Proceedings of the ASME 2019 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2019
August 18-21, 2019, Anaheim, CA, USA

IDETC2019-98098

INTERFACE MODELS FOR MULTIRATE CO-SIMULATION OF NONSMOOTH MULTIBODY SYSTEMS

Albert Peiret
József Kövecses
Dept. of Mechanical Engineering and
Centre for Intelligent Machines
McGill University
Montréal, Canada
Email: albert.peiret@mail.mcgill.ca
Email: jozsef.kovecses@mcgill.ca

Francisco González
Laboratorio de Ingeniería Mecánica
University of A Coruña
Ferrol, Spain
Email: f.gonzalez@udc.es

Marek Teichmann
CM Labs Simulations
Montréal, Canada
Email: marek@cm-labs.com

ABSTRACT

Co-simulation techniques enable the coupling of physically diverse subsystems in an efficient and modular way. Complex engineering applications can be simulated in co-simulation setups, in which each subsystem is solved and integrated using numerical methods tailored to its physical behaviour. Co-simulation implies that the 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 are generally not accessible to the rest of the simulation environment. In non-iterative co-simulation schemes, this may lead to the instability of the integration. Increasingly demanding requirements in the simulation of machinery have led to the coupling, in real-time co-simulation setups, of multibody models of mechanical systems to computational representations of non-mechanical subsystems, such as hydraulics and electronics. Often, these feature faster dynamics than their mechanical counterparts, which leads to the use of multirate integration in non-iterative co-simulation environments. The stability of the integration in these cases can be enhanced using interface models, i.e., reduced representations of the multibody system, to provide meaningful input values to faster subsystems between communication points. This work describes such interface models that can be used to represent non-smooth mechanical systems subjected to unilateral contact and friction.

INTRODUCTION

Predictive simulation of engineering systems is currently 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 with this method, as well as the expectations about its performance and the results that it is able to deliver. This is also the case with forward-dynamics simulation of multibody systems. Presently, multibody simulations are able to deal with challenging phenomena such as flexibility, contacts, and friction in an efficient way [1]; 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 in 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 [2,3], showing good stability and efficiency properties [4]. Co-simulation, on the other hand, consists in modelling and integrating separately the different subsystems in an engineering application. The dynamics of each of them can then be formulated and solved using meth-

ods especially suited to its physical behaviour. The subsystem solvers only exchange information, i.e., their *coupling variables*, through a minimum interface at discrete-time communication points; otherwise, the numerical integration of each subsystem proceeds independently from the others. The time interval between communication points is referred to as *macro-step*. This modular approach makes co-simulation environments particularly suitable for collaborative projects, as it enables each partner to use its own modelling and solution methods regardless of the implementation chosen by others. Moreover, the internal details of each subsystem remain hidden, 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 [5]. However, the discrete-time communication between subsystems gives rise to coupling errors and discontinuities that may compromise the stability of the integration process and the accuracy of the results [6].

In general, iterative coupling schemes have been shown to exhibit a more stable behaviour than their non-iterative counterparts [7]. These schemes update the input variables in each iteration and subsequently retake the integration step of one or more subsystems. There exist applications, however, in which restarting the integration from a previous state is not possible, either because certain subsystems do not allow such an operation, or because the available time to carry out the computations is limited. In these cases, explicit, non-iterative coupling approaches must be used; Jacobi-schemes are a common choice in non-iterative co-simulation, as they permit the parallelization of subsystem integration. In a Jacobi scheme, the subsystems exchange inputs at the beginning of a macro-step and proceed with their integration independently until the next communication point.

Keeping non-iterative co-simulation schemes stable is challenging, especially if subsystems have direct feed-through, i.e., their outputs explicitly depend on their inputs [7]. In some applications it may be difficult to know whether this is the case, because information about subsystem internals may not be available and the subsystems behave effectively as *black boxes*. Also, even if the integration process remains stable, coupling errors at the interface may cause the simulation results to be inaccurate [8]. Extrapolation and approximation of subsystem inputs are strategies commonly used to improve non-iterative schemes [9] ; adaptive extrapolation techniques can be used as well [10]. Another possibility is to act on the coupling variables or the communication step-size to maintain the energy balance at the interface [11].

