathbf{\Lambda}}_1 \mathbf{A}_0 \right]^{-1} = \underline{\mathbf{\Lambda}}_1 \mathbf{A}_0^{-1} \underline{\mathbf{\Lambda}}_1{}^T. \tag{34}$$

5. Body co-rotated, reduced order approach (BC-RO)

The BC time stepping described in the previous section may be useful in some situations. Nevertheless, we can still aim at optimizing the runtime by employing a reduced order base. Let such base be spanned by columns of a matrix $\mathbf{E}$ so that

$$\mathbf{E}^T \mathbf{E} = \operatorname{diag}(1) \tag{35}$$

and hence $\mathbf{E}$ stores orthogonal "eigenvectors". System matrices can be projected onto this basis to produce

$$\bar{\mathbf{M}} = \mathbf{E}^T \mathbf{M} \mathbf{E} \text{ and } \bar{\mathbf{K}} = \mathbf{E}^T \mathbf{K} \mathbf{E}, \tag{36}$$

which are symmetrical and dense. We shall work with two sets of variables, the full (space) configuration and velocity $\mathbf{q}$ and $\mathbf{u}$, and the reduced (space) configuration and velocity $\bar{\mathbf{q}}$ and $\bar{\mathbf{u}}$. As before we also use the rotation operator $\mathbf{\Lambda}$ representing the gross rigid rotation of the entire body. Our intention is to work out a time stepping that closely mimics the co-rotated approach from the previous section, but employs the momentum balance only in the reduced space. We start in the usual way

$$\mathbf{q}^{t+h/2} = \mathbf{q}^t + \frac{h}{2} \mathbf{u}^t \tag{37}$$

$$\mathbf{\Lambda}_1 = \hat{\mathbf{\Lambda}} \left(\mathbf{q}^{t+h/2}, \mathbf{\Lambda}^t \right). \tag{38}$$

Function $\hat{\mathbf{\Lambda}}(\cdot, \cdot)$ is detailed Appendix A. The full mid-step configuration $\mathbf{q}^{t+h/2}$ contains some rigid and some deformable motion extrapolated from the initial state $\mathbf{q}^t$. It may have components along all of the possible deformation modes. The current amount of deformation is

$$\mathbf{d} = \left(\mathbf{I} - \mathbf{\Lambda}_1 \right) \mathbf{Z} + \mathbf{q}^{t+h/2} \tag{39}$$

in the full space. After rotating it back to the reference configuration, it is projected onto the reduced base

$$\bar{\mathbf{q}}^{t+h/2} = \mathbf{E}^T \mathbf{\Lambda}_1^T \mathbf{d} \tag{40}$$

We use the full space half-step in order to produce the needed coupling between the rigid and deformable motions. This gives us the reduced configuration $\bar{\mathbf{q}}^{t+h/2}$ and the reduced internal force can now be computed as $\bar{\mathbf{f}}_{int}^{t+h/2} = \bar{\mathbf{K}} \bar{\mathbf{q}}^{t+h/2}$. We then write down the discrete momentum balance

$$\underline{\mathbf{\Lambda}}_1 \left(\mathbf{M} + \left(\frac{\eta h}{2} + \frac{h^2}{4} \right) \mathbf{K} \right) \underline{\mathbf{\Lambda}}_1^T \left(\mathbf{u}^{t+h} - \mathbf{u}^t \right) = \mathbf{b} + \mathbf{H}^T \mathbf{R} \tag{41}$$

and transform it as follows

$$\mathbf{E}^T \left(\mathbf{M} + \left(\frac{\eta h}{2} + \frac{h^2}{4} \right) \mathbf{K} \right) \mathbf{E} \left(\bar{\mathbf{u}}^{t+h} - \bar{\mathbf{u}}^t \right) = \mathbf{E}^T \underline{\mathbf{\Lambda}}_1^T \left(\mathbf{b} + \mathbf{H}^T \mathbf{R} \right) \tag{42}$$

to produce a reduced system

$$\left(\bar{\mathbf{M}} + \left(\frac{\eta h}{2} + \frac{h^2}{4}\right)\bar{\mathbf{K}}\right)\left(\bar{\mathbf{u}}^{t+h} - \bar{\mathbf{u}}^t\right) = \mathbf{E}^T\underline{\boldsymbol{\Lambda}}_1^T\left(\mathbf{b} + \mathbf{H}^T\mathbf{R}\right). \tag{43}$$

Above, (42) is obtained by left multiplication by $\mathbf{E}^T\underline{\boldsymbol{\Lambda}}_1^T$, combined with an assumption that $\mathbf{u}^{t+h} - \mathbf{u}^t = \underline{\boldsymbol{\Lambda}}_1\mathbf{E}\left(\bar{\mathbf{u}}^{t+h} - \bar{\mathbf{u}}^t\right)$. As a result the reduced momentum balance reads

$$\bar{\mathbf{A}}\left(\bar{\mathbf{u}}^{t+h} - \bar{\mathbf{u}}^t\right) = \bar{\mathbf{b}} + \bar{\mathbf{H}}^T\mathbf{R} \tag{44}$$

where

$$\bar{\mathbf{A}} = \bar{\mathbf{M}} + \left(\frac{\eta h}{2} + \frac{h^2}{4}\right)\bar{\mathbf{K}} \tag{45}$$

$$\bar{\mathbf{H}} = \mathbf{H}\underline{\boldsymbol{\Lambda}}_1\mathbf{E} \tag{46}$$

$$\bar{\mathbf{b}} = h\left(\mathbf{E}^T\underline{\boldsymbol{\Lambda}}_1{}^T\mathbf{f}^{t+h/2} - \bar{\mathbf{K}}\left(\bar{\mathbf{q}}^{t+h/2} + \eta\bar{\mathbf{v}}^t\right)\right) \tag{47}$$

and

$$\bar{\mathbf{v}}^t = \mathbf{E}^T\underline{\boldsymbol{\Lambda}}_1{}^T\mathbf{u}^t. \tag{48}$$

Note, that for the damping contribution to $\bar{\mathbf{b}}$ we use $\bar{\mathbf{v}}^t$ rather than $\bar{\mathbf{u}}^t$, following precisely the time integrator from the previous section. As before (44) is solved together with the transformation

$$\mathbf{U} = \bar{\mathbf{H}}\bar{\mathbf{u}}^{t+h} \tag{49}$$

and the constraints equations

$$\mathbf{C}\left(\mathbf{U}, \mathbf{R}\right) = \mathbf{0}. \tag{50}$$

Once $\bar{\mathbf{u}}^{t+h}$ is known, the next full velocity and configuration read

$$\mathbf{u}^{t+h} = \underline{\boldsymbol{\Lambda}}_1\mathbf{E}\left(\bar{\mathbf{u}}^{t+h} - \bar{\mathbf{u}}^t\right) + \mathbf{u}^t \tag{51}$$

$$\mathbf{q}^{t+h} = \mathbf{q}^{t+h/2} + \frac{h}{2}\mathbf{u}^{t+h}. \tag{52}$$

We use this configuration to update the rotation

$$\boldsymbol{\Lambda}^{t+h} = \hat{\boldsymbol{\Lambda}}\left(\mathbf{q}^{t+h}, \boldsymbol{\Lambda}_1\right). \tag{53}$$

Up until (53) the above scheme mimics the time integrator from Section 4. By experimentation we notice that, when $\mathbf{E}$ includes only the rigid modes, the above scheme becomes strongly dissipative in rotational motion. This is not desirable: if $\mathbf{E}$ is reduced to rigid modes only, we would like to fall back onto an acceptable rigid integrator. We offer a simple remedy for this: we first project $\mathbf{u}^{t+h}$ onto reduced $\bar{\mathbf{u}}^{t+h}$ (overwrite it), and then resolve it back into $\mathbf{u}^{t+h}$ according to the non-incremental formula. This results in correct approximate representation of both rigid and deformable motions.

$$\bar{\mathbf{u}}^{t+h} = \mathbf{E}^T\left(\underline{\boldsymbol{\Lambda}}^{t+h}\right)^T\mathbf{u}^{t+h} \tag{54}$$

$$\mathbf{u}^{t+h} = \underline{\boldsymbol{\Lambda}}^{t+h}\mathbf{E}\bar{\mathbf{u}}^{t+h}. \tag{55}$$

We then compute the final full space and reduced space configurations

$$\mathbf{q}^{t+h} = \mathbf{q}^{t+h/2} + \frac{h}{2}\mathbf{u}^{t+h} \tag{56}$$

$$\bar{\mathbf{q}}^{t+h} = \mathbf{E}^T \left(\underline{\boldsymbol{\Lambda}}^{t+h} \right)^T \left[\left(\underline{\mathbf{I} - \boldsymbol{\Lambda}^{t+h}} \right) \mathbf{Z} + \mathbf{q}^{t+h} \right]. \tag{57}$$

Although the primary merit of developing reduced order models is in reducing the computational time, the smaller storage size is also a welcome consequence. From the above we can see that by storing $\mathbf{E}$, $\boldsymbol{\Lambda}^t$, $\bar{\mathbf{u}}^t$ and $\bar{\mathbf{q}}^t$ one can reproduce $\mathbf{u}^t$ and $\mathbf{q}^t$. Hence the storage size is significantly reduced. Although restoring $\mathbf{q}^t$ from $\bar{\mathbf{q}}^t$ is not used in the above scheme, when reading data the following procedure is used

$$\mathbf{q}^t = \underline{\boldsymbol{\Lambda}}^t \mathbf{E} \bar{\mathbf{q}}^t - \left(\underline{\mathbf{I} - \boldsymbol{\Lambda}^t} \right) \mathbf{Z}. \tag{58}$$

This is acceptable for the case of small co-rotated strains where in the initial (not co-rotated) configuration the velocities and the displacements belong to the same linear space.

