Co-Simulation of Multibody Systems With Contact Using Reduced Interface Models

Albert Peiret
Department of Mechanical Engineering and
Centre for Intelligent Machines,
McGill University,
817 Sherbrooke St. West,
Montréal, QC H3A 0C3, Canada
e-mail: albert.peiret@mail.mcgill.ca

Francisco González
Laboratorio de Ingeniería Mecánica,
University of A Coruña, Mendizábal s/n,
Ferrol 15403, Spain
e-mail: f.gonzalez@udc.es

József Kövecses
Department of Mechanical Engineering and
Centre for Intelligent Machines,
McGill University,
817 Sherbrooke St. West,
Montréal, QC H3A 0C3, Canada
e-mail: jozsef.kovecses@mcgill.ca

Marek Teichmann
CM Labs Simulations,
645 Wellington,
Montréal, QC H3C 1T2, Canada
e-mail: marek@cm-labs.com

Co-simulation techniques enable the coupling of physically diverse subsystems in an efficient and modular way. Communication between subsystems takes place at discrete-time instants and is limited to a given set of coupling variables, while the internals of each subsystem remain undisclosed and are generally not accessible to the rest of the simulation environment. In noniterative co-simulation schemes, commonly used in real-time applications, this may lead to the instability of the numerical integration. The stability of the integration in these cases can be enhanced using interface models, i.e., reduced representations of one or more subsystems that provide physically meaningful input values to the other subsystems between communication points. This work describes such an interface model that can be used to represent nonsmooth mechanical systems subjected to unilateral contact and friction. The dynamics of the system is initially formulated as a mixed linear complementarity problem (MLCP), from which the effective mass and force terms of the interface model are derived. These terms account for contact detachment and stick–slip transitions, and can also include constraint regularization in case of redundancy in the system. The performance of the proposed model is shown in several challenging examples of noniterative multirate co-simulation schemes of a mechanical system with hydraulic components, which feature faster dynamics than the multibody subsystem. Using an interface model improves simulation stability and allows for larger integration step-sizes, thus resulting in a more efficient simulation.
[DOI: 10.1115/1.4046052]

Keywords: co-simulation, multibody system dynamics, contact, multirate, multiphysics, reduced-order model, co-simulation interface

1 Introduction

Predictive simulation of engineering systems is a valuable tool in the development of new products and industrial applications. Improvements in computational power and software capabilities during the latest decades have expanded the range of problems that can be addressed, as well as the expectations about performance and the quality of the results. This is also the case with forward-dynamics simulation of multibody systems. Early multibody system dynamics studies focused on the solution of relatively simple mechanical problems. Presently, multibody simulations are able to deal with challenging phenomena such as flexibility, contacts, and friction in an efficient way [1,2]; in many cases, the interaction of the mechanical system with elements of a different physical nature, like hydraulics and electronics, is also taken into consideration. The techniques to include this interaction in simulation can be categorized into two main groups, namely, monolithic methods and co-simulation approaches.

Monolithic formulations describe the dynamics of all the components in an engineering application with a single set of equations, solved with its corresponding integrator. This approach has been successfully applied to mechatronics and hydraulically actuated mechanical systems [3,4], showing good stability and efficiency properties [5]. Co-simulation, on the other hand, consists in modeling and integrating separately the different subsystems in an engineering application. The dynamics of each can then be formulated and solved using methods especially suited to its physical behavior. The subsystem solvers only exchange information through a reduced set of *coupling variables* at discrete-time communication points; otherwise, the numerical integration of each subsystem proceeds independently from the execution of others. The time interval between communication points is usually referred to as *macrostep*. This modular approach makes co-simulation environments particularly suitable for collaborative projects, as it enables each partner to use its own modeling and solution methods regardless of the implementation chosen by other developers. Moreover, the internal details of each subsystem remain hidden because only the communication interface needs to be accessible from the outside, which avoids the disclosure of intellectual property. Additionally, co-simulation makes it easier to distribute the computational workload between several processing units and to introduce interactions with physical components in the simulation process, as in the case of Human- and Hardware-in-the-Loop (HiL) setups [6]. However, the discrete-time communication between subsystems gives rise to coupling errors, such as discontinuities and time delays in the coupling variables, which may compromise the stability of the integration process and the accuracy of the results [7].

Multiple approaches have been proposed in the literature to improve the stability properties of co-simulation algorithms. In general, iterative coupling schemes have been shown to exhibit a more stable behavior than their noniterative counterparts [8,9]. The stability of implicit methods can be further improved using polynomial interpolations [10] and the partial derivatives of the subsystem states with respect to the coupling variables, if they are available [11]. These schemes update the input variables in each iteration and subsequently retake the integration step of one or more subsystems. In some applications, however, iterative schemes cannot be used, either because certain subsystems do not allow resetting the integration to a previous state, or because the available time to carry out the numerical integration is limited. In these cases, explicit noniterative coupling approaches must be used. When this is the case, Jacobi schemes, where the subsystems exchange inputs at the beginning of a macrostep and proceed with

Contributed by the Design Engineering Division of ASME for publication in the Journal of Computational and Nonlinear Dynamics. Manuscript received May 15, 2019; final manuscript received January 10, 2020; published online February 24, 2020. Assoc. Editor: Paramsothy Jayakumar.

their integration independently until the next communication point, are a common choice. These noniterative co-simulation schemes permit the parallelization of subsystem integration, as opposed to Gauss–Seidel approaches, where the subsystems are simulated in a sequential fashion [9].

Keeping noniterative co-simulation schemes stable is challenging, especially if subsystems have direct feed-through, i.e., their outputs explicitly depend on their inputs [8]. In some applications, it may be difficult to know whether this is the case, because information about the internals of a subsystem may not be available to the rest of the co-simulation framework; the subsystems behave effectively as *black boxes*, accessible only through their communication interfaces. Also, even if the integration process remains stable, coupling errors at the interface may cause the simulation results to be inaccurate and not representative of the behavior of the system under study [12]. Extrapolation and approximation of subsystem inputs are commonly used strategies to improve the performance of noniterative schemes [13–15]; adaptive extrapolation techniques can be used as well [16]. Another possibility is to act on the coupling variables or the communication step-size to maintain the energy balance at the interface [17,18].

The above-mentioned methods only require the information contained in the coupling variables to operate and do not need further knowledge of the subsystems internals. The availability of additional information, however, enables the definition and use of alternative strategies to keep explicit co-simulation schemes stable and accurate. The directional derivatives of the subsystems can be used to this end [19] if they are known; otherwise, it is also possible to estimate them using subspace identification algorithms [20]. If the information exchanged between subsystems includes details about their internal energy balance, energy monitoring and correction algorithms can be employed to enforce the fulfillment of the energy balance of the whole system [21].

Reduced interface models (RIMs) have been recently proposed to enhance the explicit co-simulation of multibody systems in multirate environments [22]. In many engineering applications, multibody systems are often coupled to other subsystems with faster dynamics, such as hydraulics and electronics, which require smaller integration step-sizes. As a consequence, these subsystems perform more than one integration time-step during each global macrostep. The inputs that the multibody subsystem provides to these, however, cannot be updated until the next communication point, and this may result in instabilities and the inaccuracy of the results, even if extrapolation techniques are used. The RIM is intended to provide these faster subsystems with a physics-based prediction of the evolution of their inputs during the macrostep. Such a model is characterized by the interface between the mechanical subsystem and its co-simulation environment, which can be parameterized by a set of generalized velocities that define its interaction with the rest of the system. These generalized interface velocities can be used to define a subspace in the dynamic model, the *interface subspace*, analogous to the subspace of constrained motion [23], in which the dynamics of the whole multibody system can be decomposed. The RIM thus obtained constitutes a reduced-order expression of the dynamics of the mechanical system, represented by an effective mass matrix and an effective force term.

The concept of RIM was introduced in Ref. [22] for smooth mechanical systems and formulated at the acceleration level. The research presented in this paper extends the use of RIMs to nonsmooth mechanical systems, such as those subjected to unilateral contacts, impacts, and dry friction. The dynamics of such systems is often formulated at the impulse-momentum level using time-stepping schemes [24,25]. These methods can handle collisions seamlessly and remove inconsistencies and indeterminacies caused by friction that affect their acceleration-based counterparts [26,27]. Although the system dynamics can be formulated as a *mixed linear complementarity problem* (MLCP) for the frictionless case [28], considering Coulomb friction at the contact points leads to *nonlinear complementarity problems* (NCP). Some

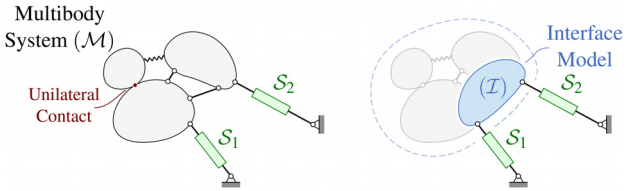

Fig. 1 Interfacing a multibody system $\mathcal{M}$ with a subsystem $\mathcal{S}$ using an interface model $\mathcal{I}$

techniques from mathematical optimization can be used to solve such nonlinear problems [29]. Nevertheless, it is possible to discretize the Coulomb model via the so-called facetization of the friction cone so that the formulation recovers the form of an MLCP [24–26,30]. Formulating the dynamics as an MLCP is convenient because a large array of solver algorithms for such a problem is available in the literature [31–33]. Alternatively, cone complementarity methods formulate the dynamics as a convex optimization problem and are often used in large-scale multibody systems (such as in granular material simulation). On the other hand, the contact problem is nonconvex by nature and the so-called convexification is used in these methods to guarantee the solution and convergence of the problem. But this can lead to artifacts when the relative sliding velocity in a contact pair is not small [29,34].

This paper develops the RIM for nonsmooth mechanical systems with contact and friction, whose dynamics is formulated at the impulse-momentum level as an MLCP. This reformulation is compatible with unilateral contact with friction and able to model stick–slip transitions. The proposed RIM was tested in the numerical simulation of hydraulically actuated mechanical systems. Results confirmed the ability of the RIM to enhance the stability and accuracy of noniterative multirate co-simulation setups with multiphysics components.

2 Co-Simulation of Multibody Systems With Interface Models

Let us consider a multibody system $\mathcal{M}$ that interacts with another subsystem $\mathcal{S}$, which can also be composed of several other subsystems $\mathcal{S}_1$, $\mathcal{S}_2$,… $\mathcal{S}_n$, as shown in Fig. 1. As far as the multibody system is concerned, all the interactions with $\mathcal{S}$ can be considered by means of one single interface. In many practical applications, $\mathcal{S}$ represents components with dynamics and time scales different from those of a multibody system, e.g., hydraulics or electronics. Such components often need to be integrated at faster rates than their mechanical counterparts. Here, we consider a multirate setup where subsystem $\mathcal{S}$ uses a smaller integration step-size than the multibody system, i.e., $h_{\mathcal{M}} > h_{\mathcal{S}}$.

The multibody system $\mathcal{M}$ and the subsystem $\mathcal{S}$ exchange information in the form of inputs $\mathbf{u}$ and outputs $\mathbf{y}$ at discrete-time communication points, as shown in Fig. 2. Each subsystem, $\mathcal{M}$ and $\mathcal{S}$, has its own states and integration methods, which are not accessible to the rest of the co-simulation environment. The communication between the two subsystems takes place via a co-simulation interface, and the data are exchanged at the beginning of each macrostep of size H. Here, we chose a noniterative scheme, where the subsystems are integrated in parallel (i.e., Jacobi scheme). Therefore, the subsystems do not receive any information from each other until the next communication point. For this reason, the input values of the fast subsystem $\mathbf{u}_{\mathcal{S}}$ must be extrapolated in some way within the macrotime-step. A RIM can provide a physics-based prediction of the coupling variables between communication points. Instead of extrapolating the inputs using time-history data of the coupling variables, RIMs approximate the output variables of the multibody system $\mathbf{y}_{\mathcal{M}}$ between communication points in terms of its most recent available inputs $\mathbf{u}_{\mathcal{M}}$ and its dynamics [22].

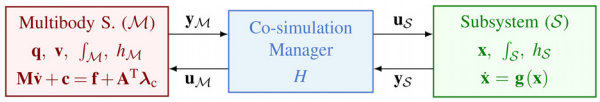

Fig. 2 Block diagram of a multibody system $\mathcal{M}$ and subsystem $\mathcal{S}$ coupled in a co-simulation setup

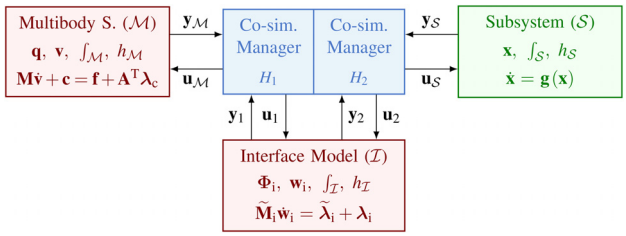

Fig. 3 Block diagram of a multibody system $\mathcal{M}$ and subsystem $\mathcal{S}$ coupled in a co-simulation setup via an interface model $\mathcal{I}$ of the multibody system

The introduction of a RIM of subsystem $\mathcal{M}$ in the co-simulation leads to the scheme shown in Fig. 3, a multirate integration algorithm in which two communication step-sizes, H_1 and H_2, can be employed. With this approach, subsystem $\mathcal{S}$ exchanges information through the co-simulation manager only with the RIM, denoted by $\mathcal{I}$. Both $\mathcal{I}$ and $\mathcal{S}$ are integrated at a faster rate and synchronized every H_2. Furthermore, if the computational power allows it, they can even be integrated simultaneously, i.e., $H_2 = h_{\mathcal{S}} = h_{\mathcal{I}}$. The full multibody system model $\mathcal{M}$, on the other hand, is integrated at a slower rate and synchronized with the rest of the system every macrotime-step of size $H_1 > H_2$. The timeline of the resulting co-simulation setup is shown in Fig. 4. It must be noted that the output $\mathbf{y}_{\mathcal{M}}$, and subsequently the inputs of $\mathcal{I}$, $\mathbf{u}_1$, must contain all the information necessary to generate the RIM.