The above mentioned methods only require the information contained in the coupling variables to operate. The availability of additional data about subsystem internals, 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 [12] if they are known; otherwise it is also possible to estimate them using subspace identification algorithms [13]. If the information exchanged between subsystems includes details about their internal energy, monitoring and correction algorithms can be employed to improve the energy balance of the whole system [14].

Interface models (IM) represent a recently developed approach to enhance the explicit co-simulation of multibody systems in multirate environments [15]. Multibody systems are often coupled to other subsystems with faster dynamics, such as hydraulics and electronics, which take more than one integration step between communication points. The inputs that the multibody subsystem provides to these, however, cannot be updated until the next exchange of coupling variables, and this may result in instabilities and inaccurate results, even if extrapolation techniques are used. The IM is intended to provide these faster subsystems with a physics-based prediction of the evolution of their inputs during the macro-step. The IM is obtained from the characterization of the interaction between the mechanical subsystem and its environment, which can be done through an interface described by a set of generalized velocities. 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 [16], in which the dynamics of the whole multibody system can be decomposed. The IM thus obtained constitutes a reduced order expression of the dynamics of the mechanical system, represented by an effective mass matrix matrix and an effective force term.

The concept of IM was introduced in [15] for mechanical systems with smooth dynamics formulated at the acceleration level. The research in this paper extends the use of IM to nonsmooth mechanical systems, such as those with unilateral contact, impacts, and dry friction. The dynamics of such systems is often formulated at the impulse-momentum level using time-stepping schemes [17, 18]. These methods can handle collisions seamlessly and remove inconsistencies and indeterminacies caused by friction that affect their acceleration-based counterparts [19, 20]. Although the system dynamics can be formulated as a *linear complementarity problem* (LCP) for the frictionless case [21], considering Coulomb friction at the contact points leads to *nonlinear complementarity problems* (NCP). 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 LCP [19,22,23].Formulating the dynamics as an LCP is advantageous due to the notable amount of solver algorithms available in the literature [24].

In this paper, we develop the IM for nonsmooth mechanical systems whose dynamics is formulated at the impulse-momentum level as a mixed linear complementarity problem. This reformulation is compatible with unilateral contact with friction and able to model stick-slip transitions. The proposed IM was tested in numerical experiments with hydraulically ac-

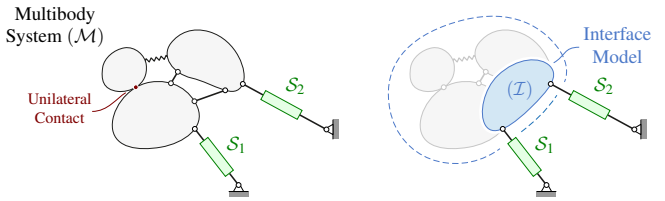

FIGURE 1. Interfacing a multibody system $\mathcal{M}$ with a subsystem $\mathcal{S}$ using an interface model $\mathcal{I}$.

tuated mechanical systems. Results confirmed the ability of the IM to enhance the stability and accuracy of non-iterative multirate co-simulation setups with multiphysics components.

CO-SIMULATION WITH INTERFACE MODELS

Let us consider a multibody system $\mathcal{M}$ that interacts with another subsystem $\mathcal{S}$, which can also be constituted of many 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 with 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 is exchanged at the beginning of each macro time-step, of size H, following a Jacobi scheme. Therefore, the subsystems do not receive any information from each other until the following communication point. For this reason, the input values of the subsystem $\mathbf{u}_\mathcal{S}$ must be extrapolated in some way within the macro time-step.

Interface models (IM) can be used to provide a physics-based prediction of the inputs of the fast subsystem, $\mathbf{u}_\mathcal{S}$, until the next communication point is reached. Instead of extrapolating these inputs from the time-history of the coupling variables, IMs approximate the outputs of the multibody system $\mathbf{y}_\mathcal{M}$ in terms of its currently available inputs $\mathbf{u}_\mathcal{M}$ and its dynamics.

The introduction of an IM 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 IM, denoted by $\mathcal{I}$. Both $\mathcal{I}$ and $\mathcal{S}$ are integrated at a

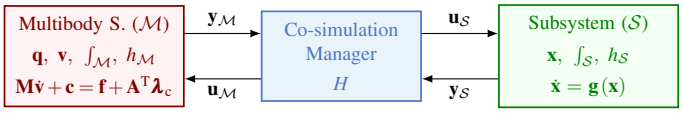

FIGURE 2. Block diagram of a multibody system $\mathcal{M}$ and subsystem $\mathcal{S}$ coupled in a co-simulation setup.