6. Handling constraints

In order to compute the next step velocity a nonlinear system needs to be solved

$$\mathbf{A} \left(\mathbf{u}^{t+h} - \mathbf{u}^t \right) = \mathbf{b} + \mathbf{H}^T \mathbf{R} \tag{59}$$

$$\mathbf{U} = \mathbf{H} \mathbf{u}^{t+h} \tag{60}$$

$$\mathbf{C} \left(\mathbf{U}, \mathbf{R} \right) = \mathbf{0}. \tag{61}$$

A variety of constraints can be described in the form (61). These include frictional contact and a range of equality constraints. A fixed point constraint for example reads

$$\mathbf{C}_\alpha \left(\mathbf{U}, \mathbf{R} \right) = \mathbf{U}_\alpha \tag{62}$$

where $\mathbf{U}_\alpha$ is the 3-component local velocity of a point indexed by α. An example of solving the system (59-61) in case of frictional contact constraints can be found in [6, 11, 8], while a broader overview can be found in [5, 1]. In the examples section below we use the frictional contact formulation detailed in [6, 9].

In practice it is sometimes convenient to reformulate the algebra of (59-60) so that only constraints space variables are used. This is done by eliminating $\mathbf{u}^{t+h}$

$$\mathbf{U} = \mathbf{H} \mathbf{A}^{-1} \left(\mathbf{b} + \mathbf{H}^T \mathbf{R} \right) \tag{63}$$

or

$$\mathbf{U} = \mathbf{W} \mathbf{R} + \mathbf{B}. \tag{64}$$

$\mathbf{W}$ is a symmetric semi-positive definite operator composed of dense 3×3 blocks. It's sparsity structure depends on topological connectivity of constraints. For typical multibody problems the number of constraints usually exceeds the number of available kinematic freedom. In such cases constraint solvers usually require many iterations (e.g. of a nonlinear Gauss-Seidel scheme) to achieve reasonable accuracy. In many such cases it is more efficient to explicitly compute $\mathbf{W}$. In our case we have either

$$\mathbf{W} = \mathbf{H} \underline{\boldsymbol{\Lambda}}_1 \mathbf{A}_0^{-1} \underline{\boldsymbol{\Lambda}}_1^T \mathbf{H}^T \tag{65}$$

where we should first compute $\mathbf{H}_1 = \mathbf{H} \underline{\boldsymbol{\Lambda}}_1$ and then $\mathbf{H}_1^T \mathbf{A}_0^{-1} \mathbf{H}_1^T$, using a fixed factorization of $\mathbf{A}_0^{-1}$; or we have

$$\mathbf{W} = \bar{\mathbf{H}} \bar{\mathbf{A}}^{-1} \bar{\mathbf{H}}^T \tag{66}$$

where $\bar{\mathbf{H}} = \mathbf{H} \underline{\boldsymbol{\Lambda}}_1 \mathbf{E}$ is small and dense, while $\bar{\mathbf{A}}^{-1}$ can be precomputed and stored as an explicit inverse.

7. Impacts, friction and energy

In the next section we present several examples involving impacts and friction. To model impacts we use the velocity level Signorini condition

$$U_N^{t+h} \geq 0, \; R_N \geq 0, \; U_N^{t+h} R_N = 0 \tag{67}$$

where U_N^{t+h} is the normal component of the relative contact point velocity, and R_N is the normal component of the reaction impulse. To model friction we use Coulomb's law

$$\begin{cases} \|\mathbf{R}_T\| \leq \mu R_N \\ \|\mathbf{R}_T\| < \mu R_N & \Rightarrow \mathbf{U}_T^{t+h} = \mathbf{0} \\ \|\mathbf{R}_T\| = \mu R_N & \Rightarrow \exists_{\lambda \geq 0} \mathbf{U}_T^{t+h} = -\lambda \mathbf{R}_T \end{cases} \tag{68}$$

where $\mathbf{U}_T$ is the tangential component of the relative contact point velocity, $\mathbf{R}_T$ is the tangential component of the reaction impulse, and μ is the Coulomb's coefficient of friction. The above two laws can be rewritten as $\mathbf{C}(\mathbf{U}, \mathbf{R}) = \mathbf{0}$ [6], as used in the previous sections. Such combined interface law should not inject energy into a multibody system. An analysis of energetic consistency aims at ensuring that the increment of the total system energy is semi-negative

$$\mathbf{E}_{tot}^{t+h} - \mathbf{E}_{tot}^{t} \leq 0. \tag{69}$$

As already mentioned in Section 3, our integration schemes do not guarantee energy conservation and hence ensuring the above is not possible. At most we can show that conditions (67) and (68) do not additionally inject energy into the integrated system. Assuming no external loading, in the presence of contacts the total energy increment can be approximated as

$$\mathbf{E}_{tot}^{t+h} - \mathbf{E}_{tot}^{t} \simeq \triangle W \tag{70}$$

where $\triangle W$ is an increment of the work of the contact reactions. In our time integration we can start by making the first half-step

$$\mathbf{q}^{0+h/2} = \mathbf{q}^0 + \frac{h}{2} \mathbf{u}^0 \tag{71}$$

and then continue with updates

$$\mathbf{q}^{t+3h/2} = \mathbf{q}^{t+h/2} + h\mathbf{u}^{t+h}. \tag{72}$$

A configuration increment reads then

$$\triangle \mathbf{q} = \mathbf{q}^{t+3h/2} - \mathbf{q}^{t+h/2} = h\mathbf{u}^{t+h}. \tag{73}$$

If we transform this increment into the contact points space

$$\triangle \mathbf{Q} = \mathbf{H} \triangle \mathbf{q} = h\mathbf{U}^{t+h} \tag{74}$$

we are able to see that contact forces produce the following work increment

$$\triangle W = h \left\langle \mathbf{U}^{t+h}, \mathbf{R} \right\rangle = \triangle W_T + \triangle W_N = h \left\langle \mathbf{U}_T^{t+h}, \mathbf{R}_T \right\rangle + h \left\langle U_N^{t+h}, R_N \right\rangle. \tag{75}$$

Since the Coulomb's law ensures $\triangle W_T \leq 0$ and the Signorini condition gives $\triangle W_N = 0$, we and up with $\triangle W \leq 0$, so that conditions (67) and (68) are dissipative.

8. Examples

We present three examples. The first example (Rotating bar) overviews the most basic properties of the proposed formulations. The second example (Pipe impact) demonstrates performance on a simple impact problem. The third example (Array excitation) applies the proposed schemes to a structural multi-body setup where some experimental results are available.
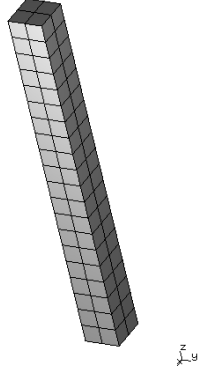
Figure 1: Rotating bar: geometry of the $2 \times 2 \times 20$ mesh.

8.1. Rotating bar

A prismatic bar with x, y, z dimensions $0.1 \times 0.1 \times 1$ m and material parameters $E \in \{200\text{E}4 \text{ Pa}, 200\text{E}9 \text{ Pa}\}$, $\nu = 0.26, \rho = 7.8\text{E}3$ kg $/\text{m}^3$ is rotating about the x-axis with an initial angular velocity of 1 rad/s. We use the smaller Young's modulus in most demonstrations since in this case the time scales of the rigid and deformable motions are not far apart. For $E = 200\text{E}9$ Pa the material parameters correspond to steel. In this case the time scale of elastic deformations is much faster then that of the rigid rotation. We use this material to test performance of the proposed integrators when applied with large time steps. In what follows, unless stated otherwise, the value $E = 200\text{E}4$ is used by default.

The bar is discretized with $2 \times 2 \times 20$ fully integrated trilinear hexahedral elements, Figure 1. This results in 576 degrees of freedom for the full order model. For the reduced order BC-RO approach we used 100 co-rotated displacement samples from the Total Lagrangian solution (with rigid rotation factored out) and 6 rigid motion modes (generated algebraically) as input for the Python modred package, and calculated 11 Proper Orthogonal Decomposition modes. In the current example, we also include the approach from Appendix B, for which the 6 rigid modes where combined with deformable modes 14, 19, 26, 34, 39 (of the initial linearized system), picked by hand as visually corresponding to the predominantly longitudinal deformation of the bar. Hence, the BC-MODAL case also had 11 degrees of freedom.

We first compare the solutions for damped and undamped cases. Figure 2 illustrates the time history of the elongation of the bar, measured between the top and bottom center nodes, for undamped 1s runs with time steps $h \in \{1/64\text{s}, 1/256\text{s}\}$. It can be seen that the results compare well between the formulations proposed here and the Total Lagrangian approach integrated in an analogous manner. A damped case is illustrated in Figure 3 for the large time step of $h = 1/64\text{s}$. Also in this case the agreement is good. The Total Lagrangian based solution deviates from the co-rotational solution since for the TL the stiffness matrix magnitude changes with configuration, which affects the stiffness proportional damping. We can conclude the 5 deformable modes used by the BC-RO and BC-MODAL approaches represent well the longitudinal oscillation accompanying the rotational motion of the bar.

Figure 4 illustrates the time history of the total energy of the bar for 100s runs with time steps $h \in \{1/64\text{s}, 1/256\text{s}\}$. For all but the BC-RO approach, the small oscillation of energy remains bounded with time. The full space co-rotational BC approach is better behaved than BC-MODAL: in nearly coincides with the Total Lagrangian approach. The relatively small oscillation of energy, present in TL, BC, and BC-MODAL cases, is characteristic for the adopted time stepping and it has been also demonstrated by Zhang and Skeel [13] on simpler model problems. The larger oscillation and dissipation of energy, clear in case of the BC-RO approach, remains a shortcoming and it may be addressed in future revisions of this paper.