The configuration and velocity of a multibody system can be parameterized by a set of r generalized coordinates $\mathbf{q}$ and a set of n generalized velocities $\mathbf{v}$; in general, $r \geqslant n$. The relation between them can always be written as $\dot{\mathbf{q}} = \mathbf{N}\mathbf{v}$, where $\mathbf{N}(\mathbf{q})$ is an $r \times n$ transformation matrix. The interactions between the bodies can be parameterized by a set of m_{c} velocity components $\mathbf{w}_{\mathrm{c}}$, which can be related to the generalized velocities as

$$\mathbf{w}_{\mathrm{c}} = \mathbf{A}\mathbf{v} \tag{1}$$

where $\mathbf{A}(\mathbf{q})$ is the $m_{\mathrm{c}} \times n$ constraint Jacobian matrix. Such interactions can be modeled via kinematic constraints, either bilateral or unilateral. The dynamic equations of a multibody system can be written as

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{c} = \mathbf{f} + \mathbf{A}^{\mathrm{T}}\boldsymbol{\lambda}_{\mathrm{c}} \tag{2}$$

where $\mathbf{M}(\mathbf{q})$ is the $n \times n$ mass matrix, $\mathbf{c}(\mathbf{q}, \mathbf{v})$ is the $n \times 1$ array of Coriolis and centrifugal terms, $\mathbf{f}$ is the $n \times 1$ array of generalized forces, and $\boldsymbol{\lambda}_{\mathrm{c}}$ is the $m_{\mathrm{c}} \times 1$ array of constraint forces. Here, the term force is used in the generalized sense, and the components in $\boldsymbol{\lambda}_{\mathrm{c}}$ can be forces or moments depending on the kind of velocity component constrained in $\mathbf{w}_{\mathrm{c}}$. The generalized forces can be decomposed as $\mathbf{f} = \mathbf{f}_{\mathrm{a}} + \mathbf{f}_{\mathrm{i}}$, where $\mathbf{f}_{\mathrm{a}}$ contains the generalized applied forces, and $\mathbf{f}_{\mathrm{i}}$ contains the generalized interface forces.

The interface between the multibody system $\mathcal{M}$ and subsystem $\mathcal{S}$ can be parametrized using m_{i} interface velocities

$$\mathbf{w}_{\mathrm{i}} = \mathbf{B}\mathbf{v} \tag{3}$$

where $\mathbf{B}(\mathbf{q})$ is the $m_{\mathrm{i}} \times n$ interface Jacobian matrix. In some cases, the interface velocities can be defined as the time derivatives of a set of interface coordinates $\boldsymbol{\Phi}_{\mathrm{i}}(\mathbf{q})$, as $\mathbf{w}_{\mathrm{i}} = \dot{\boldsymbol{\Phi}}_{\mathrm{i}}$. With such a parameterization, the generalized interface forces $\mathbf{f}_{\mathrm{i}}$ can then be expressed in terms of m_{i} interface forces $\boldsymbol{\lambda}_{\mathrm{i}}$ as

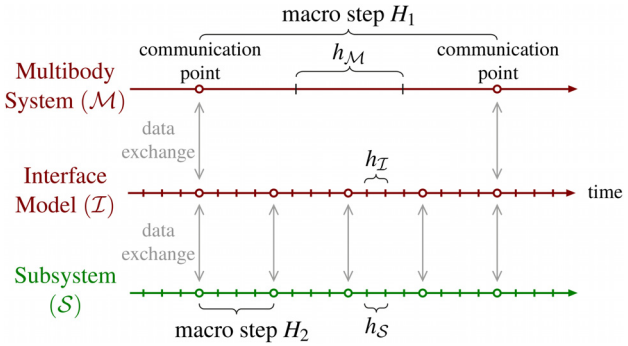

Fig. 4 Timeline of Jacobi-scheme co-simulation using an interface model $\mathcal{I}$ of the multibody system $\mathcal{M}$

$$\mathbf{f}_{\mathrm{i}} = \mathbf{B}^{\mathrm{T}}\boldsymbol{\lambda}_{\mathrm{i}} \tag{4}$$

The force components in $\boldsymbol{\lambda}_{\mathrm{i}}$ can represent forces and moments, depending on the kind of velocity components in $\mathbf{w}_{\mathrm{i}}$. The interface forces are usually provided by the external subsystem $\mathcal{S}$, and so $\boldsymbol{\lambda}_{\mathrm{i}}$ is part of the input of the multibody system $\mathcal{M}$. In this case, the interface kinematics $\boldsymbol{\Phi}_{\mathrm{i}}$ and $\mathbf{w}_{\mathrm{i}}$ are the output $\mathbf{y}_{\mathcal{M}}$. However, other schemes where the input of the multibody system $\mathbf{u}_{\mathcal{M}}$ is the interface kinematics are also possible. Then, constraints can be used to model the interface in the multibody system, and the constraint force becomes the output $\mathbf{y}_{\mathcal{M}}$ of the multibody system. This is known as force–displacement coupling, and the direction in which the interface variables are exchanged (i.e., whether the interface force is the input or the output of the multibody system) depends on the nature of each subsystem and how the dynamics is formulated. Nevertheless, other coupling schemes are also possible, such as *force–force* coupling or *displacement–displacement* coupling [9].

The RIM can be obtained by expressing the dynamics of the multibody system $\mathcal{M}$ in terms of the interface velocities $\mathbf{w}_{\mathrm{i}}$. If all constraints are bilateral, the dynamics of the RIM can be expressed as [22]

$$\tilde{\mathbf{M}}_{\mathrm{i}}\dot{\mathbf{w}}_{\mathrm{i}} = \tilde{\boldsymbol{\lambda}}_{\mathrm{i}} + \boldsymbol{\lambda}_{\mathrm{i}} \tag{5}$$

where the *effective mass* $\tilde{\mathbf{M}}_{\mathrm{i}}$ and *effective force* $\tilde{\boldsymbol{\lambda}}_{\mathrm{i}}$ terms are

$$\tilde{\mathbf{M}}_{\mathrm{i}} = \left(\mathbf{B}(\mathbf{I} - \mathbf{P}_{\mathrm{c}})\mathbf{M}^{-1}\mathbf{B}^{\mathrm{T}}\right)^{-1} \tag{6}$$

$$\tilde{\boldsymbol{\lambda}}_{\mathrm{i}} = \tilde{\mathbf{M}}_{\mathrm{i}}\left(\mathbf{B}(\mathbf{I} - \mathbf{P}_{\mathrm{c}})\mathbf{M}^{-1}(\mathbf{f}_{\mathrm{a}} - \mathbf{c}) + \dot{\mathbf{B}}\mathbf{v} + \mathbf{B}\mathbf{P}_{\mathrm{c}}\dot{\mathbf{v}}\right) \tag{7}$$

The projector matrix $\mathbf{P}_{\mathrm{c}} = \mathbf{M}^{-1}\mathbf{A}^{\mathrm{T}}(\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^{\mathrm{T}})^{-1}\mathbf{A}$ accounts for the topology of the system and the connection between all the bodies. However, the interface model of a nonsmooth multibody system cannot be directly described by the expression of the effective mass and force terms above. This is because of inequalities in the dynamics formulation due to unilateral contact and friction in the system. Therefore, the RIM needs to be reformulated in order to account for contact detachment and stick–slip transitions.

3 Nonsmooth Multibody System Dynamics

Constraints can represent the interactions between the bodies by specifying the motion of the system with kinematic relations. In general, we can consider three different kinds of interactions: bilateral, unilateral, and friction. Therefore, the constraint forces $\boldsymbol{\lambda}_{\mathrm{c}}$ and constraint velocities $\mathbf{w}_{\mathrm{c}}$ that act on a multibody system with unilateral contacts can be arranged as

$$\boldsymbol{\lambda}_c = \begin{bmatrix} \boldsymbol{\lambda}_b \\ \boldsymbol{\lambda}_n \\ \boldsymbol{\lambda}_t \end{bmatrix} \quad \text{and} \quad \mathbf{w}_c = \begin{bmatrix} \mathbf{w}_b \\ \mathbf{w}_n \\ \mathbf{w}_t \end{bmatrix} = \begin{bmatrix} \mathbf{A}_b \\ \mathbf{A}_n \\ \mathbf{A}_t \end{bmatrix} \mathbf{v} \tag{8}$$

where $\boldsymbol{\lambda}_b$ contains the m_b bilateral constraint force components, and $\boldsymbol{\lambda}_n$ and $\boldsymbol{\lambda}_t$ contain the m_n normal contact force components and m_t tangential (or friction) force components of all the contact points. Here, we consider one normal component and two tangential components for each contact point. Likewise, the m_b bilateral constraint velocity components are arranged in $\mathbf{w}_b$, and the m_n normal contact velocity components and the m_t tangential (or friction) velocity components are arranged in $\mathbf{w}_n$ and $\mathbf{w}_t$, respectively.

Contact between the bodies can be represented with unilateral constraints. The contact interface can be described by m_n contact points, and the distance between contact surfaces at the points is defined to be non-negative. Therefore, the m_n contact distances can be arranged into the array $\boldsymbol{\Phi}_n(\mathbf{q}) \geqslant \mathbf{0}$, which defines the normal velocity components as $\mathbf{w}_n = \dot{\boldsymbol{\Phi}}_n$. The normal contact force must always be compressive, and so it is also defined to be non-negative

$$\boldsymbol{\lambda}_n \geqslant \mathbf{0} \tag{9}$$

Furthermore, the complementarity condition between contact force and contact distance can be written as

$$\mathbf{0} \leqslant \boldsymbol{\Phi}_n \perp \boldsymbol{\lambda}_n \geqslant \mathbf{0} \tag{10}$$

where the operator $\perp$ denotes componentwise complementarity, i.e., $\Phi_{nj}\lambda_{nj} = 0$, $\forall j = 1 \ldots m_n$. This condition represents that the contact force is zero when the contact detaches, and positive when the contact is closed. In addition, if we only consider the contacts that are closed (i.e., $\boldsymbol{\Phi}_n = \mathbf{0}$), then complementarity can also be defined between force and velocity as

$$\mathbf{0} \leqslant \mathbf{w}_n \perp \boldsymbol{\lambda}_n \geqslant \mathbf{0} \tag{11}$$

Static friction can also be represented by constraints, which prevent the contact points from sliding, i.e., $\mathbf{w}_t = \mathbf{0}$. However, the magnitude of the static friction force is limited by Coulomb's law as

$$\|\boldsymbol{\lambda}_{tj}\| \leqslant \mu_j \lambda_{nj}, \quad \forall j = 1 \ldots m_n \tag{12}$$

where $\boldsymbol{\lambda}_{tj}$ is the 2×1 array of the friction force components of the jth contact point, λ_{nj} is its normal force component, and μ_j is the friction coefficient. On the other hand, the kinetic friction is defined by the sliding direction as

$$\boldsymbol{\lambda}_{tj} = -\mu_j \lambda_{nj} \mathbf{e}_{tj}, \quad \forall j = 1 \ldots m_n \tag{13}$$

where $\mathbf{e}_{tj} = \left(\mathbf{w}_{tj} / \|\mathbf{w}_{tj}\|\right)$ is the 2×1 array of the unit vector, which defines the sliding direction, and $\mathbf{w}_{tj}$ contains the two components of the sliding velocity.

These conditions that define friction are clearly nonlinear, and using them together with the dynamics equations of systems with unilateral contact leads to the formulation of a NCP. However, it is possible to replace these conditions with linear ones and formulate a MLCP. For this, it is necessary to select a finite number of directions on the tangent plane. Then, an upper and lower bound can be defined for the component of the friction force along each direction as

$$-\mu_j \lambda_{nj} \leqslant \mathbf{e}_k^T \boldsymbol{\lambda}_{tj} \leqslant +\mu_j \boldsymbol{\lambda}_{nj} \tag{14}$$

where $\mathbf{e}_k$ is the 2×1 array with the two components of the unit vector along that direction. Ideally, we would need to enforce these bounds in every direction on the plane in order to satisfy Coulomb's law. If only a few directions are chosen, the model will approximate the friction cone by a faceted cone (or pyramid). Here, we choose two orthogonal directions on the tangent plane (i.e., $k = 1, 2$) and define the bounds for the tangent contact force $\boldsymbol{\lambda}_t$ as

$$-\boldsymbol{\mu}\boldsymbol{\lambda}_n \leqslant \boldsymbol{\lambda}_t \leqslant +\boldsymbol{\mu}\boldsymbol{\lambda}_n \tag{15}$$

where $\boldsymbol{\mu} = \text{diag}\left([\mu_1\ \mu_1]^T \ldots [\mu_k\ \mu_k]^T\right)$ is the friction coefficient matrix. The coupling between normal and tangential directions can be considered via implementing this definition for the bounds in an iterative fashion. Nevertheless, the values of the bounds can also be approximated using the normal force of the previous time-step.

Such bounds in the constraint forces introduce nonsmoothness into the system, and can be defined in general as

$$\boldsymbol{\lambda}_c^{lo} \leqslant \boldsymbol{\lambda}_c \leqslant \boldsymbol{\lambda}_c^{up} \tag{16}$$

where $\boldsymbol{\lambda}_c^{lo}$ and $\boldsymbol{\lambda}_c^{up}$ are the lower and upper force bounds, which can be set to infinity for bilateral constraints. Unilateral constraints representing the normal contact force require a zero lower bound since the contact force must be non-negative, then

$$\mathbf{0} \leqslant \boldsymbol{\lambda}_n \leqslant +\infty \tag{17}$$

and friction force bounds, on the other hand, can be defined by Eq. (15). These force bounds define the limit of the interactions described by the constraints. Therefore, constraints can only be enforced if the force is within bounds, and in such a case, they cannot be distinguished from bilateral constraints. However, when the force is at the bound, the constraint cannot be maintained and must yield, for instance, when the maximum friction force is reached and the contact starts sliding. Such a behavior can be described by complementarity conditions, but since upper and lower bounds are considered, the constraint velocity needs to be decomposed in two non-negative components as

$$\mathbf{w}_c = \mathbf{w}_c^{lo} - \mathbf{w}_c^{up} \tag{18}$$

These components are also complementary to each other, meaning that only one can be different from zero, which can be enforced through the following complementarity conditions with the force bounds as:

$$\left.\begin{aligned} \mathbf{0} \leqslant \mathbf{w}_c^{lo} \quad \perp \left(\boldsymbol{\lambda}_c - \boldsymbol{\lambda}_c^{lo}\right) \geqslant \mathbf{0} \\ \mathbf{0} \leqslant \mathbf{w}_c^{up} \quad \perp \left(\boldsymbol{\lambda}_c^{up} - \boldsymbol{\lambda}_c\right) \geqslant \mathbf{0} \end{aligned}\right\} \tag{19}$$

Here, to simplify the notation, the complementarity condition in Eq. (19) will also be written in a more compact form as

$$\mathbf{w}_c \perp \boldsymbol{\lambda}_c \in \left[\boldsymbol{\lambda}_c^{lo}, \boldsymbol{\lambda}_c^{up}\right] \tag{20}$$

These force bounds introduce nonsmoothness into the system dynamics. An efficient way of dealing with nonsmooth systems is to formulate the dynamics at the velocity level using a first-order integrator because the order of accuracy in such systems cannot be higher than one [35]. Here, the time discretization is done via the formulation of the equations at the impulse-momentum level as it is shown below.