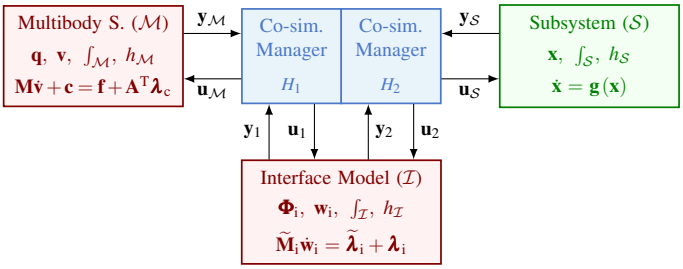

FIGURE 3. Block diagram of a multibody system $\mathcal{M}$ and subsystem $\mathcal{S}$ coupled in a co-simulation setup via an interface model $\mathcal{I}$ of the multibody system.

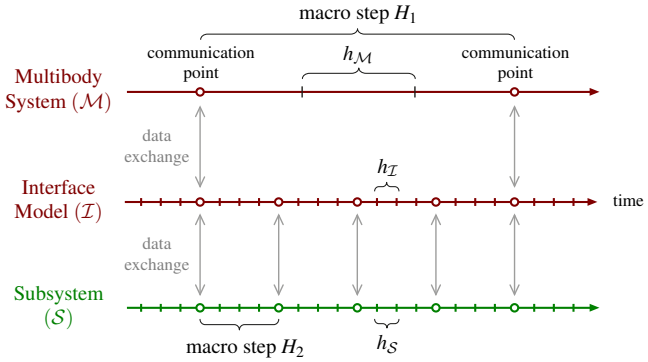

FIGURE 4. Timeline of Jacobi-scheme co-simulation using an interface model $\mathcal{I}$ of the multibody system $\mathcal{M}$.

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 system every macro time-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 IM.

The configuration and velocity of a multibody system can be parametrized 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 parametrized by a set of m_c velocity components $\mathbf{w}_c$, which can be related to the generalized velocities as

$$\mathbf{w}_c = \mathbf{A}\mathbf{v} \tag{1}$$

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

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

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 $\boldsymbol{\lambda}_c$ is the $m_c \times 1$ array of constraint forces. The generalized forces can be decomposed as $\mathbf{f} = \mathbf{f}_a + \mathbf{f}_i$, where $\mathbf{f}_a$ contains the generalized applied forces, and $\mathbf{f}_i$ 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

$$\mathbf{w}_i = \mathbf{D}\mathbf{v} \tag{3}$$

where $\mathbf{D}(\mathbf{q})$ is the $m_i \times n$ interface Jacobian matrix. In some cases, the interface velocity can be defined as time derivatives of the interface coordinates $\boldsymbol{\Phi}_i(\mathbf{q})$, as $\mathbf{w}_i = \dot{\boldsymbol{\Phi}}_i$. Furthermore, the generalized interface forces $\mathbf{f}_i$ can be expressed in terms of m_i interface forces $\boldsymbol{\lambda}_i$ as

$$\mathbf{f}_i = \mathbf{D}^{\mathrm{T}}\boldsymbol{\lambda}_i. \tag{4}$$

If a *force-displacement coupling* is selected to conduct the co-simulation of multibody systems, these interface forces are usually provided by the external subsystem $\mathcal{S}$, and so $\boldsymbol{\lambda}_i$ is part of the input of the multibody system $\mathcal{M}$. In this case, the interface kinematics $\boldsymbol{\Phi}_i$ and $\mathbf{w}_i$ are the output $\mathbf{y}_{\mathcal{M}}$. Nevertheless, the input $\mathbf{u}_{\mathcal{M}}$ can be the interface kinematics instead, so that $\boldsymbol{\Phi}_i$ and $\mathbf{w}_i$ are given by $\mathcal{S}$ are need to be enforced through kinematic constraints.