The second order convergence rate is illustrated in Figure 5. Two cases are considered: undamped motion of a soft bar $E = 200\text{E}4$ and damped motion of stiff bar $E = 200\text{E}9$. In both cases the reference solution q_{ref} was obtained at time $t = 1/2^4\text{s}$ with time step $h = 1/2^{16}\text{s}$. For the softer material the time scales of the rigid and deformable motion are not far apart and hence it is easier to observe the
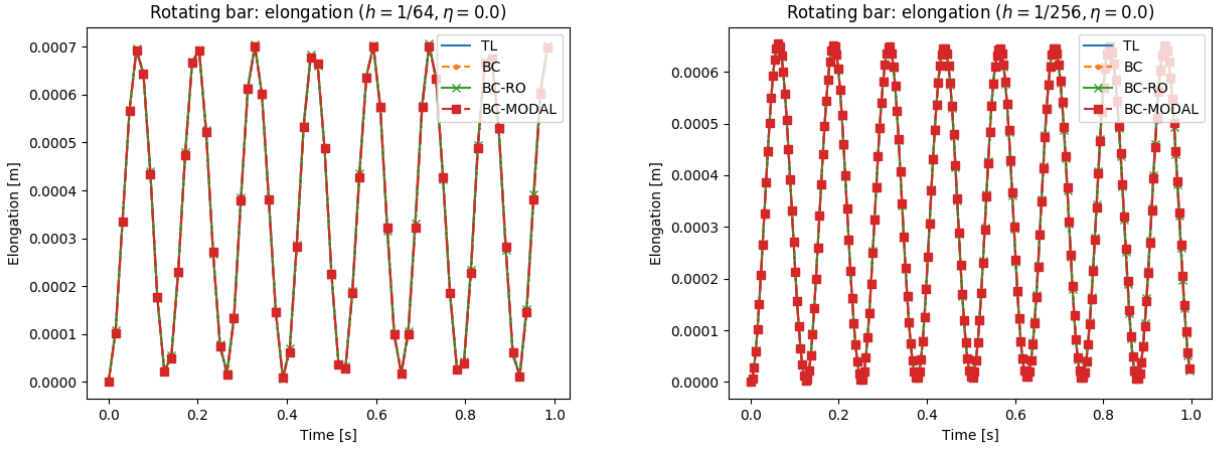
Figure 2: Undamped rotating bar: time history of the elongation for $h \in \{1/64\text{s}, 1/256\text{s}\}$ and 1s runs.
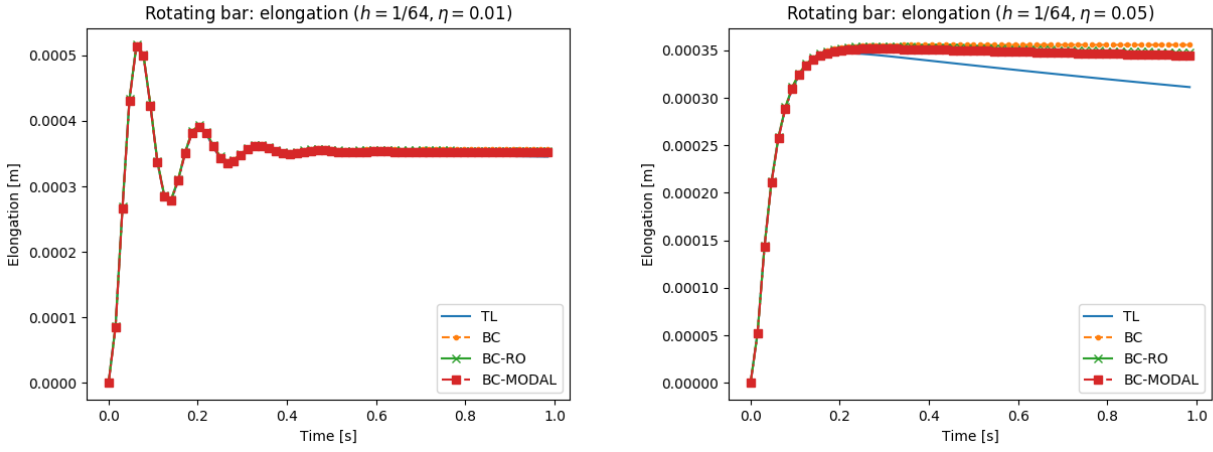

Figure 3: Damped rotating bar: time history of the elongation for $h = 1/64\text{s}$, $\eta \in \{0.01, 0.05\}$ and 1s runs.
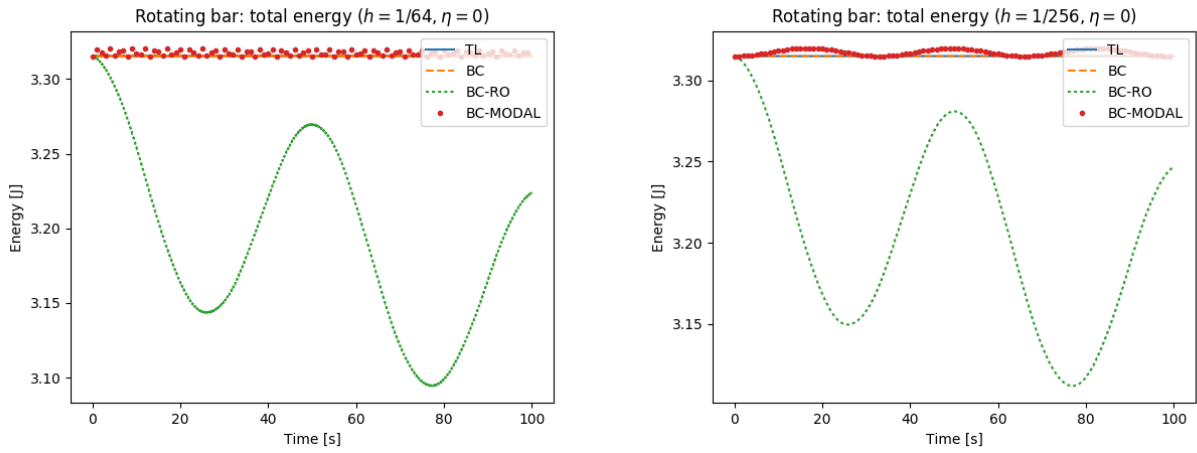

Figure 4: Rotating bar: time history of total energy for $h \in \{1/64\text{s}, 1/256\text{s}\}$ and 100s runs.
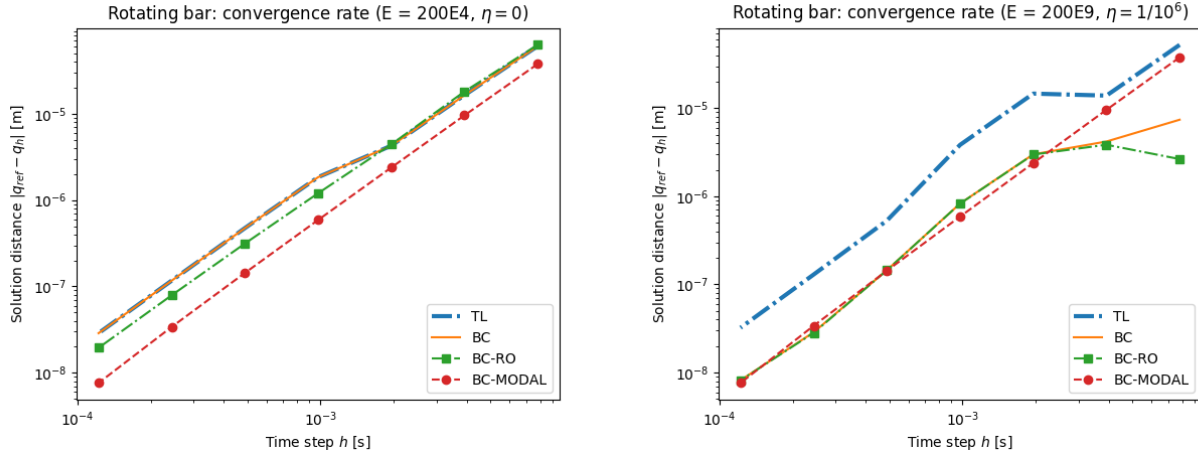
Figure 5: Rotating bar: second order convergence of displacements for soft undamped ($E = 200E4$, $\eta = 0$) and stiff damped ($E = 200E9$, $\eta = 1/10^6$) materials.
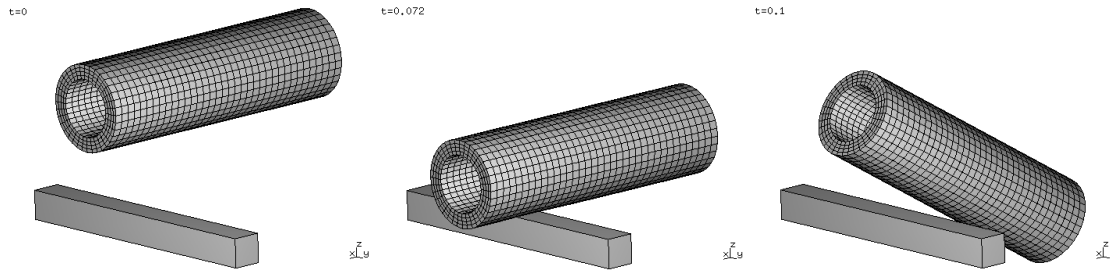
Figure 6: Pipe impact: initial, impact, and final configurations (left to right).

second order convergence for larger time steps. For the stiffer model we used damping $\eta = 1/10^6$ in order to damp out the high frequency oscillations that could not be well represented for the large time steps used here.