3.1 Impulse-Momentum Formulation. Given an instant of time t_k, the configuration $\mathbf{q} = \mathbf{q}(t_k)$ and velocity $\mathbf{v} = \mathbf{v}(t_k)$ of the system are assumed to be known. We then consider a time-step of size h so that the configuration $\mathbf{q}^+ = \mathbf{q}(t_{k+1})$ and velocity $\mathbf{v}^+ = \mathbf{v}(t_{k+1})$ at the instant $t_{k+1} = t_k + h$ are unknown. The dynamic equations at the impulse-momentum level can be written as [30,36,37]

$$\hat{\mathbf{M}}\mathbf{v}^+ + h\hat{\mathbf{c}} = \mathbf{M}\mathbf{v} + h\mathbf{f} + \mathbf{A}^{\mathrm{T}} h\boldsymbol{\lambda}_{\mathrm{c}}^+ \tag{21}$$

where the expressions for the modified mass matrix $\hat{\mathbf{M}}$, and Coriolis and centrifugal terms $\hat{\mathbf{c}}$ depend on how the time discretization is carried out. These can be obtained via an implicit formulation of the inertial and gyroscopic terms in $\mathbf{M}$ and $\mathbf{c}$, which can improve simulation stability. Nevertheless, the simplest case is $\hat{\mathbf{M}} = \mathbf{M}$ and $\hat{\mathbf{c}} = \mathbf{c}$, which corresponds to the so-called Moreau time-stepping scheme [35].

Furthermore, the constraint velocity at the next time-step $\mathbf{w}_{\mathrm{c}}^+ = \mathbf{w}_{\mathrm{c}}(t_{k+1})$ can be estimated via a first-order Taylor series of the constraint velocity $\mathbf{w}_{\mathrm{c}}$ at t_k as

$$\mathbf{w}_{\mathrm{c}}^+ = \mathbf{w}_{\mathrm{c}} + \frac{\mathrm{d}\mathbf{w}_{\mathrm{c}}}{\mathrm{d}t} h + \mathcal{O}(h^2) \tag{22}$$

where $\mathbf{w}_{\mathrm{c}}(t_k) = \mathbf{A}\mathbf{v}$ is the known constraint velocity at time t_k. Therefore, given the expression for the derivative of the constraint velocity $(\mathrm{d}\mathbf{w}_{\mathrm{c}}/\mathrm{d}t) = \dot{\mathbf{w}}_{\mathrm{c}} = \mathbf{A}\dot{\mathbf{v}} + \dot{\mathbf{A}}\mathbf{v}$, and using the finite difference approximation $\dot{\mathbf{v}} = (\mathbf{v}^+ - \mathbf{v})/h$, the constraint velocity at the instant t_{k+1} can be written as

$$\mathbf{w}_{\mathrm{c}}^+ = \mathbf{A}\mathbf{v}^+ + h\dot{\mathbf{A}}\mathbf{v} \tag{23}$$

where $\dot{\mathbf{A}}(\mathbf{q}, \mathbf{v})$ is computed using the known quantities $\mathbf{q}$ and $\mathbf{v}$ at the instant t_k. Then, complementarity conditions in Eq. (19) can be defined between the unknown constraint velocities $\mathbf{w}_{\mathrm{c}}^+$ and the constraint forces $\boldsymbol{\lambda}^+$ to formulate a complementarity problem.

3.2 The Mixed Linear Complementarity Problem. The dynamic equations at the impulse-momentum level in Eq. (21), together with the constraint velocities in Eq. (23), and the complementarity conditions in Eq. (20), form the MLCP

$$\left.\begin{gathered}\begin{bmatrix} \hat{\mathbf{M}} & -\mathbf{A}^{\mathrm{T}} \\ \mathbf{A} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{v}^+ \\ h\boldsymbol{\lambda}_{\mathrm{c}}^+ \end{bmatrix} + \begin{bmatrix} h(\hat{\mathbf{c}} - \mathbf{f}) - \mathbf{M}\mathbf{v} \\ h\dot{\mathbf{A}}\mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{w}_{\mathrm{c}}^+ \end{bmatrix} \\ \mathbf{w}_{\mathrm{c}}^+ \perp \boldsymbol{\lambda}_{\mathrm{c}}^+ \in \left[\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{lo}}, \boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{up}}\right] \end{gathered}\right\} \tag{24}$$

Furthermore, eliminating $\mathbf{v}^+$ in Eq. (24), one can obtain

$$\left(\mathbf{A}\hat{\mathbf{M}}^{-1}\mathbf{A}^{\mathrm{T}}\right) h\boldsymbol{\lambda}_{\mathrm{c}}^+ + \mathbf{A}\hat{\mathbf{M}}^{-1}(h(\mathbf{f} - \mathbf{c}) + \mathbf{M}\mathbf{v}) + h\dot{\mathbf{A}}\mathbf{v} = \mathbf{w}_{\mathrm{c}}^+ \tag{25}$$

for which $\boldsymbol{\lambda}_{\mathrm{c}}^+$ are the system unknowns. The reduced form of the MLCP can then be written as

$$\mathbf{H}\, h\boldsymbol{\lambda}_{\mathrm{c}}^+ + \mathbf{b} = \mathbf{w}_{\mathrm{c}}^+ \quad \perp \quad \boldsymbol{\lambda}_{\mathrm{c}}^+ \in \left[\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{lo}}, \boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{up}}\right] \tag{26}$$

where $\mathbf{H} = \mathbf{A}\hat{\mathbf{M}}^{-1}\mathbf{A}^{\mathrm{T}}$ is the $m_{\mathrm{c}} \times m_{\mathrm{c}}$ symmetric matrix that represents the inverse effective mass of the system in the constraint space, and the $m_{\mathrm{c}} \times 1$ array $\mathbf{b} = \mathbf{A}\hat{\mathbf{M}}^{-1}(h(\mathbf{f} - \mathbf{c}) + \mathbf{M}\mathbf{v}) + h\dot{\mathbf{A}}\mathbf{v}$ is known. Matrix $\mathbf{H}$ is positive definite if the constraints are not dependent on each other. Otherwise, it may become positive semi-definite and constraint force indeterminacy can occur, which can be resolved via constraint regularization, for instance.

The solution of the MLCP in Eq. (26) can be found by means of a pivoting algorithm, searching through the set of candidate solutions. For each possible solution, we can distinguish between *tight* and *active* constraints. Tight constraint forces can be either at the lower or upper bound, $\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{lo}}$ or $\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{up}}$, while active constraint forces must be within bounds. The constraint force and velocity components, $\boldsymbol{\lambda}_{\mathrm{c}}$ and $\mathbf{w}_{\mathrm{c}}$, can be rearranged as

$$\boldsymbol{\lambda}_{\mathrm{c}} = \begin{bmatrix} \boldsymbol{\lambda}_\alpha \\ \boldsymbol{\lambda}_\tau \end{bmatrix} \quad \text{and} \quad \mathbf{w}_{\mathrm{c}} = \begin{bmatrix} \mathbf{w}_\alpha \\ \mathbf{w}_\tau \end{bmatrix} = \begin{bmatrix} \mathbf{A}_\alpha \mathbf{v} \\ \mathbf{A}_\tau \mathbf{v} \end{bmatrix} \tag{27}$$

where $\boldsymbol{\lambda}_\alpha$ and $\mathbf{w}_\alpha$ represent the active constraint forces and velocities, respectively, and $\boldsymbol{\lambda}_\tau$ and $\mathbf{w}_\tau$ the tight ones. With this grouping, the MLCP in Eq. (26) can be rewritten as

$$\begin{bmatrix} \mathbf{H}_{\alpha\alpha} & \mathbf{H}_{\alpha\tau} \\ \mathbf{H}_{\tau\alpha} & \mathbf{H}_{\tau\tau} \end{bmatrix} \begin{bmatrix} h\boldsymbol{\lambda}_\alpha^+ \\ h\boldsymbol{\lambda}_\tau^+ \end{bmatrix} + \begin{bmatrix} \mathbf{b}_\alpha \\ \mathbf{b}_\tau \end{bmatrix} = \begin{bmatrix} \mathbf{w}_\alpha^+ \\ \mathbf{w}_\tau^+ \end{bmatrix} \tag{28}$$

since the tight forces in $\boldsymbol{\lambda}_\tau^+$ are known, and equal to either $\boldsymbol{\lambda}_\tau^{\mathrm{lo}}$ or $\boldsymbol{\lambda}_\tau^{\mathrm{up}}$, the active forces can be evaluated as

$$h\boldsymbol{\lambda}_\alpha^+ = -\mathbf{H}_{\alpha\alpha}^{-1}\left(\mathbf{b}_\alpha + \mathbf{H}_{\alpha\tau} h\boldsymbol{\lambda}_\tau^+\right) \tag{29}$$

where $\mathbf{w}_\alpha^+ = \mathbf{0}$ has been used, because of the complementarity condition in Eq. (19). Then, $\mathbf{w}_\tau^+$ can be computed as

$$\mathbf{w}_\tau^+ = \mathbf{H}_{\tau\alpha} h\boldsymbol{\lambda}_\alpha^+ + \mathbf{H}_{\tau\tau} h\boldsymbol{\lambda}_\tau^+ + \mathbf{b}_\tau \tag{30}$$

A pivoting algorithm essentially iterates over different possible solutions and checks if the computed force $\boldsymbol{\lambda}_\alpha^+$ is within the bounds, as well as if the slack velocity $\mathbf{w}_\tau^+$ satisfies complementarity with the tight force $\boldsymbol{\lambda}_\tau$. Then, when all the conditions are satisfied, the algorithm terminates and a solution is found.

4 Reduced Interface Models of Nonsmooth Systems

The interface of a multibody system $\mathcal{M}$ with another subsystem $\mathcal{S}$ can be characterized by the interface velocities $\mathbf{w}_{\mathrm{i}}$ and forces $\boldsymbol{\lambda}_{\mathrm{i}}$ from Eqs. (3) and (4). The interface velocity can be discretized as the constraint velocity in Eq. (23) as

$$\mathbf{w}_{\mathrm{i}}^+ = \mathbf{B}\mathbf{v}^+ + h\dot{\mathbf{B}}\mathbf{v} \tag{31}$$

These velocity quantities can be incorporated into the dynamics formulation so that the MLCP in Eq. (24) can be rearranged as

$$\left.\begin{gathered}\begin{bmatrix} \hat{\mathbf{M}} & -\mathbf{A}^{\mathrm{T}} & -\mathbf{B}^{\mathrm{T}} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}^+ \\ h\boldsymbol{\lambda}_{\mathrm{c}}^+ \\ h\boldsymbol{\lambda}_{\mathrm{i}}^+ \end{bmatrix} + \begin{bmatrix} h(\hat{\mathbf{c}} - \mathbf{f}_{\mathrm{a}}) - \mathbf{M}\mathbf{v} \\ h\dot{\mathbf{A}}\mathbf{v} \\ h\dot{\mathbf{B}}\mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{w}_{\mathrm{c}}^+ \\ \mathbf{w}_{\mathrm{i}}^+ \end{bmatrix} \\ \mathbf{w}_{\mathrm{c}}^+ \perp \boldsymbol{\lambda}_{\mathrm{c}}^+ \in \left[\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{lo}}, \boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{up}}\right] \end{gathered}\right\} \tag{32}$$

Depending on the coupling approach, either $\mathbf{w}_{\mathrm{i}}^+$ or $\boldsymbol{\lambda}_{\mathrm{i}}^+$ must become the input variable of the multibody system. Therefore, one of them is known when solving the MLCP. However, to derive a RIM, we need the relation between the two in order to predict the output of the multibody system independently from the chosen coupling approach.