The *interface model* $\mathcal{I}$ can be obtained by expressing the dynamics of the multibody system $\mathcal{M}$ in terms of the interface velocities $\mathbf{w}_i$. If all constraints are bilateral, the dynamics of the *interface model* $\mathcal{I}$ can be expressed as [15]

$$\widetilde{\mathbf{M}}_i\dot{\mathbf{w}}_i = \widetilde{\boldsymbol{\lambda}}_i + \boldsymbol{\lambda}_i \tag{5}$$

where the *effective mass* $\widetilde{\mathbf{M}}_i$ and *effective force* $\widetilde{\boldsymbol{\lambda}}_i$ terms are

$$\widetilde{\mathbf{M}}_i = \left(\mathbf{D}(\mathbf{I}-\mathbf{P}_c)\mathbf{M}^{-1}\mathbf{D}^{\mathrm{T}}\right)^{-1} \tag{6}$$

$$\widetilde{\boldsymbol{\lambda}}_i = \widetilde{\mathbf{M}}_i\left(\mathbf{D}(\mathbf{I}-\mathbf{P}_c)\mathbf{M}^{-1}(\mathbf{f}_a-\mathbf{c}) + \dot{\mathbf{D}}\mathbf{v} + \mathbf{D}\mathbf{P}_c\dot{\mathbf{v}}\right) \tag{7}$$

The projector matrix $\mathbf{P}_c = \mathbf{M}^{-1}\mathbf{A}^{\mathrm{T}}\left(\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^{\mathrm{T}}\right)^{-1}\mathbf{A}$ 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 IM needs to be reformulated in order to account for contact detachment and stick-slip transitions.

NONSMOOTH MULTIBODY SYSTEM DYNAMICS

In general, we can consider three different kinds of interactions: bilateral, unilateral, and friction. Then, the constraint forces $\boldsymbol{\lambda}_c$ and constraint velocities $\mathbf{w}_c$ can be arranged as

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

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 forces and m_t tangential contact forces, respectively. Here, for each contact point, one normal component and two tangential components along two orthogonal directions on the tangent plane are used.

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

$$\boldsymbol{\lambda}_c^{\mathrm{lo}} \leqslant \boldsymbol{\lambda}_c \leqslant \boldsymbol{\lambda}_c^{\mathrm{up}} \tag{9}$$

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

$$\mathbf{0} \leqslant \boldsymbol{\lambda}_n \leqslant +\boldsymbol{\infty}. \tag{10}$$

Upper and lower bounds can be defined for friction forces using a faceted approximation of the friction cone [18, 19, 22]. If μ_j is the friction coefficient, then for each of the two friction components of the j-th contact point, $-\mu_j\lambda_{n_j} \leqslant \lambda_{t_{j,1}}, \lambda_{t_{j,2}} \leqslant +\mu_j\lambda_{n_j}$. Usually, the dependency of the bound on the normal force λ_{n_j} is not enforced explicitly in the formulation, but rather through iterations [18], or using values from previous time instants [23].

Constraints can only be satisfied if the forces are within bounds. Otherwise, when a force reaches a bound, the kinematic constraint becomes unilateral, which can be defined through

complementarity as

$$\left.\begin{aligned} \mathbf{0} \leqslant \mathbf{w}_\mathrm{c}^\mathrm{lo} \;\perp\; \left(\boldsymbol{\lambda}_\mathrm{c} - \boldsymbol{\lambda}_\mathrm{c}^\mathrm{lo}\right) \geqslant \mathbf{0} \\ \mathbf{0} \leqslant \mathbf{w}_\mathrm{c}^\mathrm{up} \;\perp\; \left(\boldsymbol{\lambda}_\mathrm{c}^\mathrm{up} - \boldsymbol{\lambda}_\mathrm{c}\right) \geqslant \mathbf{0} \end{aligned}\right\} \tag{11}$$

where $\mathbf{w}_\mathrm{c} = \mathbf{w}_\mathrm{c}^\mathrm{lo} - \mathbf{w}_\mathrm{c}^\mathrm{up}$ is the constraint velocity decomposed into positive and negative components, and $\perp$ denotes componentwise complementarity. For unilateral constraints, this complementarity conditions becomes

$$\mathbf{0} \leqslant \mathbf{w}_\mathrm{n} \;\perp\; \boldsymbol{\lambda}_\mathrm{n} \geqslant \mathbf{0}. \tag{12}$$

Here, this complementarity condition will be also written in a more compact form as

$$\mathbf{w}_\mathrm{c} \;\perp\; \boldsymbol{\lambda}_\mathrm{c} \in \left[\boldsymbol{\lambda}_\mathrm{c}^\mathrm{lo}, \boldsymbol{\lambda}_\mathrm{c}^\mathrm{up}\right]. \tag{13}$$