8.2. Pipe impact

An elastic pipe free falls onto a block obstacle. Figure 6 illustrates the initial, impact and final configurations. The inner pipe radius is 0.01 m, its thickness is 0.005 m, and its length is 0.1 m. The pipe lowest point is initially located 0.025 m above the top surface of the block. Figure 7 shows the characteristic points, used to extract results. Point p0 impacts the geometrical center of the block. Points p0, p1, p2, p3 are 0.025 m away from p4, when measuring along the pipe length. The pipe is made of $36 \times 36 \times 4$ fully integrated trilinear hexahedral elements. The elastic properties of the pipe are $E = 1E7$ Pa, $\nu = 0.25$ and its density is $\rho = 1E3$ kg /m^3. The obstacle has dimensions $0.01 \times 0.1 \times 0.01$ m and it is modeled as a geometrical boundary which does not contribute mass. The gravity along the z-direction is -10 m/s^2. The simulation time step is 0.001 s, the duration is 0.1 s, and the amount of damping is $\eta = 1E-4$. Schemes TL, BC, and BC-RO are compared. 100 co-rotated displacement samples from the Total Lagrangian solution and 6 rigid motion modes are used as input for the Python modred package, to produce 18 Proper Orthogonal Decomposition modes for the BC-RO approach.

Figure 8 illustrates the energy balance: in all cases the graphs coincide relatively well. While the BC approach closely follows the Total Lagrangian approach (TL), the reduced order BC-RO approach slightly deviates in its post-impact response. Similar general response characteristic can be seen in Figures 9, 10, 11 where other types of results are compared: we have some variation in the post-impact response, while overall the results match. We note, that the rotation angle in Figure 9 was calculated using
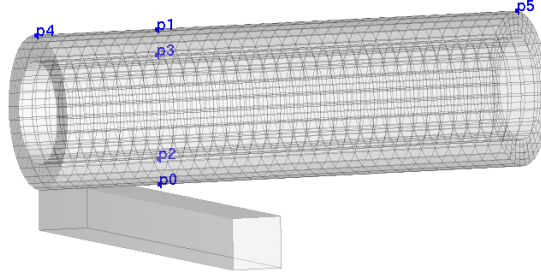
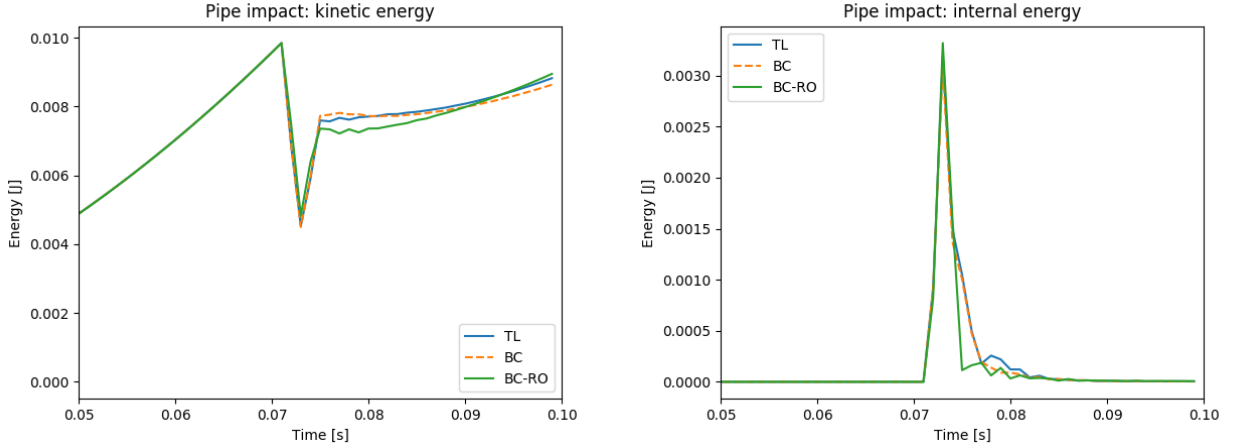Figure 7: Pipe impact: mesh characteristic points.

Figure 8: Pipe impact: kinetic (left), and internal energy between $t = 0.05$s and $t = 0.1$s.

$$\text{angle} = \arcsin\left(\left|\frac{\text{p5} - \text{p4}}{|\text{p5} - \text{p4}|} \times (-1, 0, 0)\right|\right).$$

Table 1 illustrates the runtime and storage statistics. The body co-rotational (BC) approach is about 6 times faster than TL; the reduced order (BC-RO) approach is about 50 time faster and requires about 50 times less storage.

8.3. Array excitation

The experiment behind this example comes from the context of the civil nuclear power generation in the UK. A series of dynamic tests were carried out by National Nuclear Corporation in 1985 as part of the seismic endorsement of the Heysham II / Torness AGR[1] core design. These tests used an array of 81 graphite bricks as shown in Figure 12 (left), consisting of alternating 'fuel' and 'interstitial' bricks. The array was mounted in a rigid frame on a shaker table. The boundary frame was driven in the horizontal plane with synthesized seismic or swept-sine input motions in either one or both axes simultaneously. Measurements were taken of the resulting velocities of certain bricks in the array (determined from accelerometer data) and also of the forces within the keying system (using load cells

---

[1]AGR stands for an Advanced Gass-cooled Reactor

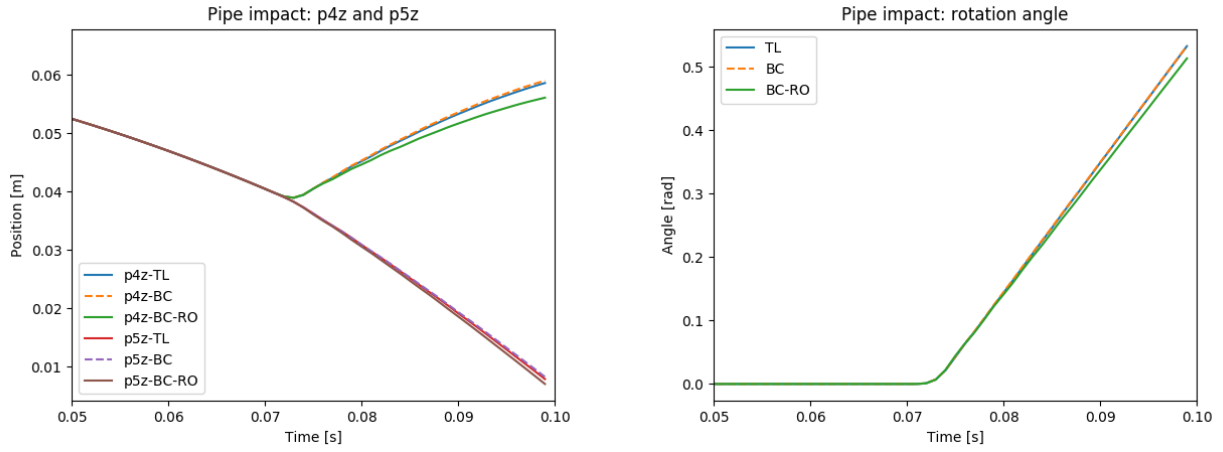| Formulation	TL	BC	BC-RO
Runtime [s]	101	16.2	2.0
Storage [MB]	13	13	0.264

Table 1: Pipe impact: runtime and storage comparison.

Figure 9: Pipe impact: z-coordinates of points p4, p5 (left), and pipe rotation angle (about y) between $t = 0.05$s and $t = 0.1$s.
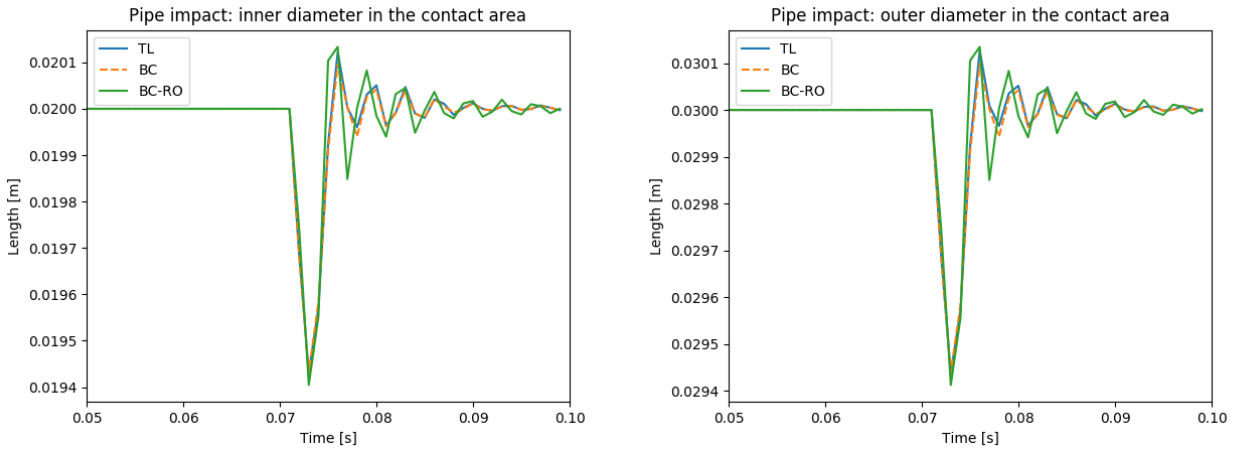

Figure 10: Pipe impact: inner $|p2 - p3|$ (left), and outer $|p0 - p1|$ pipe diameter in the contact area between $t = 0.05$s and $t = 0.1$s.