From the first row in Eq. (32), the generalized velocities $\mathbf{v}^+$ can be expressed in terms of the constraint forces $\boldsymbol{\lambda}_{\mathrm{c}}^+$ and interface forces $\boldsymbol{\lambda}_{\mathrm{i}}^+$ as

$$\mathbf{v}^+ = \hat{\mathbf{M}}^{-1}\mathbf{A}^{\mathrm{T}}\boldsymbol{\lambda}_{\mathrm{c}}^+ + \hat{\mathbf{M}}^{-1}\mathbf{B}^{\mathrm{T}}\boldsymbol{\lambda}_{\mathrm{i}}^+ + \hat{\mathbf{M}}^{-1}(h(\mathbf{f}_{\mathrm{a}} - \hat{\mathbf{c}}) + \mathbf{M}\mathbf{v}) \tag{33}$$

Then, by substituting this expression into the other two rows, a reduced version of that MLCP can be written as

$$\left.\begin{gathered}\begin{bmatrix} \mathbf{A}\hat{\mathbf{M}}^{-1}\mathbf{A}^{\mathrm{T}} & \mathbf{A}\hat{\mathbf{M}}^{-1}\mathbf{B}^{\mathrm{T}} \\ \mathbf{B}\hat{\mathbf{M}}^{-1}\mathbf{A}^{\mathrm{T}} & \mathbf{B}\hat{\mathbf{M}}^{-1}\mathbf{B}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} h\boldsymbol{\lambda}_{\mathrm{c}}^+ \\ h\boldsymbol{\lambda}_{\mathrm{i}}^+ \end{bmatrix} + \begin{bmatrix} \mathbf{b}_{\mathrm{c}} \\ \mathbf{b}_{\mathrm{i}} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_{\mathrm{c}}^+ \\ \mathbf{w}_{\mathrm{i}}^+ \end{bmatrix} \\ \mathbf{w}_{\mathrm{c}}^+ \perp \boldsymbol{\lambda}_{\mathrm{c}}^+ \in \left[\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{lo}}, \boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{up}}\right] \end{gathered}\right\} \tag{34}$$

where $\mathbf{b}_{\mathrm{c}} = \mathbf{A}\hat{\mathbf{M}}^{-1}h(\mathbf{f}_{\mathrm{a}} - \hat{\mathbf{c}}) + \mathbf{A}\mathbf{v} + h\dot{\mathbf{A}}\mathbf{v}$, and $\mathbf{b}_{\mathrm{i}} = \mathbf{B}\hat{\mathbf{M}}^{-1}h(\mathbf{f}_{\mathrm{a}} - \hat{\mathbf{c}}) + \mathbf{B}\mathbf{v} + h\dot{\mathbf{B}}\mathbf{v}$ are known.

A similar procedure can be employed to eliminate the unknown constraint forces $\boldsymbol{\lambda}_{\mathrm{c}}^+$ from Eq. (34). However, due to the existence of force bounds, an especial treatment is required for the constraints. Tight constraints have the force at the lower or upper bound, $\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{lo}}$ or $\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{up}}$, while active constraints have the forces within bounds. This distinction is important for the prediction of the

system dynamics because the active constraint forces are the only ones that actually satisfy the constraint equation $\mathbf{w}_c^+ = \mathbf{0}$. Tight constraints, on the other hand, do not satisfy any constraint equation but rather an inequality condition for the constraint velocity according to the complementarity condition in Eq. (19). Therefore, tight constraints do not behave as constraints any more, which is the case, for instance, when a contact detaches and starts sliding.

Let us rearrange the constraint force and velocity arrays into tight and active as in Eq. (27). Then, the MLCP in Eq. (34) can also be rearranged as

$$\left[\begin{array}{cc|c} \mathbf{H}_{\alpha\alpha} & \mathbf{H}_{\alpha\tau} & \mathbf{H}_{\alpha i} \\ \mathbf{H}_{\tau\alpha} & \mathbf{H}_{\tau\tau} & \mathbf{H}_{\tau i} \\ \hline \mathbf{H}_{i\alpha} & \mathbf{H}_{i\tau} & \mathbf{H}_{ii} \end{array}\right] \left[\begin{array}{c} h\boldsymbol{\lambda}_\alpha^+ \\ h\boldsymbol{\lambda}_\tau^+ \\ \hline h\boldsymbol{\lambda}_i^+ \end{array}\right] + \left[\begin{array}{c} \mathbf{b}_\alpha \\ \mathbf{b}_\tau \\ \hline \mathbf{b}_i \end{array}\right] = \left[\begin{array}{c} \mathbf{w}_\alpha^+ \\ \mathbf{w}_\tau^+ \\ \hline \mathbf{w}_i^+ \end{array}\right] \tag{35}$$

Tight constraint forces are known and are assumed to be constant (i.e., $\boldsymbol{\lambda}_\tau^+ = \boldsymbol{\lambda}_\tau^{\text{lo}}$ or $\boldsymbol{\lambda}_\tau^{\text{up}}$), which can be taken from the solution of the MLCP from the previous time-step. Active constraint forces are considered within bounds, and so complementarity in Eq. (19) ensures that $\mathbf{w}_\alpha^+ = \mathbf{0}$. Therefore, active constraint forces $\boldsymbol{\lambda}_\alpha^+$ can be eliminated from the system by substitution from the first row into the last one so that an expression with the interface force and velocity can be written as

$$\left(\mathbf{H}_{ii} - \mathbf{H}_{i\alpha}\mathbf{H}_{\alpha\alpha}^{-1}\mathbf{H}_{\alpha i}\right)h\boldsymbol{\lambda}_i^+ + \mathbf{b}_i + \mathbf{H}_{i\tau}h\boldsymbol{\lambda}_\tau^+ - \mathbf{H}_{i\alpha}\mathbf{H}_{\alpha\alpha}^{-1}\left(\mathbf{b}_\alpha + \mathbf{H}_{\alpha\tau}h\boldsymbol{\lambda}_\tau^+\right) = \mathbf{w}_i^+ \tag{36}$$

This expression above can be interpreted as a reduced-order model of the nonsmooth multibody system, where the dynamics have been projected onto the space parameterized by the interface velocities $\mathbf{w}_i$.

Alternatively, the RIM in Eq. (36) can also be written as the impulse-momentum dynamics equations of the RIM in Eq. (5)

$$\tilde{\mathbf{M}}_i\left(\mathbf{w}_i^+ - \mathbf{w}_i\right) = h\left(\tilde{\boldsymbol{\lambda}}_i + \boldsymbol{\lambda}_i^+\right) \tag{37}$$

where now the effective mass $\tilde{\mathbf{M}}_i$ and force $\tilde{\boldsymbol{\lambda}}_i$ are

$$\tilde{\mathbf{M}}_i = \left(\mathbf{B}\left(\mathbf{I} - \hat{\mathbf{P}}_c\right)\hat{\mathbf{M}}^{-1}\mathbf{B}^{\mathrm{T}}\right)^{-1} \tag{38}$$

$$\tilde{\boldsymbol{\lambda}}_i = \tilde{\mathbf{M}}_i\left(\mathbf{B}\left(\mathbf{I} - \hat{\mathbf{P}}_c\right)\hat{\mathbf{M}}^{-1}\left(\mathbf{f}_a - \hat{\mathbf{c}} + \mathbf{B}^{\mathrm{T}}\boldsymbol{\lambda}_\tau^+\right) + \dot{\mathbf{B}}\mathbf{v} + \mathbf{B}\hat{\mathbf{P}}_c\frac{\mathbf{v}^+ - \mathbf{v}}{h}\right) \tag{39}$$

where $\hat{\mathbf{P}}_c = \hat{\mathbf{M}}^{-1}\mathbf{A}_\alpha^{\mathrm{T}}\left(\mathbf{A}_\alpha\hat{\mathbf{M}}^{-1}\mathbf{A}_\alpha^{\mathrm{T}}\right)^{-1}\mathbf{A}_\alpha$, and $\boldsymbol{\lambda}_\tau^+$ and $\mathbf{v}^+$ need to be determined by solving the MLCP in Eq. (32). The expressions of the effective mass and force terms are similar to the ones for smooth systems in Eqs. (6) and (7). However, matrix $\hat{\mathbf{P}}_c$ is now computed using the Jacobian matrix of the active constraints $\mathbf{A}_\alpha$, which makes the RIM account for contact states.

4.1 Constraint Regularization. Constraint redundancy occurs frequently in contact problems. In such cases, the constraint forces are no longer independent and the problem becomes indeterminate. To overcome this problem, constraint regularization can be used, which relaxes the constraints allowing constraint violations. Then, the constraint violations are used as representation for deformation profiles to define the forces considering constitutive relations, such as

$$\boldsymbol{\lambda}_c^+ = -\mathbf{K}\boldsymbol{\Phi}_c^+ - \mathbf{D}\mathbf{w}_c^+ \tag{40}$$

where $\mathbf{K}$ and $\mathbf{D}$ are $m_c \times m_c$ stiffness and damping matrices. Then we can use a first-order series expansion of the contact distance $\boldsymbol{\Phi}_c^+ = \boldsymbol{\Phi}_c(t_{k+1}) = \boldsymbol{\Phi}_c(t_k) + \mathbf{w}_c^+ h$, in order to express the constraint forces in terms of the unknown velocities. Then, the regularized version of the MLCP in Eq. (24) can be written as [36]

$$\left.\begin{array}{c} \left[\begin{array}{cc} \hat{\mathbf{M}} & -\mathbf{A}^{\mathrm{T}} \\ \mathbf{A} & \mathbf{C} \end{array}\right]\left[\begin{array}{c} \mathbf{v}^+ \\ h\boldsymbol{\lambda}_c^+ \end{array}\right] + \left[\begin{array}{c} h(\hat{\mathbf{c}} - \mathbf{f}) - \mathbf{M}\mathbf{v} \\ h\dot{\mathbf{A}}\mathbf{v} + \boldsymbol{\gamma}\boldsymbol{\Phi}_c h^{-1} \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \mathbf{w}_c^+ \end{array}\right] \\ \mathbf{w}_c^+ \perp \boldsymbol{\lambda}_c^+ \in \left[\boldsymbol{\lambda}_c^{\text{lo}}, \boldsymbol{\lambda}_c^{\text{up}}\right] \end{array}\right\} \tag{41}$$

where $\mathbf{C}$ and $\boldsymbol{\gamma}$ are $m_c \times m_c$ matrices containing the regularization terms, which are often assumed to be diagonal and positive definite. Furthermore, the equation for the constraint forces can be expressed as

$$\underbrace{\left(\mathbf{A}\hat{\mathbf{M}}^{-1}\mathbf{A}^{\mathrm{T}} + \mathbf{C}\right)}_{\mathbf{H}} h\boldsymbol{\lambda}_c^+ + \underbrace{\mathbf{A}\hat{\mathbf{M}}^{-1}(h(\mathbf{f} - \hat{\mathbf{c}}) + \mathbf{M}\mathbf{v}) + h\dot{\mathbf{A}}\mathbf{v} + \boldsymbol{\gamma}\boldsymbol{\Phi}_c h^{-1}}_{\mathbf{b}} = \mathbf{w}_c^+ \tag{42}$$

where $\mathbf{H}$ is now positive definite despite constraint redundancy.

If constraints are regularized, the expressions for the effective mass $\tilde{\mathbf{M}}_i$ and effective force $\tilde{\boldsymbol{\lambda}}_i$ in Eqs. (38) and (39) can be derived as

$$\tilde{\mathbf{M}}_i = \left(\mathbf{B}\left(\mathbf{I} - \hat{\mathbf{P}}_c\right)\hat{\mathbf{M}}^{-1}\mathbf{B}^{\mathrm{T}}\right)^{-1} \tag{43}$$

$$\tilde{\boldsymbol{\lambda}}_i = \tilde{\mathbf{M}}_i\Big(\mathbf{B}\left(\mathbf{I} - \hat{\mathbf{P}}_c\right)\hat{\mathbf{M}}^{-1}\left(\mathbf{f}_a - \hat{\mathbf{c}} + \mathbf{B}^{\mathrm{T}}\boldsymbol{\lambda}_\tau^+\right) + \dot{\mathbf{B}}\mathbf{v} - \mathbf{B}\hat{\mathbf{M}}^{-1}\mathbf{A}_\alpha^{\mathrm{T}}\left(\mathbf{A}_\alpha\hat{\mathbf{M}}^{-1}\mathbf{A}_\alpha^{\mathrm{T}} + \mathbf{C}_\alpha\right)^{-1}\left(\dot{\mathbf{A}}\mathbf{v} + \boldsymbol{\gamma}\boldsymbol{\Phi}_c h^{-2}\right)\Big) \tag{44}$$

where $\hat{\mathbf{P}}_c = \hat{\mathbf{M}}^{-1}\mathbf{A}_\alpha^{\mathrm{T}}\left(\mathbf{A}_\alpha\hat{\mathbf{M}}^{-1}\mathbf{A}_\alpha^{\mathrm{T}} + \mathbf{C}_\alpha\right)^{-1}\mathbf{A}_\alpha$.

4.2 Integration of the Reduced Interface Model. The time integration of the interface model is carried out using a first-order semi-implicit Euler integrator so that interface velocities and coordinates are updated as follows:

$$\left.\begin{array}{l} \mathbf{w}_i^+ = \mathbf{w}_i + h_{\mathcal{S}}\tilde{\mathbf{M}}_i^{-1}\left(\tilde{\boldsymbol{\lambda}}_i + \boldsymbol{\lambda}_i^+\right) \\ \boldsymbol{\Phi}_i^+ = \boldsymbol{\Phi}_i + h_{\mathcal{S}}\mathbf{w}_i^+ \end{array}\right\} \tag{45}$$

where $h_{\mathcal{S}}$ is the microtime-step of subsystem $\mathcal{S}$, ($h_{\mathcal{S}} < h_{\mathcal{M}}$). This integration is carried out between communication points; therefore, the effective mass and force terms $\tilde{\mathbf{M}}_i$ and $\tilde{\boldsymbol{\lambda}}_i$ are kept constant in the macrotime-step.

5 Examples

In this section, we describe the implementation of a co-simulation setup for multibody systems $\mathcal{M}$ with hydraulic components $\mathcal{H}$ using the interface model introduced above. Models of a hydraulic manipulator and a hydraulic crane, each with two actuators, were used to assess the methods. Nevertheless, the proposed methodology is general and can incorporate other subsystems of different nature. The results of numerical experiments are analyzed and discussed. The multibody models were created using the VORTEX simulation software package [38]; the co-simulation manager as well as the hydraulic model were written in C++ using the Eigen library for linear algebra and embedded into the software package. The simulations were performed on an Intel Core i7-8750H machine with a 6-core CPU at 2.20 GHz and 16 GB of RAM running 64-bit Windows 10.

5.1 Hydraulic Actuator Model. Hydraulic systems regulate the pressure of a fluid to generate the force that moves other

mechanical components. The dynamics of such systems governs the evolution of pressure during motion, which is largely controlled by valves and pumps. A hydraulic actuator is an enclosed cylinder with two chambers separated by a piston, see Fig. 5. The pressure difference between the two chambers generates a resultant force on the piston, which is transferred to the attachment points of the actuator. The magnitude of the hydraulic force exerted by the actuator can be expressed in terms of the hydraulic pressure difference as [5]

$$f_{\mathrm{h}} = (p_2 - p_1)a_{\mathrm{p}} - c\dot{s} \tag{46}$$

where p_1 and p_2 are the fluid pressures within the cylinder, and a_{p} is the total piston area. Viscous friction is considered in the cylinder through a viscous coefficient c so that the friction force is proportional to the actuator velocity $\dot{s}$.