The Mixed Linear Complementarity Problem

The dynamic equations can be formulated at the impulse-momentum level in order to include the complementarity conditions. This is possible by using a finite difference approximation of both the generalized acceleration $\dot{\mathbf{v}} = (\mathbf{v}^+ - \mathbf{v})/h$, and the constraint acceleration $\dot{\mathbf{w}}_\mathrm{c} = (\mathbf{w}_\mathrm{c}^+ - \mathbf{w}_\mathrm{c})/h$, where h is the step size. Then, the constraint velocities in Eq. (1), together with the dynamic equations in Eq. (2), and the complementarity conditions in Eq. (11), formulate the *mixed linear complementarity problem* (MLCP)

$$\left.\begin{aligned} \begin{bmatrix} \mathbf{M} & -\mathbf{A}^\mathrm{T} \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}^+ \\ h\boldsymbol{\lambda}_\mathrm{c}^+ \end{bmatrix} + \begin{bmatrix} h(\mathbf{c}-\mathbf{f}) - \mathbf{M}\mathbf{v} \\ h\dot{\mathbf{A}}\mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{w}_\mathrm{c}^+ \end{bmatrix} \\ \mathbf{w}_\mathrm{c}^+ \;\perp\; \boldsymbol{\lambda}_\mathrm{c}^+ \in \left[\boldsymbol{\lambda}_\mathrm{c}^\mathrm{lo}, \boldsymbol{\lambda}_\mathrm{c}^\mathrm{up}\right] \end{aligned}\right\} \tag{14}$$

where $\mathbf{v}$ and $\mathbf{v}^+$ are the generalized velocities at the beginning and at the end of the time-step, respectively; $\mathbf{w}_\mathrm{c}$ and $\mathbf{w}_\mathrm{c}^+$ are the constraint velocity at the beginning and at the end of the step, repsectively, and $h\boldsymbol{\lambda}_\mathrm{c}^+$ are the unknown constraint impulses of the step. The derivative of the Jacobian matrix $\dot{\mathbf{A}}(\mathbf{q},\mathbf{v})$ is computed using the known $\mathbf{q}$ and $\mathbf{v}$ at the beginning of the step.

Furthermore, eliminating $\mathbf{v}^+$ in Eq. (14) one obtains

$$\underbrace{\left(\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^\mathrm{T}\right)}_{\mathbf{H}} h\boldsymbol{\lambda}_\mathrm{c}^+ + \underbrace{\mathbf{A}\mathbf{M}^{-1}\left(h(\mathbf{f}-\mathbf{c}) + \mathbf{M}\mathbf{v}\right) + h\dot{\mathbf{A}}\mathbf{v}}_{\mathbf{b}} = \mathbf{w}_\mathrm{c}^+ \tag{15}$$

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

$$\mathbf{H}h\boldsymbol{\lambda}_\mathrm{c}^+ + \mathbf{b} = \mathbf{w}_\mathrm{c}^+ \quad \perp \quad \boldsymbol{\lambda}_\mathrm{c}^+ \in \left[\boldsymbol{\lambda}_\mathrm{c}^\mathrm{lo}, \boldsymbol{\lambda}_\mathrm{c}^\mathrm{up}\right] \tag{16}$$

where $\mathbf{H}$ is the $m_\mathrm{c} \times m_\mathrm{c}$ matrix that represents the inverse effective mass of the system in the constraint space, and is positive definite if the constraints are not redundant. Otherwise, matrix $\mathbf{H}$ becomes positive semi-definite and constraint force indeterminacy might occur.