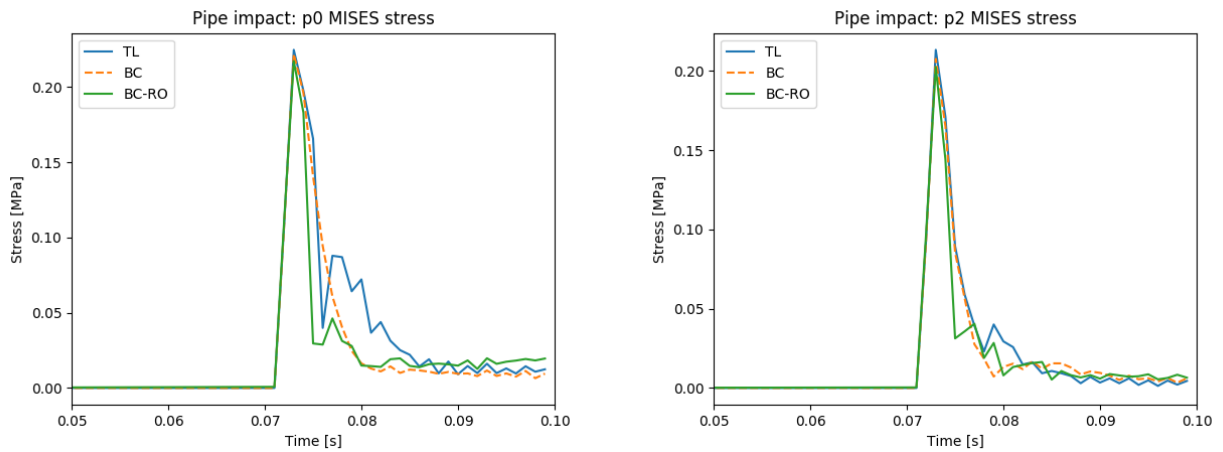Figure 11: Pipe impact: inner point p0 (left), and outer point p2 MISES stress between $t = 0.05$s and $t = 0.1$s.
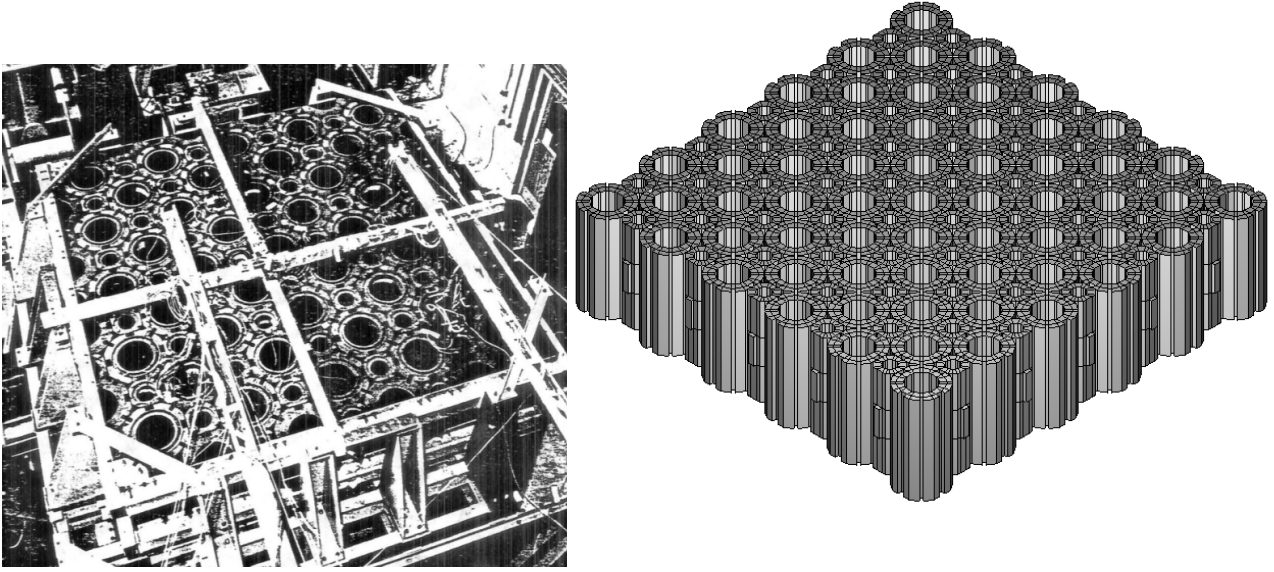
Figure 12: 81 bricks array: experimental setup (left) and meshed model (right: base mesh).
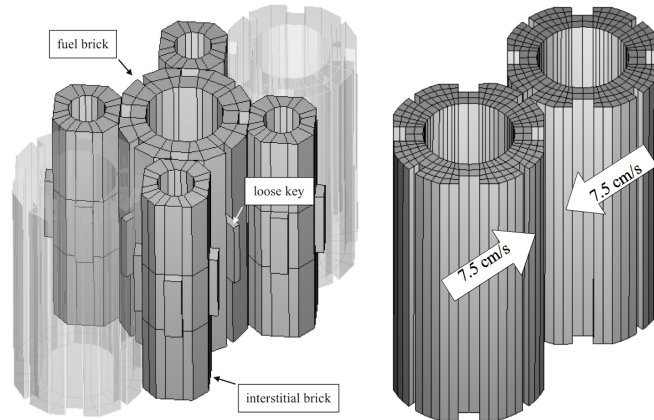

Figure 13: 81 bricks array: array module closeup (left: base mesh) and the binary impact test (right: fine mesh).

built into some components). Some details of the experimental setup has been described in [3, 2]. We note, that this example is also called the "81 bricks array", or simply the "81 array", because the total number of unconstrained fuel and interstitial bricks is 81.

In the current demonstration we bypass many of the details of our mechanical model. The ultimate aim is modeling the entire AGR core ($\sim$ 5K fuel bricks), hence we are in the search for the lowest possible resolution model. We are going to demonstrate some results for two mesh densities: denoted as a "basic mesh" and a "fine mesh", cf. Table 2. The basic meshed model is presented in Figure 12 (right). As in the experiment all bodies are vertically restrained so that they move in a frictionless manner in the horizontal plane. Since the behavior of the assembly is essentially two dimensional the number of elements along the height is kept to minimum. In Figure 13 one can see that the number of elements used in the horizontal brick sections is also small. For fuel brick meshes the error in the lowest eigenmodes corresponding to planar squeezing is below 35% (as compared with finer meshes), which we tentatively accept in the quest for a lowest possible resolution model.

The following material parameters were assumed for the graphite used in the experiment: 1.18E10Pa for Young's modulus, 0.21 for Poisson's ratio, and for mass density 1688kg/m^3, 1637kg/m^3, 1740kg/m^3 for respectively fuel bricks, interstitial bricks and loose keys (the mass densities were tuned to obtain

			Fuel brick	Interstitial brick	Loose key	Model total
BC	base mesh	DOFs	432	456	24	56112
	fine mesh		1512	456	24	121992
BC-RO	base mesh		64	64	24	10144
	fine mesh		64	64	24	10144

Table 2: 81 bricks array: base and fine mesh statistics. There are 61 fuel bricks, 60 interstitial bricks, and 100 loose keys in each case.

the total brick weights reported in the experiment). The coefficient of Coulomb's friction was assumed 0.1. Among several simple impact tests performed in the preliminary experimental work, a binary impact between fuel bricks was pursued: Figure 13 (right). A range of relative input velocities was used (5-35 cm/s) for which the ratio of the output velocity to the input velocity remained consistently in the area of 0.9. It should be noted that due to geometrical setup, this particular type of impact is not present in our full array model. We then have two independent experiments, which we try to validate simultaneously.

We use three families of models: solely rigid (RG), body co-rotational (BC), and reduced order (BC-RO). In all cases the time step is 0.0001s. Co-rotated displacements from the BC simulation were sampled at 0.02s intervals and together with the 6 algebraically generated rigid modes, for each distinct mesh instance, served as an input for the Python modred package, to produce POD bases for the BC-RO approach: 64 modes were used for both the fuel and the interstitial bricks. The loose keys were modeled as single elements based on the co-rotated BC approach. For the deformable models we used damping $\eta = $ 1E-7, which, for the finer mesh, roughly reproduced the required macroscopic velocity restitution of 0.9 for the binary impact test from Figure 13 (right): restitution of 0.91 was produced by the BC approach, and 0.86 by BC-RO. For the basic mesh, the reproduced coefficient of restitution was 0.85 for the BC approach, and 0.82 for BC-RO. In case of the rigid body model the impact restitution was zero, resulting in a totally passive response.

In the initial validation of the entire array model we aimed at reproducing a swept-sine constant amplitude acceleration test. In the experiment the array was subject to a 3s 3Hz settling dwell at start, followed by a 3Hz to 10Hz sweep with 0.1Hz/s buildup rate and constant amplitude of 0.3g, all this amounting to the total duration of 72s. The sweep direction was aligned with one of the sides of the array. Linear velocity histories of the centre points of several bricks were recorded at 50Hz. Magnitudes of these velocities were then averaged using 0.1Hz window and plotted as time series normalized by the corresponding magnitudes of the input velocity.

The input acceleration signal produces a smoothly decaying envelope of displacement, which given a fixed amount of clearance between bricks, initially builds up their "rattling" interactions and then passes a threshold beyond which interactions cease. Initially the displacements are much larger than the clearance and the entire array is swept back and forth. When the boundary displacements reach the level of the clearance (few mm) the kinetic energy starts being injected into higher modes of the system. This builds up a peak in the velocity response. When the input displacements fall below the clearance distance the bricks disengage and there is a drop in the velocity response.