The dynamics of the hydraulic system can be described with the following set of first order, ordinary differential equations:

$$\dot{p}_1 = \frac{\beta_1}{a_{\mathrm{p}} l_1}\left[a_{\mathrm{p}}\dot{s}_1 + a_{\mathrm{i}}c_{\mathrm{d}}\sqrt{\frac{2(p_{\mathrm{P}} - p_1)}{\rho}}\delta_{\mathrm{P1}} - a_{\mathrm{o}}c_{\mathrm{d}}\sqrt{\frac{2(p_1 - p_{\mathrm{T}})}{\rho}}\delta_{\mathrm{T1}}\right] \tag{47}$$

$$\dot{p}_2 = \frac{\beta_2}{a_{\mathrm{p}} l_2}\left[-a_{\mathrm{p}}\dot{s}_1 + a_{\mathrm{o}}c_{\mathrm{d}}\sqrt{\frac{2(p_{\mathrm{P}} - p_2)}{\rho}}\delta_{\mathrm{P2}} - a_{\mathrm{i}}c_{\mathrm{d}}\sqrt{\frac{2(p_2 - p_{\mathrm{T}})}{\rho}}\delta_{\mathrm{T2}}\right] \tag{48}$$

where l_1 and l_2 are the variable lengths of the chambers on each side of the piston, a_{i} and a_{o} are the variable valve areas that connect these cylinder chambers to the pump and the tank in the hydraulics system, c_{d} is the discharge coefficient of the valves, ρ stands for the fluid density, p_{P} and p_{T} are the hydraulic pressure at the pump and the tank, respectively. Coefficients δ_{P1}, δ_{P2}, δ_{T1}, and δ_{T2} are 0 when the quantity inside the square root that precedes them is negative and 1 otherwise. Terms β_1 and β_2 stand for the bulk modulus in each cylinder chamber, and they are evaluated as a function of the fluid pressure [39]

$$\beta_i = \frac{1 + ap_i + bp_i^2}{a + 2bp_i}, \quad i = 1, 2 \tag{49}$$

where a and b are constants for the fluid. Assuming that the two cylinder chambers have the same volume at the starting time of the simulation, chamber lengths l_1 and l_2 are given by

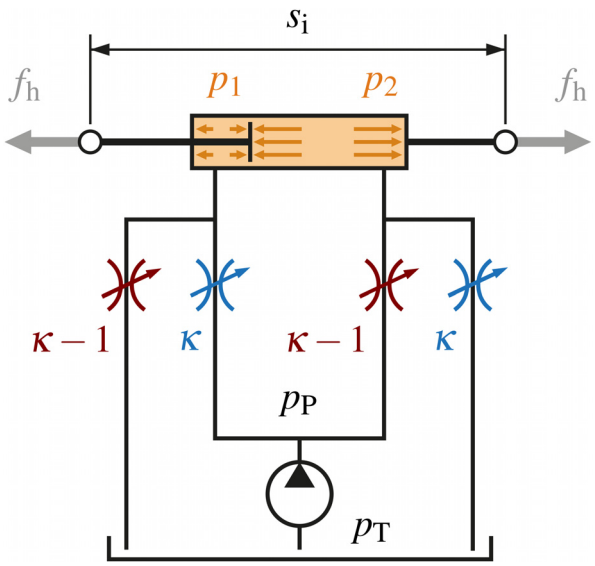

Fig. 5 Hydraulic model of an actuator

Table 1 Hydraulic parameters

Hydraulic parameters		Example 1	Example 2
Piston area	a_{p}	6.5×10^{-3} m^2	6.5×10^{-3} m^2
Cylinder length	l	0.442 m	1 m
Friction coefficient	c	10^5 Ns/m	5×10^5 Ns/m
Valve discharge coefficient	c_{d}	0.67	0.67
Fluid density	ρ	850 kg/m^3	850 kg/m^3
Hydraulic pressure at the pump	p_{P}	7.6 MPa	50 MPa
Hydraulic pressure at the tank	p_{T}	0.1 MPa	0.1 MPa
Compressibility coefficient	a	6.53×10^{-10} Pa	6.53×10^{-10} Pa
Compressibility coefficient	b	-1.19×10^{-18}	-1.19×10^{-18}

$$\begin{aligned} l_1 &= 0.5l + s_{1,0} - s_1 \\ l_2 &= 0.5l + s_1 - s_{1,0} \end{aligned} \tag{50}$$

where $s_{1,0}$ is the initial length of the actuator. Valve areas a_{i} and a_{o} have m^2 units and are obtained as

$$\begin{aligned} a_{\mathrm{i}} &= 5 \cdot 10^{-4}\kappa \\ a_{\mathrm{o}} &= 5 \cdot 10^{-4}(1 - \kappa) \end{aligned} \tag{51}$$

In Eq. (51), $\kappa \in [0, 1]$ is the valve control parameter or spool displacement, i.e., the kinematic input that controls the motion of the piston. The hydraulic parameters used in the following examples are shown in Table 1.

5.2 Example 1: Hydraulic Manipulator Model. The first test problem used in this research is a manipulator with two hydraulic actuators, shown in Fig. 6. The manipulator consists of two links, the first one connected to the ground via a revolute joint at point O, and the second one connected to the first via another revolute joint at point Q. Both joint axes are parallel to the global z axis. Two actuators control the device: one between the ground and the first link, and another between the two links. The mass of the actuators is negligible, and they are connected to other elements via frictionless spherical joints.

Link 1 has a homogeneously distributed mass m_1 and length L_1. The mass of link 2 is concentrated at points P and Q, with masses m_{P} and m_{Q}, respectively; its length is L_2. The links are assumed to be very slender, however, a small moment of inertia I_{a} about the longitudinal axis of both links is considered. Table 2 contains the value of all the parameters of the model. Actuator 1 is attached to the ground at point A represented by coordinates x_{A} and y_{A}. The other end of the actuator is attached to link 1 at a distance L_3 from

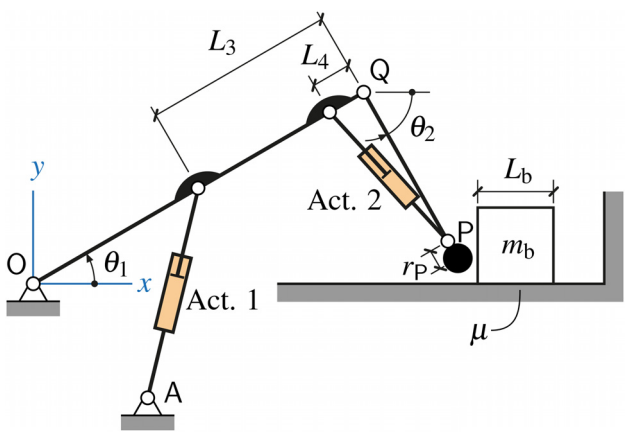

Fig. 6 Model of a hydraulic manipulator with two actuators in contact with a block

Table 2 System parameters of the 3D model of a hydraulic manipulator

Parameters		
Mass of link 1	m_1	200 kg
Mass at point Q	m_Q	250 kg
Mass at point P	m_P	100 kg
Inertia about link axes	I_a	0.1 kg m^2
Mass of the block	m_b	500 kg
Side of the block	L_b	0.2 m
Length on link 1	L_1	1 m
Length of link 2	L_2	0.5 m
Radius of the sphere at P	r_P	0.05 m
Attachment distance	L_3	0.5 m
Attachment distance	L_4	0.4 m
Friction coefficient	μ	0.5
Attachment point A	(x_A, y_A)	$(3, -3)$ m
Wall position	x_w	1.5 m
Initial orientation of link 1	θ_1	30deg
Initial orientation of link 2	θ_2	−60deg
Initial position of the block	x_b	1.3 m

point Q. Actuator 2 is attached to link 1 at a distance L_4 from point Q, and the other end is attached to link 2 at a distance r_P from point P.

During the maneuver, the end effector of the manipulator collides with a cubic block of mass m_b with side length L_b that is initially resting on the ground. The geometry of the end of link 2 is a sphere centered at point P, with the radius equal to the distance r_P. The friction coefficient between block and ground is μ, but the interface between manipulator and block is considered to be frictionless.

The actuation law to perform such a maneuver increases the length of actuator 2 by modifying the valve displacement κ_1, while keeping the valve displacement of the first actuator κ_2 constant. The actuation law can be written as

$$\kappa_1(t) = \kappa_1^0 \quad \text{and} \quad \kappa_2(t) = \begin{cases} \kappa_2^0 & \text{if} \quad t \leqslant t_{\text{ini}} \\ \kappa_2^0 + \Delta\kappa_2 \dfrac{t - t_{\text{ini}}}{t_{\text{end}} - t_{\text{ini}}} & \text{if} \quad t_{\text{ini}} < t \leqslant t_{\text{end}} \\ \kappa_2^0 + \Delta\kappa_2 & \text{if} \quad t > t_{\text{end}} \end{cases} \tag{52}$$

where $\Delta\kappa_2 = -0.015$ is the increment of the valve displacement between the time instants t_{ini} and t_{end}. The initial values $\kappa_1^0 = 0.4416$ and $\kappa_2^0 = 0.4871$ satisfy the static equilibrium of the system in the configuration specified in Table 2. Maneuver times were set to $t_{\text{ini}} = 1$ s and $t_{\text{end}} = 2$ s.

Several simulations were performed using different step-sizes for the integration of the multibody system $h_{\mathcal{M}}$, which was set to be the same as the macrostep-size H. The step-size for the

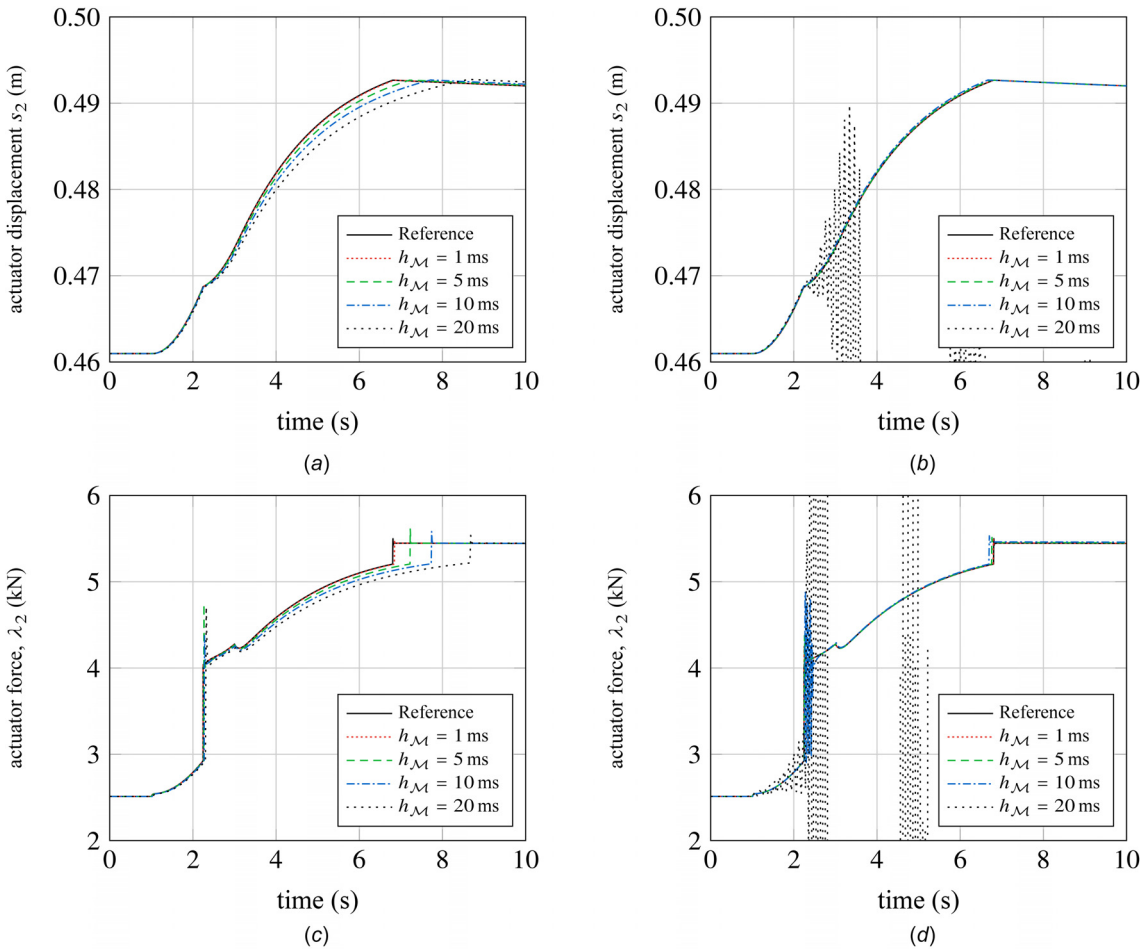

Fig. 7 Displacement and force of actuator 2 of the hydraulic manipulator using both the proposed interface model (left) and a zero-order hold of the inputs (right). Different step-sizes $h_{\mathcal{M}}$ were used for the multibody system, while the same step-size $h_{\mathcal{H}} = 0.2$ ms was used for the hydraulics in all the cases: (*a*) displacement with interface model, (*b*) displacement with zero-order hold, (*c*) force with interface model, and (*d*) Force with zero-order hold.

hydraulics was set to $h_{\mathcal{H}} = 0.2$ ms in all the simulations, in order to keep the stability of the hydraulic system. Figure 7 shows the displacement (a and b) and force (c and d) of the second actuator. Left-column plots show results obtained using the interface model described in Sec. 4; they are compared to the plots in the right column, which were obtained with zero-order hold (ZOH) direct co-simulation.