INTERFACE MODELS OF NONSMOOTH SYSTEMS

The interface velocities $\mathbf{w}_\mathrm{i}$ and forces $\boldsymbol{\lambda}_\mathrm{i}$ from Eqs. (3) and (4), can be incorporated into Eq. (14), so that the MLCP can be rearranged as

$$\left.\begin{aligned} \begin{bmatrix} \mathbf{M} & -\mathbf{A}^\mathrm{T} & -\mathbf{D}^\mathrm{T} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{D} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}^+ \\ h\boldsymbol{\lambda}_\mathrm{c}^+ \\ h\boldsymbol{\lambda}_\mathrm{i}^+ \end{bmatrix} + \begin{bmatrix} h(\mathbf{c}-\mathbf{f}_\mathrm{a}) - \mathbf{M}\mathbf{v} \\ h\dot{\mathbf{A}}\mathbf{v} \\ h\dot{\mathbf{D}}\mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{w}_\mathrm{c}^+ \\ \mathbf{w}_\mathrm{i}^+ \end{bmatrix} \\ \mathbf{w}_\mathrm{c}^+ \;\perp\; \boldsymbol{\lambda}_\mathrm{c}^+ \in \left[\boldsymbol{\lambda}_\mathrm{c}^\mathrm{lo}, \boldsymbol{\lambda}_\mathrm{c}^\mathrm{up}\right] \end{aligned}\right\} \tag{17}$$

Depending on the coupling approach, either $\mathbf{w}_\mathrm{i}^+$ or $\boldsymbol{\lambda}_\mathrm{i}^+$ should be known when solving the MLCP. However, to derive an interface model we need the relation between the two in order to predict the output of the multibody system independently from the chosen coupling approach. First, we eliminate the velocities $\mathbf{v}^+$ from the first row by substituting them into the other two rows, so that the reduced version of the MLCP in Eq. (17) can be written as

$$\left.\begin{aligned} \begin{bmatrix} \mathbf{A}\mathbf{M}^{-1}\mathbf{A}^\mathrm{T} & \mathbf{A}\mathbf{M}^{-1}\mathbf{D}^\mathrm{T} \\ \mathbf{D}\mathbf{M}^{-1}\mathbf{A}^\mathrm{T} & \mathbf{D}\mathbf{M}^{-1}\mathbf{D}^\mathrm{T} \end{bmatrix} \begin{bmatrix} h\boldsymbol{\lambda}_\mathrm{c}^+ \\ h\boldsymbol{\lambda}_\mathrm{i}^+ \end{bmatrix} + \begin{bmatrix} \mathbf{b}_\mathrm{c} \\ \mathbf{b}_\mathrm{i} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_\mathrm{c}^+ \\ \mathbf{w}_\mathrm{i}^+ \end{bmatrix} \\ \mathbf{w}_\mathrm{c}^+ \;\perp\; \boldsymbol{\lambda}_\mathrm{c}^+ \in \left[\boldsymbol{\lambda}_\mathrm{c}^\mathrm{lo}, \boldsymbol{\lambda}_\mathrm{c}^\mathrm{up}\right] \end{aligned}\right\} \tag{18}$$

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

Once the MLCP is solved, the constraint forces are determined, and we can distinguish between *tight* and *active* 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, and also for the interface model. Here, we assume that tight constraint forces will not change their value, while active constraint forces may, since they have not reached a bound.

Let us rearrange the constraint force and velocity arrays into tight and active as

$$\boldsymbol{\lambda}_\mathrm{c} = \begin{bmatrix} \boldsymbol{\lambda}_\alpha \\ \boldsymbol{\lambda}_\tau \end{bmatrix} \qquad \text{and} \qquad \mathbf{w}_\mathrm{c} = \begin{bmatrix} \mathbf{w}_\alpha \\ \mathbf{w}_\tau \end{bmatrix} = \begin{bmatrix} \mathbf{A}_\alpha \\ \mathbf{A}_\tau \end{bmatrix} \mathbf{v} \tag{19}$$

where $\boldsymbol{\lambda}_\alpha$ and $\mathbf{w}_\alpha$ are active, and $\boldsymbol{\lambda}_\tau$ and $\mathbf{w}_\tau$ are tight. Then,

Eq. (18) can also be rearranged as

$$\left[\begin{array}{cc|c} \mathbf{H}_{\alpha\alpha} & \mathbf{H}_{\alpha\tau} & \mathbf{H}_{\alpha\mathrm{i}} \\ \mathbf{H}_{\tau\alpha} & \mathbf{H}_{\tau\tau} & \mathbf{H}_{\tau\mathrm{i}} \\ \hline \mathbf{H}_{\mathrm{i}\alpha} & \mathbf{H}_{\mathrm{i}\tau} & \mathbf{H}_{\mathrm{ii}} \end{array}\right] \left[\begin{array}{c} h\boldsymbol{\lambda}_{\alpha}^{+} \\ h\boldsymbol{\lambda}_{\tau}^{+} \\ \hline h\boldsymbol{\lambda}_{\mathrm{i}}^{+} \end{array}\right] + \left[\begin{array}{c} \mathbf{b}_{\alpha} \\ \mathbf{b}_{\tau} \\ \hline \mathbf{b}_{\mathrm{i}} \end{array}\right] = \left[\begin{array}{c} \mathbf{w}_{\alpha}^{+} \\ \mathbf{w}_{\tau}^{+} \\ \hline \mathbf{w}_{\mathrm{i}}^{+} \end{array}\right] \tag{20}$$