Figures 14-17 show time histories of normalized output velocity magnitudes for selected fuel and interstitial bricks, compared with those obtained in the experiment. One can see that relatively good agreement is obtained for all combinations of base and fine FE meshes combined with the body co-rotational (BC) and reduced order BC-RO (dented as RO in the figures) formulations. The best overall match seems to be obtained in case of the BC-fine model: combining the body co-rotational formulation (BC) and the fine mesh (where only the fuel bricks are refined). This is specifically visible in Figures 14 and 17 where the performance of the BC-fine model for the FB2(2)(2) fuel brick is better than of the BC-base and RO-fine models. There seems to be some correlation between the ability to reproduce the 0.9 binary impact velocity restitution from the experiment, and the ability to match most of the experimental curves in Figures 14-17. On the other hand, Figure 18 compares the performance of the BC-base approach with that of the purely rigid body model (RG). Clearly, the rigid body approach is not able to reproduce the experimentally observed peaks.
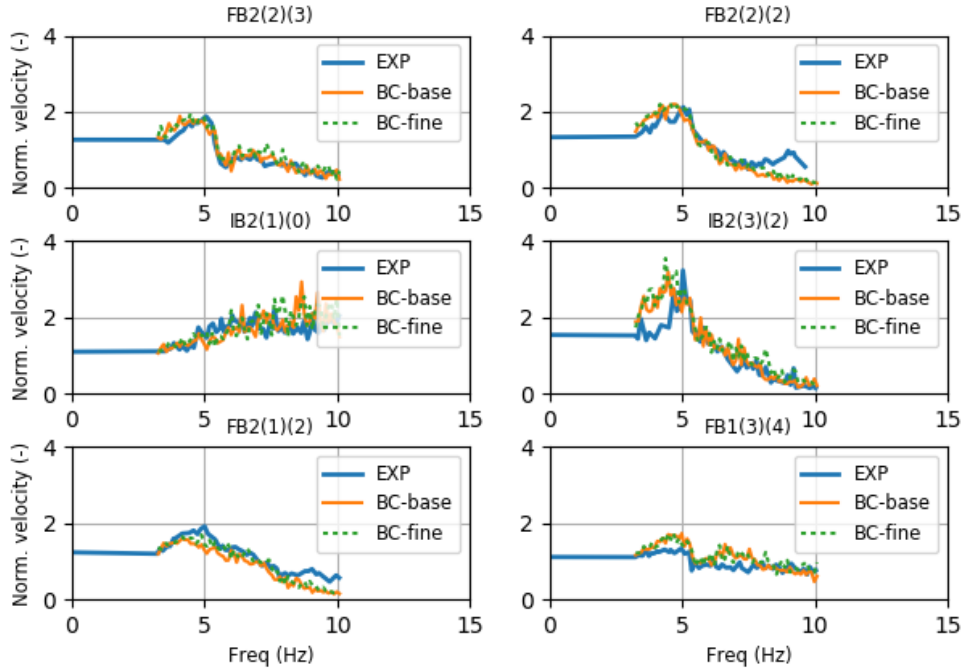
Figure 14: 81 bricks array: frequency histories of normalized output velocity magnitudes. EXP is the experiment. BC-base and BC-fine denote the body co-rotational formulation using respectively the base and the fine mesh.
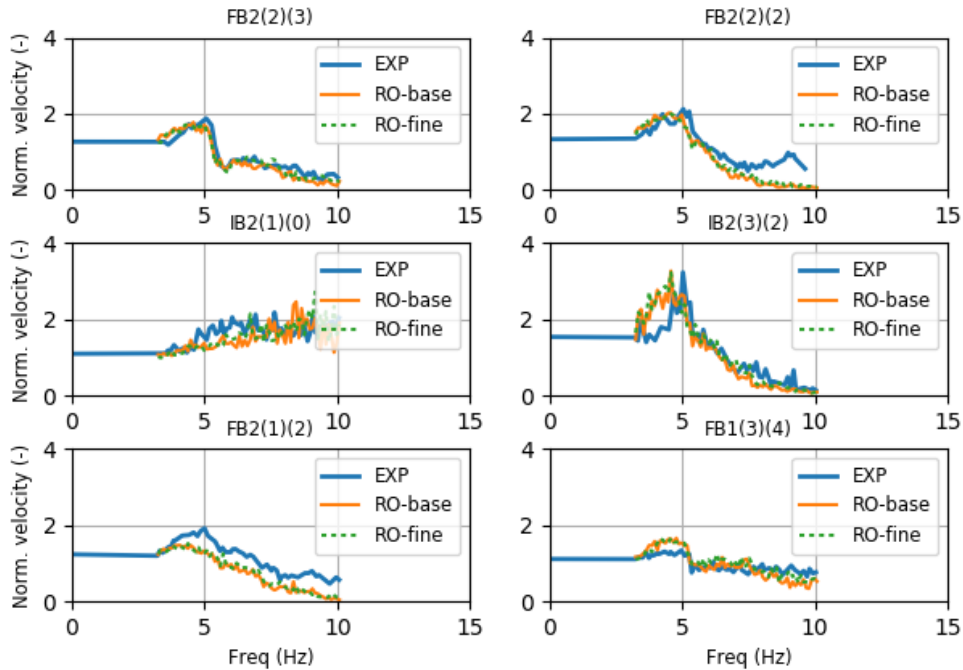

Figure 15: 81 bricks array: frequency histories of normalized output velocity magnitudes. EXP is the experiment. RO-base and RO-fine denote the reduced order formulation using respectively the base and the fine mesh.

Figure 16: 81 bricks array: frequency histories of normalized output velocity magnitudes. EXP is the experiment. BC-base and RO-base are respectively the BC and the BC-RO formulations, both using the base mesh.

Figure 17: 81 bricks array: frequency histories of normalized output velocity magnitudes. EXP is the experiment. BC-fine and RO-fine are respectively the BC and the BC-RO formulations, both using the fine mesh.
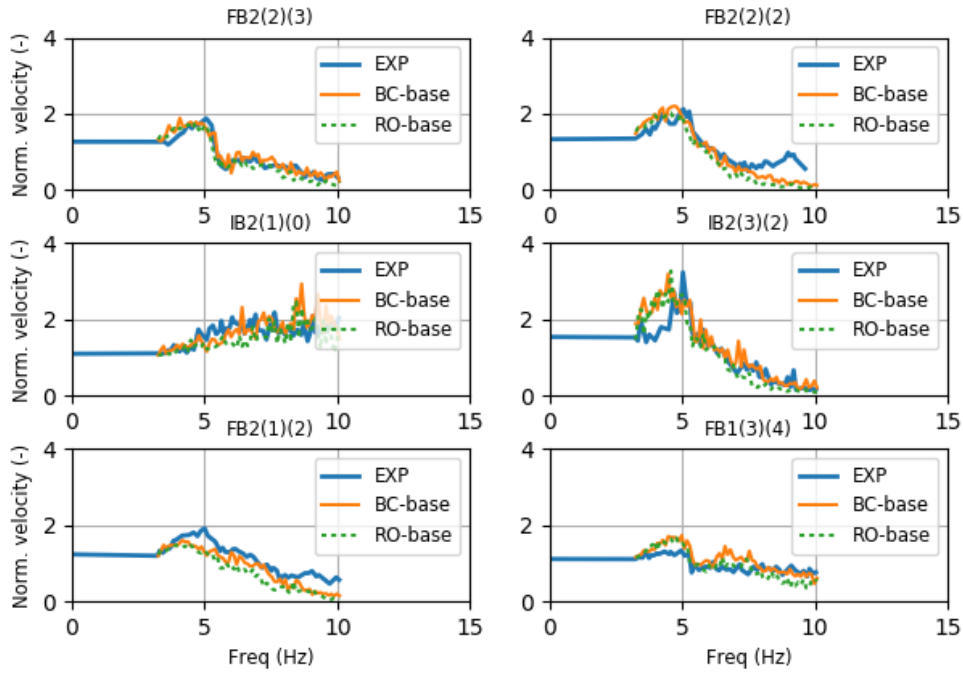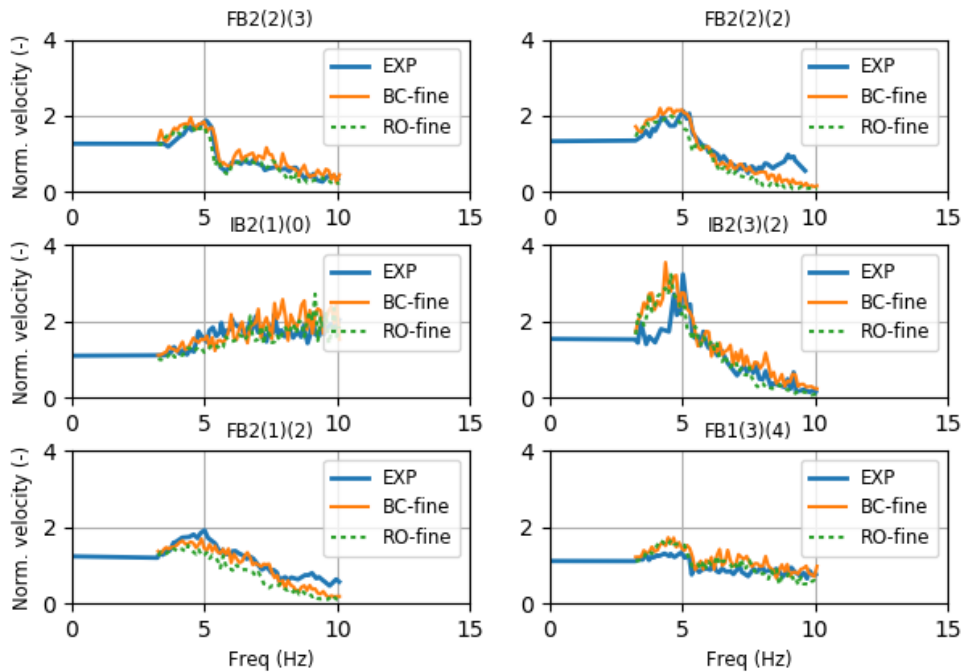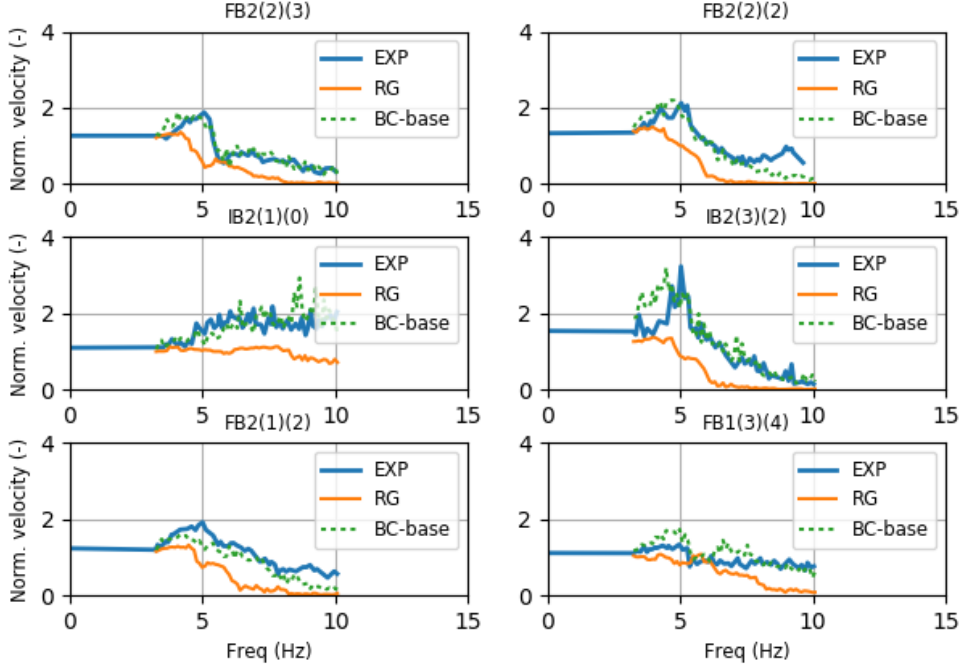
Figure 18: 81 bricks array: frequency histories of normalized output velocity magnitudes. EXP is the experiment. RG and BC-base are respectively the rigid body and the body co-rotational formulations, both using the base mesh.