Results show that the simulation remains stable in all cases when using the proposed interface model (Fig. 7). This was not the case if conventional ZOH co-simulation was used instead, with which the numerical integration became unstable for large step-sizes, $h_{\mathcal{M}} \geq 10$ ms. The collision between the end effector of the manipulator and the block, right after $t = 2$ s, was a critical instant that triggered an oscillatory behavior in the system. Conventional ZOH co-simulation was unable to recover its stability after this point for $h_{\mathcal{M}} = 20$ ms and the numerical integration delivered inaccurate results. After $t = 2$ s, the manipulator stops briefly, until it overcomes the friction resistance between block and ground, and then makes the block slide until it reaches the wall and stops. Note that, in any case, the manipulator does not reach a steady-state at the end, i.e., the velocity is not constant, because there is no velocity control in this model.

Detailed plots of the velocity of actuator 2 at the time of impact with the block are shown in Fig. 8. The comparison of the two approaches highlights that both solutions agree for small step-sizes of the multibody system ($h_{\mathcal{M}} = 2$ ms). Larger step-sizes make the simulation unstable when using a zero-order hold, whereas using a reduced interface model allows for taking larger step-sizes without compromising stability. It can also be noted that, as expected, the larger the communication step-size, the further away the simulation results are from the reference solution with $h_{\mathcal{M}} = 2$ ms. For large communication step-sizes, however, the use of the interface model improved both the accuracy and the stability of the co-simulation.

5.3 Example 2: Hydraulic Crane With Gripper. To test the proposed methodology in a more realistic engineering application, the detailed 3D model of a hydraulic crane with two actuators and a gripper was used. This model has a total of 18 bodies and 22 joints, which include spherical, revolute, and prismatic joints. Figure 9 illustrates the main parts of the crane: boom, stick, and gripper. Each of these parts and the connections between them can contain several bodies and joints, such as the connection between boom and stick, which is essentially a four-bar linkage. Actuator 1 controls the boom elevation with respect to the base, which is fixed to the ground, and actuator 2 controls the stick elevation with respect to the boom. The gripper has three nonactuated rotational degrees-of-freedom with respect to the stick, and two movable claws that are controlled via kinematic constraints.

The inputs of the actuators are given by a velocity controller with proportional and derivative gains k^{P} and k^{D}. The output of the controller was chosen to be the derivative of the valve displacement $\dot{\kappa}$ so that the control law for each actuator $j = 1, 2$ can be defined as

$$\dot{\kappa}_j = -k_j^{\mathrm{P}}\left(w_j^* - w_j\right) - k_j^{\mathrm{D}}\left(\dot{w}_j^* - \dot{w}_j\right) \tag{53}$$

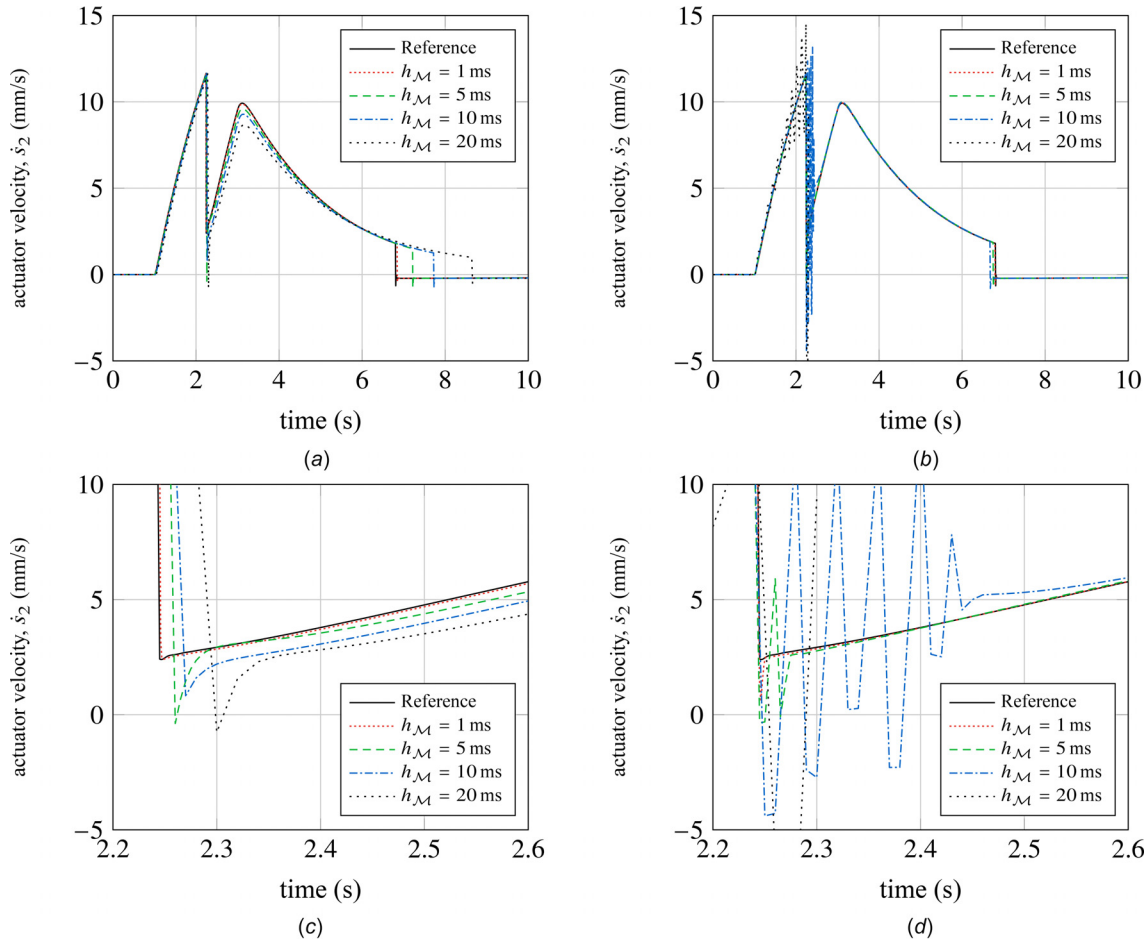

Fig. 8 Velocity of actuator 2 and detail at the time of impact with the block, using an interface model (left), and a zero-order-hold. Different step-sizes $h_{\mathcal{M}}$ were used for the multibody system, while the same step-size $h_{\mathcal{H}} = 0.2$ ms was used for the hydraulics in all the cases: (*a*) velocity with interface model, (*b*) velocity with zero-order hold, (*c*) velocity detail with interface model, and (*d*) velocity detail with zero-order hold.

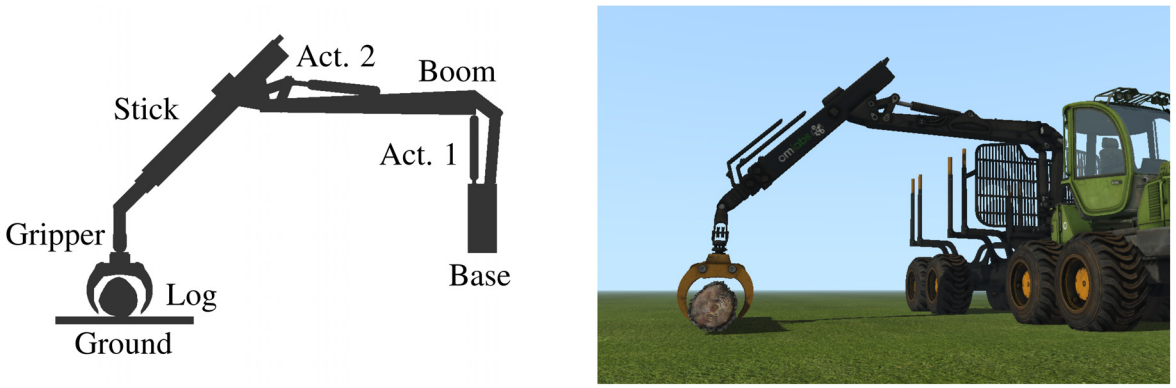

Fig. 9 Model of a log handling crane with two hydraulic actuators

where w_j^* is the controller desired velocity, and $\dot{w}_j^*$ is its time derivative. The gains were tuned manually to damp any vibrations in the system. Then, for actuator 1, $k_1^{\mathrm{P}} = 6.0\,\mathrm{m}^{-1}$ and $k_1^{\mathrm{D}} = 1.0\,\mathrm{sm}^{-1}$, and for actuator 2, $k_2^{\mathrm{P}} = 4.0\,\mathrm{m}^{-1}$ and $k_2^{\mathrm{D}} = 0.1\,\mathrm{sm}^{-1}$. Note that the controller has negative feedback, because, by design, an increment in the valve displacement κ would decrease the actuator force. Additionally, the controller output was limited to $\dot{\kappa} \in [-3, +3]\,\mathrm{s}^{-1}$.

The desired velocity of the actuators is displayed in Fig. 10, which can be written for both actuators as

$$w_j^* = \begin{cases} 0 & \text{if} \quad t \leqslant t_{\mathrm{ini}} \\ w_j^{\max} \dfrac{t - t_{\mathrm{ini}}}{t_{\max} - t_{\mathrm{ini}}} & \text{if} \quad t_{\mathrm{ini}} < t \leqslant t_{\max} \\ w_j^{\max} & \text{if} \quad t_{\max} < t \leqslant t_{\mathrm{end}} \\ 0 & \text{if} \quad t > t_{\mathrm{end}} \end{cases} \tag{54}$$

where $t_{\mathrm{ini}} = 5\,\mathrm{s}$ is the initial time without motion of the arm, where the grasping of the log occurs. The desired velocity is increased from zero to the maximum value at $t_{\max} = 6\,\mathrm{s}$, which is kept until $t_{\mathrm{end}} = 9\,\mathrm{s}$. The maximum desired velocity is set to be $w_1^{\max} = -50\,\mathrm{mm/s}$, for actuator 1, and $w_2^{\max} = -20\,\mathrm{mm/s}$, for actuator 2.

Friction is considered in all the contact interfaces with a coefficient $\mu = 0.5$. Additionally, since there is contact redundancy at the interface between gripper and log, the contact constraints are regularized with a stiffness $K_{\mathrm{n}} = 10^6\,\mathrm{N/m}$ and a damping coefficient $D_{\mathrm{n}} = 10^5\,\mathrm{Ns/m}$.

As in the previous example, the step-size of the hydraulics was fixed to $h_{\mathcal{H}} = 0.2$ ms, and different step-sizes $h_{\mathcal{M}}$ were used for the multibody system, which were the same as the macrostep-size H in all the simulations. Again, the results obtained with the proposed reduced interface model were compared against the ones delivered by zero-order hold direct co-simulation. In this case, the input is a desired velocity of the controller, which establishes the valve input to the actuators. Figure 10 shows the desired velocity of both actuators and the valve displacements given by the controller. Grasping of the log occurs for $t \in [3, 5]$ s.

The displacement and force of Actuator 1 are shown in Fig. 11 for both approaches and several values of the $h_{\mathcal{M}}$ step-size. Moreover, Fig. 12 shows the velocity of Actuator 1. As can be seen, the simulation becomes unstable for large step-sizes when using a zero-order hold direct co-simulation approach. On the other hand, it is possible to take larger step-sizes with the reduced interface model without losing stability. Figure 12 shows the velocity and force of actuator 1 during the maneuver. Significant oscillations can be observed in the velocity plots, e.g., in Fig. 12(*c*). This takes place during the grasping phase, where contact and friction forces have the most significant effect on the system dynamics. The interface model contributed to alleviate the severity of these oscillations. It is also noteworthy that zero-order hold co-simulation with $h_{\mathcal{M}} = 10$ ms recovers stability after grasping the log ($t = 5$ s), in spite of the severe oscillations in actuator force and velocity during the grasping process. This can be explained by the fact

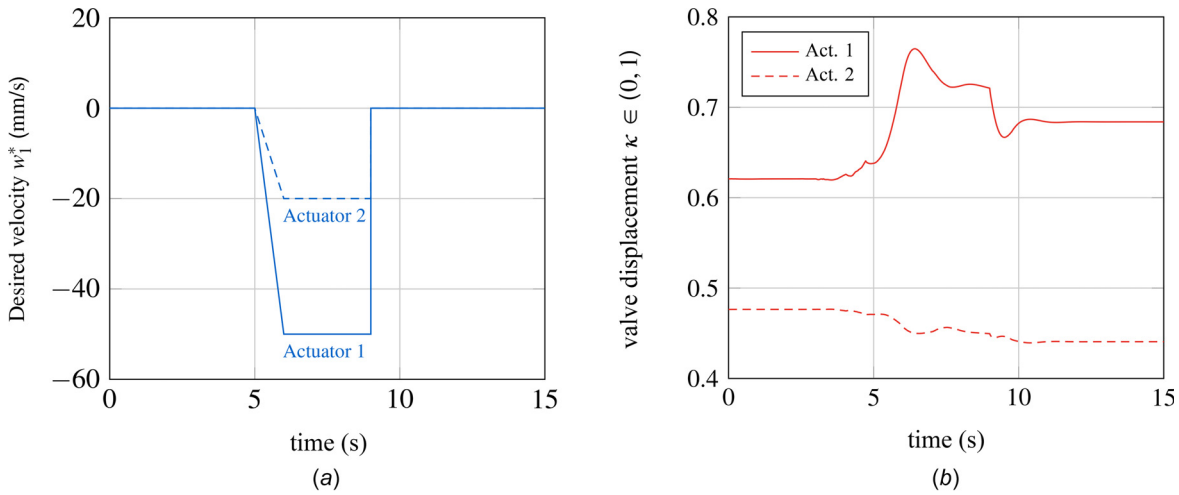

Fig. 10 Control law for the hydraulic crane with a gripper: (*a*) desired velocity of the controller, and (*b*) valve displacement as the output of the controller for the simulation using an interface model and step-size $h_{\mathcal{M}} = 16$ ms: (*a*) desired velocity and (*b*) valve displacement