Tight constraint forces are known and are assumed to be constant (i.e., $\boldsymbol{\lambda}_{\tau}^{+} = \boldsymbol{\lambda}_{\tau}^{\mathrm{lo}}$ or $\boldsymbol{\lambda}_{\tau}^{\mathrm{up}}$). Active constraint forces are considered within bounds, and so complementarity in Eq. (11) 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

$$\begin{aligned} \Big(\mathbf{H}_{\mathrm{ii}} - \mathbf{H}_{\mathrm{i}\alpha}\mathbf{H}_{\alpha\alpha}^{-1}\mathbf{H}_{\alpha\mathrm{i}}\Big)h\boldsymbol{\lambda}_{\mathrm{i}}^{+} + \\ \mathbf{b}_{\mathrm{i}} + \mathbf{H}_{\mathrm{i}\tau}h\boldsymbol{\lambda}_{\tau}^{+} - \mathbf{H}_{\mathrm{i}\alpha}\mathbf{H}_{\alpha\alpha}^{-1}(\mathbf{b}_{\alpha} + \mathbf{H}_{\alpha\tau}h\boldsymbol{\lambda}_{\tau}^{+}) = \mathbf{w}_{\mathrm{i}}^{+} \end{aligned} \tag{21}$$

The expression above can be interpreted as a reduced order model of the nonsmooth multibody system, where the dynamics have been projected onto the space parametrized by the interface velocities $\mathbf{w}_{\mathrm{i}}$. Alternatively, the interface model in Eq. (21) can also be written as the impulse-momentum dynamics equations of the IM in Eq. (5)

$$\widetilde{\mathbf{M}}_{\mathrm{i}}\big(\mathbf{w}_{\mathrm{i}}^{+} - \mathbf{w}_{\mathrm{i}}\big) = h\big(\widetilde{\boldsymbol{\lambda}}_{\mathrm{i}} + \boldsymbol{\lambda}_{\mathrm{i}}^{+}\big) \tag{22}$$

where now the effective mass $\widetilde{\mathbf{M}}_{\mathrm{i}}$ and force $\widetilde{\boldsymbol{\lambda}}_{\mathrm{i}}$ are

$$\widetilde{\mathbf{M}}_{\mathrm{i}} = \Big(\mathbf{D}(\mathbf{I} - \hat{\mathbf{P}}_{\mathrm{c}})\mathbf{M}^{-1}\mathbf{D}^{\mathrm{T}}\Big)^{-1} \tag{23}$$

$$\widetilde{\boldsymbol{\lambda}}_{\mathrm{i}} = \widetilde{\mathbf{M}}_{\mathrm{i}}\Big(\mathbf{D}(\mathbf{I} - \hat{\mathbf{P}}_{\mathrm{c}})\mathbf{M}^{-1}\big(\mathbf{f}_{\mathrm{a}} - \mathbf{c} + \mathbf{D}^{\mathrm{T}}\boldsymbol{\lambda}_{\tau}^{+}\big) + \dot{\mathbf{D}}\mathbf{v} + \mathbf{D}\hat{\mathbf{P}}_{\mathrm{c}}\frac{\mathbf{v}^{+} - \mathbf{v}}{h}\Big) \tag{24}$$

where $\hat{\mathbf{P}}_{\mathrm{c}} = \mathbf{M}^{-1}\mathbf{A}_{\alpha}^{\mathrm{T}}\left(\mathbf{A}_{\alpha}\mathbf{M}^{-1}\mathbf{A}_{\alpha}^{\mathrm{T}}\right)^{-1}\mathbf{A}_{\alpha}$, and $\boldsymbol{\lambda}_{\tau}^{+}$ and $\mathbf{v}^{+}$ need to be determined by solving the MLCP in Eq. (17). The expressions of the effective mass and force terms are similar to the ones for smooth systems in Eqs. (6) and (7). However, matrix $\hat{\mathbf{P}}_{\mathrm{c}}$ is now computed using the Jacobian matrix of the active constraints $\mathbf{A}_{\alpha}$, which makes the IM account for contacts that detach or slide.