	Formulation		BC-fine	BC-base	RO-fine	RO-base	Rigid body
Comp. stage	Time integration	(h)	4.6	2.1	3.4	1.7	0.1
	Contact detection		0.9	0.5	0.8	0.5	0.4
	Constraints equations		13.7	5.7	4.0	2.9	0.2
	Constraints solution		0.7	0.9	1.2	1.3	3.4
	Load balancing		1.3	0.7	1.0	0.7	0.3
Total runtime (h)			21.18	9.92	10.53	7.15	4.37

Table 3: 81 bricks array: comparison of runtimes, for various formulations, using 6 MPI ranks on a single Intel Xeon E5–2600 CPU based compute node.

Statistics of total runtimes and average time shares of computational stages are included in Table 3. These are collected for BC-fine/base, RO-fine/base and the rigid body models, all run using 6 MPI ranks on a single compute node, equipped with Intel Xeon E5–2600 processing units. There is some "give and take" happening in terms of the share of the computational time taken by various operations. The time integration runtime, which is significant for the deformable models, for RO-fine/base is shorter compared to the fully resolved BC-fine/base models. Assembling of constraint equations, which dominates the computational time, also takes less time in case of RO-fine/base, when compared to BC-fine/base. In the deformable model cases contact solution does not dominate the total runtime. This reversed in case of the solely rigid model, where solving the ill-conditioned constraint equations dominates. Table 4 illustrates parallel scaling, from 3 to 24 MPI-ranks, on a 24 core cluster node equipped with two Intel Xeon E5–2600 CPUs. Maximum speedups are 3.43/2.49 for the BC/RO-base models and 3.99/2.57 for the BC/RO-fine models. We note that currently Solfec does not exploit shared memory parallelism. The domain decomposition based load balancing may not be a most suitable parallelization strategy for a problem of this size, run on a single cluster node. Table 5 illustrates the size of the output storage for the tested approaches. Naturally, the reduced order models output less, compared to the fully resolved models.

MPI ranks	3	6	12	24
BC-base-72s runtime (h)	17.81	9.92	6.90	5.20
RO-base-72s runtime (h)	11.94	7.15	5.56	4.79
BC-fine-7.2s runtime (h)	4.31	2.31	1.52	1.08
RO-fine-7.2s runtime (h)	2.03	1.19	0.97	0.79

Table 4: 81 bricks array: comparison of runtime scaling, for various formulations, on a single Intel Xeon E5–2600 CPU based compute node. BC-base-72s and RO-base-72s used the base mesh and had 72s duration. BC-fine-7.2s and RO-fine-7.2s used the fine mesh and had a 10x shorter duration of 7.2s.

Formulation	BC-fine	BC-base	RO-fine	RO-base	Rigid body
Storage size	7.2GB	3.7GB	1.2GB	1.2GB	0.84GB

Table 5: 81 bricks array: comparison of storage size, for various formulations, for the 72s long runs and using the 50Hz file output frequency.

9. Conclusions

This paper describes the finite element formulations and time integrators, implemented in Solfec, and available as "DEF_LIM" integrator choice, and "TL", "BC", "BC-RO" and "BC-MODAL" formulation choice, for the Solfec's input BODY command. These formulations allow to vary the amount of deformation expressed by a finite element mesh and facilitate saving computational time and storage. They may be helpful in the context of multibody modeling.

A. Appendix - Updating rotation

Kaczmarczyk et al. [7] developed a convenient way of updating rotation, as needed in (25), and (32). Because the antisymmetric part of the gradient of the deformational displacement (9) corresponds to an infinitesimal rotation at a point, a sum

$$\overline{\nabla \mathbf{d}} = \int_{\Omega} \nabla \mathbf{d} \left(\mathbf{X}, t \right) - \nabla \mathbf{d}^T \left(\mathbf{X}, t \right) d\Omega \tag{76}$$

represents the resultant infinitesimal rotation of the body, where the $1/2$ term, being non-consequential to our derivation, has been skipped. For every $\mathbf{d}$, $\mathbf{\Lambda}$ is incremented in such a way so to minimise $\overline{\nabla \mathbf{d}}$. The divergence theorem is exploited in [7] in order to obtain

$$\text{vec} \left[\int_{\Omega} \nabla \mathbf{d} - \nabla \mathbf{d}^T d\Omega \right] = \int_{\Gamma} \mathbf{n} \times \mathbf{d} d\Gamma \tag{77}$$

where $\text{vec}[\cdot]$ makes a pseudo-vector out of an antisymmetric matrix, and $\mathbf{n}$ is the spatial outward normal to the surface Γ of the body Ω. The surface integral can be further resolved

$$\begin{aligned}
\mathbf{h} &= \int_{\Gamma} \mathbf{n} \times \mathbf{d} d\Gamma = \int_{\Gamma} \text{skew} \left[\mathbf{n} \right] \mathbf{d} d\Gamma \\
&= \int_{\Gamma} \text{skew} \left[\mathbf{\Lambda} \mathbf{N} \right] \mathbf{d} d\Gamma = \int_{\Gamma} \text{skew} \left[\mathbf{N} \right] \mathbf{\Lambda}^T \mathbf{d} d\Gamma \\
&= \int_{\Gamma} \text{skew} \left[\mathbf{N} \right] \mathbf{\Lambda}^T \left[\mathbf{X} + \mathbf{S} \left(\mathbf{X} \right) \mathbf{q} - \mathbf{\Lambda} \mathbf{X} \right] d\Gamma \\
&\simeq \int_{\Gamma_0} \text{skew} \left[\mathbf{N} \right] \mathbf{X} d\Gamma_0 + \int_{\Gamma} \text{skew} \left[\mathbf{N} \right] \mathbf{\Lambda}^T \left[\mathbf{X} + \mathbf{S} \left(\mathbf{X} \right) \mathbf{q} \right] d\Gamma \\
&= \int_{\Gamma} \text{skew} \left[\mathbf{N} \right] \mathbf{\Lambda}^T \mathbf{x} d\Gamma
\end{aligned} \tag{78}$$

where the simplification to the last line follows from the assumption $\int_\Gamma f d\Gamma \simeq \int_{\Gamma_0} f d\Gamma_0$ and from the fact that $\int_\Gamma \text{skew}\left[\mathbf{N}\right]\mathbf{X}d\Gamma = \text{vec}\left[\int_\Omega \nabla\mathbf{X} - \nabla\mathbf{X}^T d\Omega\right] = \mathbf{0}$, where $\mathbf{N}$ is the surface normal field in the reference configuration and Γ_0 is the referential surface. The operator $\text{skew}\left[\cdot\right]$ furnishes a skew symmetric matrix out of a vector so that $\text{skew}\left[\mathbf{n}\right]\mathbf{d} = \mathbf{n}\times\mathbf{d}$. The following functional is then defined

$$\mathbf{J}\left(\mathbf{q},\mathbf{\Lambda}\right) = \frac{1}{2}\mathbf{h}^T\left(\mathbf{q},\mathbf{\Lambda}\right)\mathbf{h}\left(\mathbf{q},\mathbf{\Lambda}\right) \tag{79}$$

and the $\hat{\mathbf{\Lambda}}$ mapping employed in the previous section takes form

$$\hat{\mathbf{\Lambda}}\left(\mathbf{q},\mathbf{\Lambda}\right) = \exp\left(\mathbf{\Phi}\right)\mathbf{\Lambda}, \text{ where } \mathbf{\Phi} = \arg\min\mathbf{J}\left(\mathbf{q},\exp\left(\mathbf{\Phi}\right)\mathbf{\Lambda}\right) \tag{80}$$