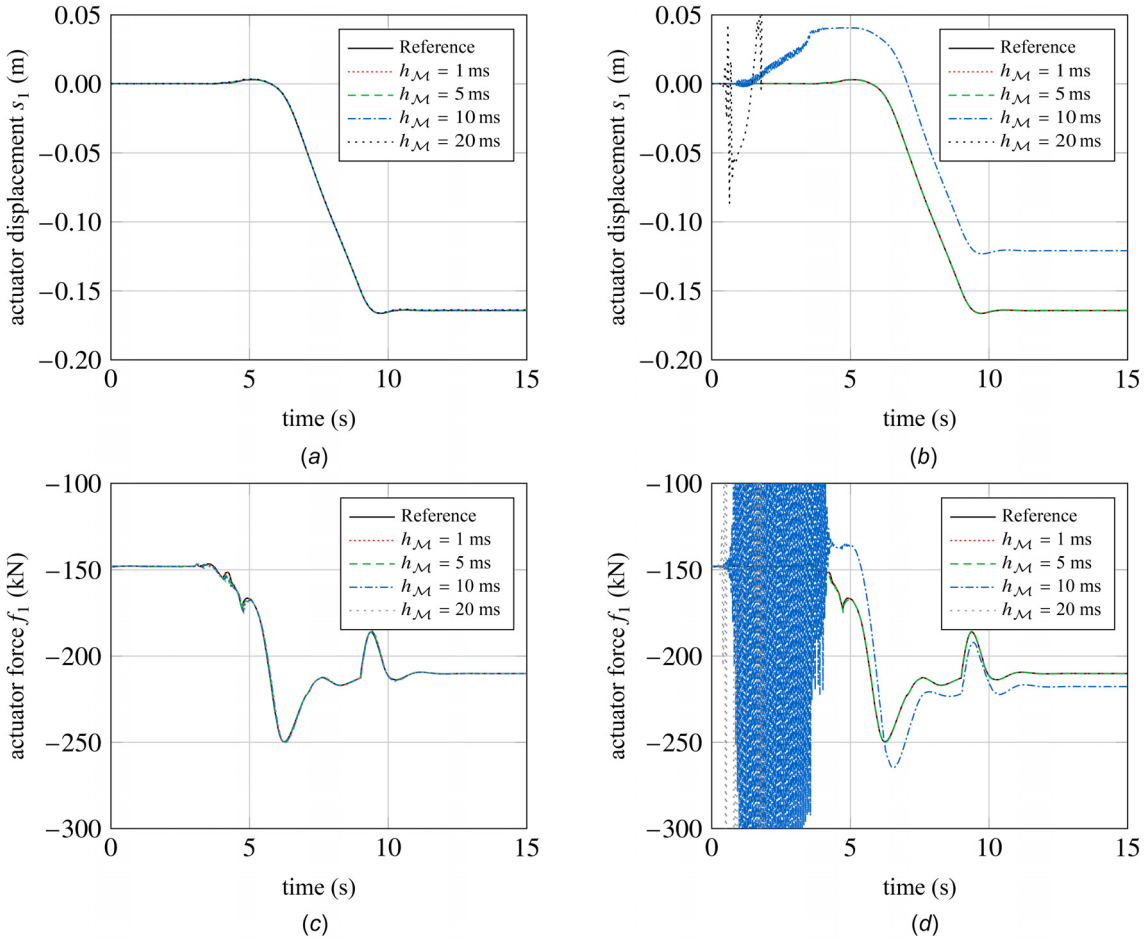

Fig. 11 Displacement of actuator 1 in the hydraulic crane using (*a*) an interface model, and (*b*) a zero-order hold. Actuator force with (*c*) interface model and (*d*) zero-order hold. Different step-sizes for the multibody system $h_{\mathcal{M}}$, and the same step-size for the hydraulics $h_{\mathcal{H}} = 0.2$ ms: (*a*) displacement with interface model, (*b*) displacement with zero-order hold, (*c*) force with interface model, and (*d*) force with zero-order hold.

that, when the claws completely grasp the log, the mass of the log becomes supported by the arm, which reduces the natural frequency of the system and makes its numerical integration more stable.

6 Discussion

In this section, the results from Sec. 5 are further analyzed and discussed. We assess the solution accuracy and stability of co-simulation setups using the proposed and other methods. Moreover, the computation details of the RIM are described, as well as its impact on simulation performance.

6.1 Accuracy and Stability. To quantify the accuracy of the RIM and compare it against other methods, we use the root-mean-square (RMS) of the absolute error. Therefore, the error associated with any variable x can be defined as

$$\mathrm{RMS}(x - x_{\mathrm{ref}}) = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(x(t_k) - x_{\mathrm{ref}}(t_k)\right)^2} \tag{55}$$

where x_{ref} is the reference solution, and N is the number of time-steps. Here, a zero-order hold co-simulation is used as reference solution, where both subsystems are integrated with the same step-size of 10^{-4} s, equal to the communication macrostep-size.

Figure 13 shows the error in the actuator displacement and velocity for the two examples presented in Sec. 5. Co-simulation results using both a ZOH and linear extrapolation for the interface variables are compared to the proposed method (RIM). In both examples, RIM-based co-simulation remains stable for all values of the step-size, while the other methods fail for large step-sizes. Although the displacement error is higher using the RIM for small step-sizes in example 1 (see Fig. 13(*a*)), it does not become unstable for larger step-sizes. Moreover, in Example 2, both the displacement and velocity errors are similar for the three methods (see Figs. 13(*c*) and 13(*d*)). But, in all the cases, the simulations using RIM can remain stable for large communication step-sizes.

6.2 Computational Performance. Using the RIM requires the computation of the effective mass and force terms in Eqs. (38) and (39), which involve the computation of the operator matrix $\mathbf{P}_{\mathrm{c}}$. These calculations can be time-consuming, especially for large systems, and some considerations are necessary for the implementation of the RIM. First, sparse matrix representations can help increase computational performance. Second, the order in which the matrices are multiplied also affects performance, and some terms might be reused in several steps along the calculation.

For the examples shown in this paper, the calculations were optimized to be able to integrate the coordinates and velocities of the RIM as described in Sec. 4.2. In this case, only the inverse of the effective mass is needed to compute the model, which requires one less inverse matrix operation. Moreover, the expression of the projector matrix $\mathbf{P}_{\mathrm{c}}$ can be used to simplify the expression of the inverse effective mass matrix so that Eq. (38) can be rewritten as

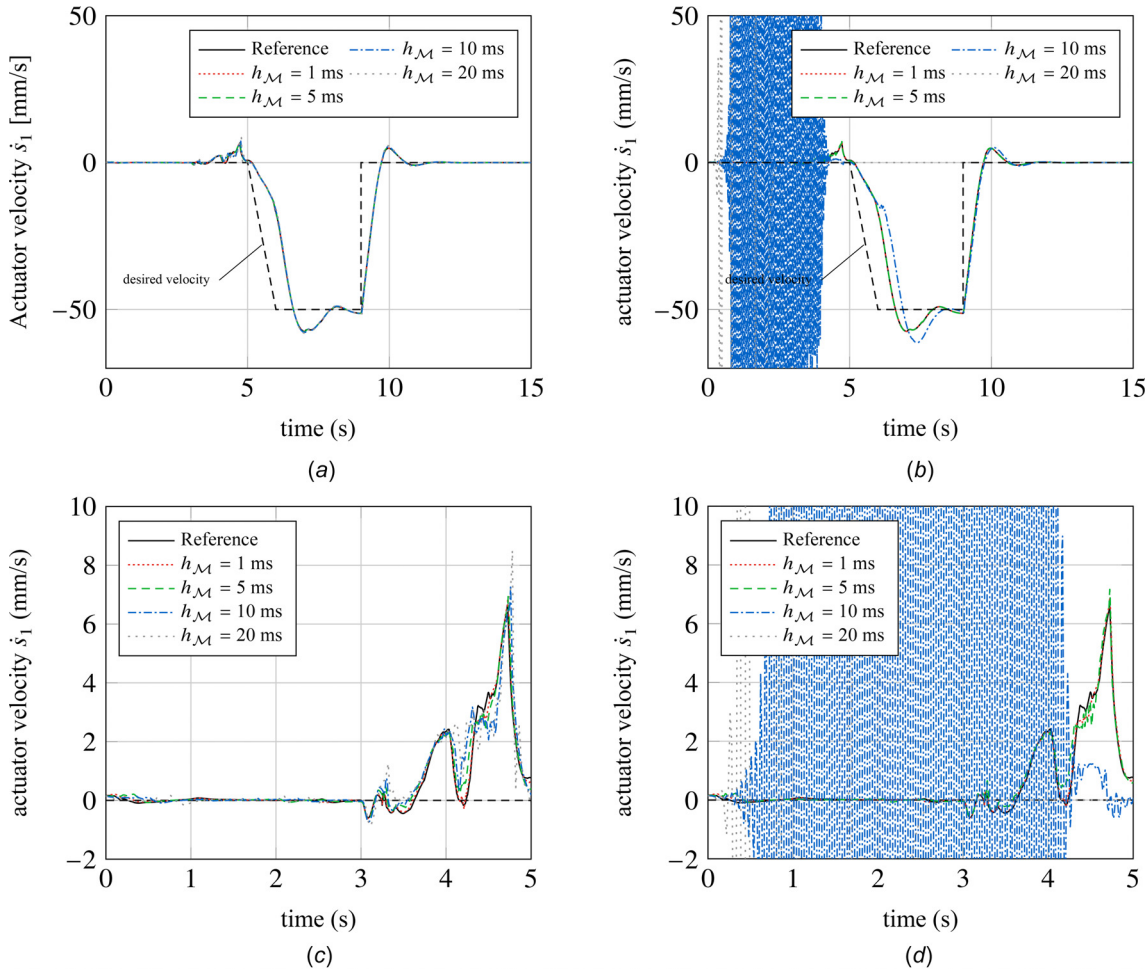

Fig. 12 Velocity of actuator 1 in the hydraulic crane using (*a*) an interface model, and (*b*) a zero-order hold. Detailed plots of the velocity during the grasping phase of the log $t \in [3, 5]$ s are shown in (*c*) and (*d*). Different step-sizes $h_{\mathcal{M}}$ were used for the multibody system, and the same step-size $h_{\mathcal{H}} = 0.2$ ms was used for the hydraulics: (*a*) velocity with interface model, (*b*) velocity with zero-order hold, (*c*) velocity detail with interface model, and (*d*) velocity detail with zero-order hold.

$$\tilde{\mathbf{M}}_{\mathrm{i}}^{-1} = \mathbf{H}_{\mathrm{ii}} - \mathbf{H}_{\mathrm{i}\alpha}\mathbf{H}_{\alpha\alpha}^{-1}\mathbf{H}_{\mathrm{i}\alpha}^{\mathrm{T}} \tag{56}$$

where these terms come from the lead matrix $\mathbf{H}$ in Eq. (35). Note that $\mathbf{H}_{\alpha\alpha}$ matrix in Eq. (56) is not explicitly inverted. Instead, the product $\mathbf{X} = \mathbf{H}_{\alpha\alpha}^{-1}\mathbf{H}_{\mathrm{i}\alpha}^{\mathrm{T}}$ is computed by solving the system $\mathbf{H}_{\alpha\alpha}\mathbf{X} = \mathbf{H}_{\mathrm{i}\alpha}^{\mathrm{T}}$, for which the LDLT decomposition was used, an efficient variant of the Cholesky factorization available in the Eigen library.

The average computational time of a macrotime-step for both examples with different step-sizes is shown in Fig. 14, where the total time is decomposed in three processes: the integration of the multibody subsystem, the integration of the hydraulics subsystem, and the computation of the RIM. The hydraulics time also includes the integration of the RIM according to Eq. (45), which appears to be negligible, and demonstrates that the proposed RIM can be simulated efficiently. These results also expose the overhead of the computation of the RIM, which is only computed once per time-step and does not depend on the time-step-size. However, it does depend on the size of the system and it is comparable to the computational cost associated with the integration of the multibody system. This computation overhead is largely dependent on the cost associated with the computation of the effective mass, which involves the solution of a linear system with a matrix $\mathbf{H}_{\alpha\alpha}$ using an LDLT decomposition. The complexity for such a decomposition is $\mathcal{O}(m_\alpha^3)$ for dense matrices, but can be reduced to $\mathcal{O}(bm_\alpha^2)$ in sparse matrix implementations, where the size of m_α is the size of $\mathbf{H}_{\alpha\alpha}$ (i.e., the number of active constraints) and b its bandwidth. Note that the maximum number of active constraints is $m_\alpha = 35$ in the first example, whereas $m_\alpha = 178$ in the second one. Although the computation time of the RIM is not negligible, a parallel implementation of the method can significantly reduce the computational cost for large-scale problems.