where $\exp\left[\cdot\right]$ is the exponential map. Gauss-Newton iterations are exploited in order to compute the minimiser

$$\mathbf{\Phi}^{i+1} = \mathbf{\Phi}^i - \left\{\partial^2\mathbf{J}/\partial\mathbf{\Phi}\partial\mathbf{\Phi}\right\}^{-1}\left\{\partial\mathbf{J}/\partial\mathbf{\Phi}\right\} \tag{81}$$

where

$$\begin{aligned}
\partial\mathbf{J}/\partial\mathbf{\Phi} &= \mathbf{h}^T\left[\partial\mathbf{h}/\partial\mathbf{\Phi}\right] \\
&= \mathbf{h}^T\left[\int_\Gamma \text{skew}\left[\mathbf{N}\right]\mathbf{\Lambda}^T\frac{\partial\exp\left(-\mathbf{\Phi}\right)}{\partial\mathbf{\Phi}}\mathbf{x}d\Gamma\right]
\end{aligned} \tag{82}$$

$$\begin{aligned}
\partial^2\mathbf{J}/\partial\mathbf{\Phi}\partial\mathbf{\Phi} &= \left[\partial\mathbf{h}/\partial\mathbf{\Phi}\right]^T\left[\partial\mathbf{h}/\partial\mathbf{\Phi}\right] + \mathbf{h}^T\left[\partial^2\mathbf{h}/\partial\mathbf{\Phi}\partial\mathbf{\Phi}\right] \\
&\simeq \left[\partial\mathbf{h}/\partial\mathbf{\Phi}\right]^T\left[\partial\mathbf{h}/\partial\mathbf{\Phi}\right] + \delta\mathbf{I}
\end{aligned} \tag{83}$$

and δ is large enough to make $\partial^2\mathbf{J}/\partial\mathbf{\Phi}\partial\mathbf{\Phi}$ positive definite, ensuring a descent direction in (81). In practice, for this 3×3 problem, $\delta = 0$ works well.

B. Appendix - Body co-rotated, modal approach (BC-MODAL)

Let a modal, reduced order base be spanned by the columns of matrix $\mathbf{E}$ so that

$$\mathbf{E}^T\mathbf{M}\mathbf{E} = \text{diag}\left(1\right) \text{ and } \mathbf{E}^T\mathbf{K}\mathbf{E} = \text{diag}\left(\lambda_i\right) \tag{84}$$

and hence $\mathbf{E},\lambda_i$ store selected eigenvectors and eigenvalues of the initial linearized dynamic system. Let the eigenvectors and eigenvalues be sorted, so that the initial six vectors represent purely rigid motion and are accompanied by zero eigenvalues. The scheme from Section 5 remains almost unchanged. The projection formula (40) becomes

$$\bar{\mathbf{q}}^{t+h/2} = \mathbf{E}^T\mathbf{M}\mathbf{\Lambda}_1^T\mathbf{d} \tag{85}$$

where the $\mathbf{M}$-scaling ensures $\langle\bar{\mathbf{q}},\bar{\mathbf{q}}\rangle = \langle\mathbf{E}\bar{\mathbf{q}},\mathbf{M}\mathbf{E}\bar{\mathbf{q}}\rangle$, and the reduced internal force becomes $\bar{\mathbf{f}}_{int}^{t+h/2} = \text{diag}\left(\lambda_i\right)\bar{\mathbf{q}}^{t+h/2}$. Further, the reduced system (43) is replaced by

$$\text{diag}\left(1 + \left(\frac{\eta h}{2} + \frac{h^2}{4}\right)\lambda_i\right)\left(\bar{\mathbf{u}}^{t+h} - \bar{\mathbf{u}}^t\right) = \mathbf{E}^T\underline{\mathbf{\Lambda}}_1^T\left(\mathbf{b} + \mathbf{H}^T\mathbf{R}\right) \tag{86}$$

and the system matrix (45) becomes

$$\bar{\mathbf{A}} = \text{diag}\left(1 + \left(\frac{\eta h}{2} + \frac{h^2}{4}\right)\lambda_i\right). \tag{87}$$

Also, instead of (47) we have

$$\bar{\mathbf{b}} = h \left(\mathbf{E}^T \underline{\boldsymbol{\Lambda}_1}^T \mathbf{f}^{t+h/2} - \mathrm{diag}\left(\lambda_i\right) \left(\bar{\mathbf{q}}^{t+h/2} + \eta \bar{\mathbf{v}}^t \right) \right) \tag{88}$$

and projection (48) is replaced by

$$\bar{\mathbf{v}}^t = \mathbf{E}^T \mathbf{M} \underline{\boldsymbol{\Lambda}_1}^T \mathbf{u}^t. \tag{89}$$

Projection (54) is replaced by

$$\bar{\mathbf{v}}^{t+h} = \mathbf{E}^T \mathbf{M} \left(\underline{\boldsymbol{\Lambda}}^{t+h} \right)^T \mathbf{u}^{t+h} \tag{90}$$

and

$$\{\bar{u}_7, \bar{u}_8, ..., \bar{u}_n\}^{t+h} = \{\bar{v}_7, \bar{v}_8, ..., \bar{v}_n\}^{t+h} \tag{91}$$

so that only deformable components of $\bar{\mathbf{u}}^{t+h}$ are substituted: this improves the long-term energy behavior. Finally, projection (57) is replaced by

$$\bar{\mathbf{q}}^{t+h} = \mathbf{E}^T \mathbf{M} \left(\underline{\boldsymbol{\Lambda}}^{t+h} \right)^T \left[\left(\underline{\mathbf{I} - \boldsymbol{\Lambda}^{t+h}} \right) \mathbf{Z} + \mathbf{q}^{t+h} \right]. \tag{92}$$

The remaining formulas remain as in Section 5.

C. Appendix - Links

Methods presented here are a part of Solfec; Solfec source code and documentation are available at:

- https://github.com/tkoziara/solfec/
- http://parmes.github.io/solfec

Implementation of presented here is included with:

- https://github.com/tkoziara/solfec/blob/master/fem.c

The approach from Section 3 is implemented in functions starting with TL_, e.g.

- https://github.com/tkoziara/solfec/blob/master/fem.c#L1757

The approach from Section 4 is implemented in functions starting with BC_, e.g.

- https://github.com/tkoziara/solfec/blob/master/fem.c#L2007

The approach from Section 5 and Appendix B is implemented in functions starting with RO_, e.g.

- https://github.com/tkoziara/solfec/blob/master/fem.c#L2354

The examples can respectively be found at:

- Rotating bar:
    - Source: https://github.com/tkoziara/solfec/tree/master/examples/reduced-order0
    - Online: http://parmes.github.io/solfec/examples/reduced_order/ro0.html
- Pipe impact:
    - Source: https://github.com/tkoziara/solfec/tree/master/examples/reduced-order1
    - Online: http://parmes.github.io/solfec/examples/reduced_order/ro1.html
- Array excitation:
    - Source: https://github.com/tkoziara/solfec/tree/master/examples/81array
    - Online: http://parmes.github.io/solfec/examples/81array.html

Acknowledgements

Revision notes

Rev. 1 Initial revision;

References

[1] V. Acary and B. Brogliato. *Numerical Methods for Nonsmooth Dynamical Systems*, volume 35 of *Lecture Notes in Applied Lecture Notes in Applied and Computational Mechanics*. Springer Verlang, 2008.

[2] K. M. Ahmed. The dynamic response of multi-layers AGR core brick arrays. *Nuclear Engineering and Design*, 104(1):1–66, 1987.

[3] Khalid Ahmed and Stefan Stojko. The non-linear seismic response of AGR core graphite brick slices – correlation of experimental and analytical results. *Earthquake Engineering & Structural Dynamics*, 15(2):159–188, 1987.

[4] T. Belytschko, W. K. Liu, and B. Moran. *Nonlinear Finite Elements for Continua and Structures*. J. Wiley and Sons, New York, 2000.

[5] Bernard Brogliato. *Nonsmooth Mechanics*. Communications and Control Engineering. Springer Verlang, 1999.

[6] M. Jean. The non-smooth contact dynamics method. *Computer Methods in Applied Mechanics and Engineering*, 177(3-4):235–257, 1999.

[7] Lukasz Kaczmarczyk, Tomasz Koziara, and Chris J. Pearce. Corotational formulation for 3d solids: an analysis of geometricaly nonlinear foam deformation. *http://arxiv.org/abs/1110.5321*, 2011.

[8] Tomasz Koziara and Nenad Bićanić. Semismooth Newton method for frictional contact between pseudo-rigid bodies. *Computer Methods in Applied Mechanics and Engineering*, 197:2763–2777, 2008.

[9] Tomasz Koziara and Nenad Bićanić. A distributed memory parallel multibody contact dynamics code. *International Journal for Numerical Methods in Engineering*, 87(1-5):437–456, 2011.

[10] T.A. Laursen and X.N. Meng. A new solution procedure for application of energy-conserving algorithms to general constitutive models in nonlinear elastodynamics. *Computer Methods in Applied Mechanics and Engineering*, 190(46 - 47):6309 – 6322, 2001.

[11] J. J. Moreau. Numerical aspects of the sweeping process. *Computer Methods in Applied Mechanics and Engineering*, 177(3-4):329–349, 1999.

[12] J. C. Simo and N. Tarnow. The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics. *Zeitschrift für Angewandte Mathematik und Physik (ZAMP)*, 43:757–792, September 1992.

[13] M. Zhang and R.D. Skeel. Cheap implicit symplectic integrators. *Applied Numerical Mathematics*, 25:297–302(6), 1997.