7 Conclusions

Reduced interface models provide a representation of the dynamics of slower subsystems, which can be used in co-simulation setups to obtain a dynamics-based prediction of the evolution of their inputs between communication points. The RIM for a mechanical system can be expressed in terms of the generalized velocities that define their interface with other subsystems in the co-simulation environment. The mechanical system can then be represented by effective mass matrix and effective force terms that describe its dynamics in the subspace defined by these interface velocities. One of the limitations of the method is that the RIM needs to be computed every macrotime-step, which can be computationally demanding in large multibody systems. However, this computation can be further optimized, for instance, by reusing some matrices that are obtained in the multibody time-step, or by

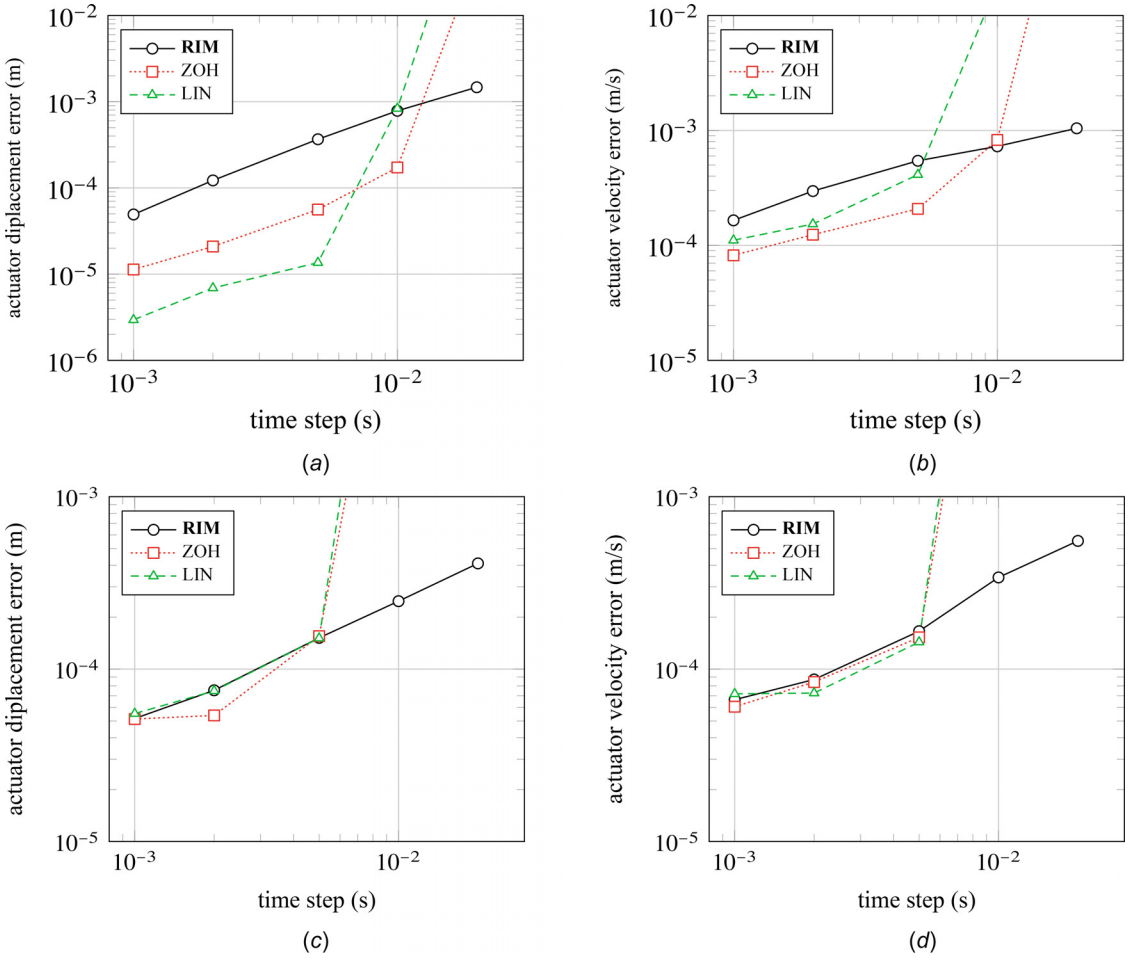

Fig. 13 Root-mean-square of the error in the actuator displacement and velocity for the hydraulic manipulator (*a* and *b*) and the hydraulic crane with gripper (*c* and *d*) in terms of the multibody subsystem time-step $h_{\mathcal{M}}$, with a step size of $h_{\mathcal{H}} = 0.2$ ms for the hydraulics subsystem. The data points corresponding to unstable simulations are not shown: (*a*) Example 1: hydraulic manipulator displacement (actuator 2), (*b*) velocity (actuator 2), (*c*) Example 2: hydraulic crane with gripper displacement (actuator 1), and (*d*) velocity (actuator 1).

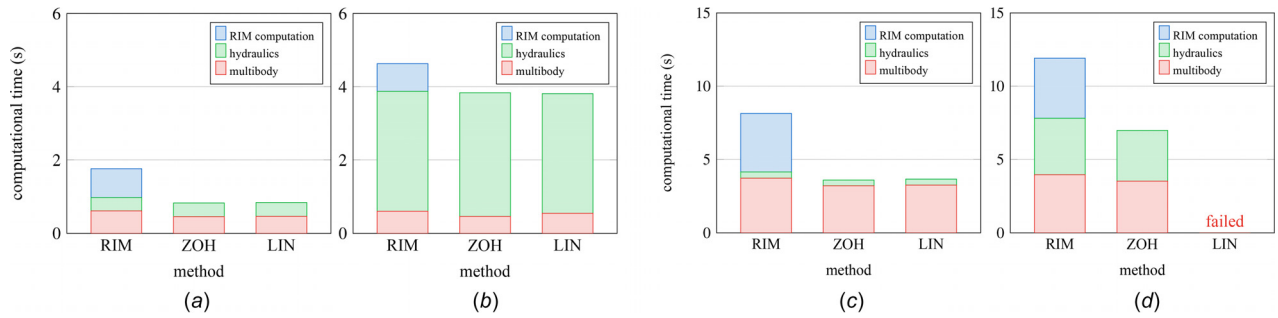

Fig. 14 Average computational time of the multibody and hydraulic subsystems and effective mass and force terms computation for the RIM in each macro time-step. Example 1: hydraulic manipulator (*a*) 1 ms (*b*) 10 ms and Example 2: hydraulic crane with gripper (*c*) 1 ms and (*d*) 10 ms.

performing computations in parallel. In this paper, the expression of those terms for mechanical systems subjected to unilateral contact and Coulomb friction has been put forward. This approach is compatible with an impulse-momentum formulation of the dynamics of the mechanical system in the form of a MLCP. Moreover, the proposed RIM was used in the multirate co-simulation of a mechanical system with hydraulic actuators. Results showed that the use of an RIM to predict the inputs of the fast subsystem between communication steps delivered more stable and accurate results than conventional extrapolation techniques.

Funding Data

- Natural Sciences and Engineering Research Council Canada (NSERC) (Funder ID: 10.13039/501100000038).

- CMLabs Simulations, Inc.
- Ministry of Economy of Spain through the Ramón y Cajal program (Contract No. RYC-2016-20222).

References

[1] O. A. Bauchau, 2011, *Flexible Multibody Dynamics*, Springer, Dordrecht, The Netherlands.

[2] Dopico, D., Luaces, A., González, M., and Cuadrado, J., 2011, "Dealing With Multiple Contacts in a Human-in-the-Loop Application," Multibody Syst. Dyn., **25**(2), pp. 167–183.

[3] Samin, J. C., Brüls, O., Collard, J. F., Sass, L., and Fisette, P., 2007, "Multiphysics Modeling and Optimization of Mechatronic Multibody Systems," Multibody Syst. Dyn., **18**(3), pp. 345–373.

[4] Rahikainen, J., Kiani, M., Sopanen, J., Jalali, P., and Mikkola, A., 2018, "Computationally Efficient Approach for Simulation of Multibody and Hydraulic Dynamics," Mech. Mach. Theory, **130**, pp. 435–446.

[5] Naya, M., Cuadrado, J., Dopico, D., and Lugris, U., 2011, "An Efficient Unified Method for the Combined Simulation of Multibody and Hydraulic Dynamics: Comparison With Simplified and Co-Integration Approaches," Arch. Mech. Eng., **58**(2), pp. 223–243.

[6] Gomes, C., Thule, C., Broman, D., Larsen, P. G., and Vangheluwe, H., 2018, "Co-Simulation: A Survey," ACM Comput. Surv., **51**(3), pp. 1–33.

[7] Benedikt, M., and Holzinger, F., 2016, "Automated Configuration for Non-Iterative Co-Simulation," Proceedings of the 17th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE), Montpellier, France, Apr. 18–20, pp. 1–7.

[8] Kübler, R., and Schiehlen, W., 2000, "Modular Simulation in Multibody System Dynamics," Multibody Syst. Dyn., **4**(2/3), pp. 107–127.

[9] Schweizer, B., Li, P., and Lu, D., 2015, "Explicit and Implicit Co-Simulation Methods: Stability and Convergence Analysis for Different Solver Coupling Approaches," ASME J. Comput. Nonlinear Dyn., **10**(5), p. 051007.

[10] Meyer, T., Li, P., Lu, D., and Schweizer, B., 2018, "Implicit Co-Simulation Method for Constraint Coupling With Improved Stability Behavior," Multibody Syst. Dyn., **44**(2), pp. 135–161.

[11] Schweizer, B., Li, P., and Lu, D., 2016, "Co-Simulation Method for Solver Coupling With Algebraic Constraints Incorporating Relaxation Techniques," Multibody Syst. Dyn., **36**(1), pp. 1–36.

[12] González, F., Naya, M. A., Luaces, A., and González, M., 2011, "On the Effect of Multi-Rate Co-Simulation Techniques in the Efficiency and Accuracy of Multibody System Dynamics," Multibody Syst. Dyn., **25**(4), pp. 461–483.

[13] Oberschelp, O., and Vöcking, H., 2004, "Multirate Simulation of Mechatronic Systems," Proceedings of the IEEE International Conference on Mechatronics, Istanbul, Turkey, June 3–5, pp. 404–409.

[14] Arnold, M., 2009, "Numerical Methods for Simulation in Applied Dynamics," *Simulation Techniques for Applied Dynamics*, Springer, Vienna, Austria, pp. 191–246.

[15] Busch, M., 2016, "Continuous Approximation Techniques for Co-Simulation Methods: Analysis of Numerical Stability and Local Error," ZAMM—J. Appl. Math. Mech., **96**(9), pp. 1061–1081.

[16] Ben Khaled-El Feki, A., Duval, L., Faure, C., Simon, D., and Gaid, M. B., 2017, "CHOPtrey: Contextual Online Polynomial Extrapolation for Enhanced Multi-Core Co-Simulation of Complex Systems," Simulation, **93**(3), pp. 185–200.

[17] Benedikt, M., Watzenig, D., Zehetner, J., and Hofer, A., 2013, "NEPCE—A Nearly Energy-Preserving Coupling Element for Weak-Coupled Problems and Co-Simulation," International Conference on Computational Methods for Coupled Problems in Science and Engineering, Ibiza, Spain, June 17–19, pp. 1–12.

[18] Sadjina, S., Kyllingstad, L. T., Skjong, S., and Pedersen, E., 2017, "Energy Conservation and Power Bonds in Co-Simulations: Non-Iterative Adaptive Step Size Control and Error Estimation," Eng. Comput., **33**(3), pp. 607–620.

[19] Schweizer, B., and Lu, D., 2014, "Semi-Implicit co-Simulation Approach for Solver Coupling," Arch. Appl. Mech., **84**(12), pp. 1739–1769.

[20] Haid, T., Stettinger, G., Watzenig, D., and Benedikt, M., 2018, "A Model-Based Corrector Approach for Explicit co-Simulation Using Subspace Identification," Proceedings of the Fifth Joint International Conference on Multibody System Dynamics, Lisbon, Portugal, June 24–28, pp. 1–18.

[21] González, F., Arbatani, S., Mohtat, A., and Kövecses, J., 2019, "Energy-Leak Monitoring and Correction to Enhance Stability in the Co-Simulation of Mechanical Systems," Mech. Mach. Theory, **131**, pp. 172–188.

[22] Peiret, A., González, F., Kövecses, J., and Teichmann, M., 2018, "Multibody System Dynamics Interface Modelling for Stable Multirate Co-Simulation of Multiphysics Systems," Mech. Mach. Theory, **127**, pp. 52–72.

[23] Kövecses, J., 2008, "Dynamics of Mechanical Systems and the Generalized Free-Body Diagram—Part I: General Formulation," ASME J. Appl. Mech., **75**(6), pp. 1–12.

[24] Stewart, D. E., and Trinkle, J. C., 1996, "An Implicit Time-Stepping Scheme for Rigid Body Dynamics With Inelastic Collisions and Coulomb Friction," Int. J. Numer. Methods Eng., **39**(15), pp. 2673–2691.

[25] Erleben, K., 2007, "Velocity-Based Shock Propagation for Multibody Dynamics Animation," ACM Trans. Graph., **26**(2), p. 12.

[26] Glocker, C., 2001, *Set-Valued Force Laws*, Springer, Troy, NY.

[27] Stewart, D. E., 1998, "Convergence of a Time-Stepping Scheme for Rigid-Body Dynamics and Resolution of Painlevé's Problem," Arch. Ration. Mech. Anal., **145**(3), pp. 215–260.

[28] Moreau, J., 1966, "Quadratic Programming in Mechanics: Dynamics of One Sided Constraints," SIAM J. Control, **4**(1), pp. 153–158.

[29] Anitescu, M., and Tasora, A., 2010, "An Iterative Approach for Cone Complementarity Problems for Nonsmooth Dynamics," Comput. Optim. Appl., **47**(2), pp. 207–235.

[30] Anitescu, M., and Potra, F. A., 1997, "Formulating Dynamic Multi-Rigid-Body Contact Problems With Friction as Solvable Linear Complementarity Problems," Nonlinear Dyn., **14**(3), pp. 231–247.

[31] Júdice, J., and Pires, F., 1992, "Basic-Set Algorithm for a Generalized Linear Complementarity Problem," J. Optim. Theory Appl., **74**(3), pp. 391–411.

[32] Júdice, J. J., 1994, "Algorithms for Linear Complementarity Problems," Algorithms Contin. Optim., **434**, pp. 435–474.

[33] Lemke, C., 1968, "On Complementary Pivot Theory," *Mathematics of the Decision Sciences* (Lectures Applied Mathematics, Vol. 2), American Mathematical Society, Providence, RI, pp. 95–114.

[34] Acary, V., Cadoux, F., Lemaréchal, C., and Malick, J., 2011, "A Formulation of the Linear Discrete Coulomb Friction Problem Via Convex Optimization," ZAMM-J. Appl. Math. Mech., **91**(2), pp. 155–175.

[35] Acary, V., and Brogliato, B., 2008, *Numerical Methods for Nonsmooth Dynamical Systems: Applications in Mechanics and Electronics*, Springer Science & Business Media, Springer, Berlin.

[36] Lacoursière, C., 2006, "A Regularized Time Stepper for Multibody Systems," Umeå University, Umeå, Sweden, Report No. UMINF-04.

[37] Anitescu, M., and Potra, F. A., 2002, "A Time-Stepping Method for Stiff Multibody Dynamics With Contact and Friction," Int. J. Numer. Methods Eng., **55**(7), pp. 753–784.

[38] CM Labs Simulations Inc., 2020, Vortex Studio Documentation. Montréal, QC, Canada.

[39] Cardona, A., and Geradin, M., 1990, "Modeling of a Hydraulic Actuator in Flexible Machine Dynamics Simulation," Mech. Mach. Theory, **25**(2), pp. 193–207.