Constraint Regularization

Constraint regularization is commonly used to overcome force indeterminacy when constraints are redundant. Essentially, constraints are relaxed and bodies are allowed to overlap with each other, so that the constraint violation is used to define the constraint force. Then, the regularized version of the MLCP in Eq. (14) is [23]

$$\left.\begin{array}{c} \begin{bmatrix} \mathbf{M} & -\mathbf{A}^{\mathrm{T}} \\ \mathbf{A} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{v}^{+} \\ h\boldsymbol{\lambda}_{\mathrm{c}}^{+} \end{bmatrix} + \begin{bmatrix} h(\mathbf{c} - \mathbf{f}) - \mathbf{M}\mathbf{v} \\ h\dot{\mathbf{A}}\mathbf{v} + \boldsymbol{\gamma}\boldsymbol{\Phi}_{\mathrm{c}}h^{-1} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{w}_{\mathrm{c}}^{+} \end{bmatrix} \\ \mathbf{w}_{\mathrm{c}}^{+} \ \perp \ \boldsymbol{\lambda}_{\mathrm{c}}^{+} \in \big[\boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{lo}}, \boldsymbol{\lambda}_{\mathrm{c}}^{\mathrm{up}}\big] \end{array}\right\} \tag{25}$$

where $\mathbf{C}$ and $\boldsymbol{\gamma}$ are $m_{\mathrm{c}} \times m_{\mathrm{c}}$ matrices containing the regularization terms, which are usually diagonal and positive definite. Furthermore, the reduced form can be written as

$$\underbrace{\big(\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^{\mathrm{T}} + \mathbf{C}\big)}_{\mathbf{H}} h\boldsymbol{\lambda}_{\mathrm{c}}^{+} + \underbrace{\mathbf{A}\mathbf{M}^{-1}h\mathbf{f} + h\dot{\mathbf{A}}\mathbf{v} + \boldsymbol{\gamma}\boldsymbol{\Phi}_{\mathrm{c}}h^{-1}}_{\mathbf{b}} = \mathbf{w}_{\mathrm{c}}^{+} \tag{26}$$

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

If constraints are regularized, the expressions for the effective mass $\widetilde{\mathbf{M}}_{\mathrm{i}}$ and effective force $\widetilde{\boldsymbol{\lambda}}_{\mathrm{i}}$ in Eqs. (23) and (24), can be derived similarly as

$$\widetilde{\mathbf{M}}_{\mathrm{i}} = \Big(\mathbf{D}(\mathbf{I} - \hat{\mathbf{P}}_{\mathrm{c}})\mathbf{M}^{-1}\mathbf{D}^{\mathrm{T}}\Big)^{-1} \tag{27}$$

$$\begin{aligned} \widetilde{\boldsymbol{\lambda}}_{\mathrm{i}} = \ \widetilde{\mathbf{M}}_{\mathrm{i}}\Big(\mathbf{D}(\mathbf{I} - \hat{\mathbf{P}}_{\mathrm{c}})\mathbf{M}^{-1}\big(\mathbf{f}_{\mathrm{a}} - \mathbf{c} + \mathbf{D}^{\mathrm{T}}\boldsymbol{\lambda}_{\tau}^{+}\big) + \dot{\mathbf{D}}\mathbf{v} \\ - \mathbf{D}\mathbf{M}^{-1}\mathbf{A}_{\alpha}^{\mathrm{T}}\big(\mathbf{A}_{\alpha}\mathbf{M}^{-1}\mathbf{A}_{\alpha}^{\mathrm{T}}\big)^{-1}\big(\dot{\mathbf{A}}\mathbf{v} + \boldsymbol{\gamma}\boldsymbol{\Phi}_{\mathrm{c}}h^{-2}\big)\Big) \end{aligned} \tag{28}$$

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

Integration of the Interface Model

The time integration of the interface model is carried out using a first-order implicit Euler integrator, so that interface velocities and coordinates are updated as follows

$$\left.\begin{array}{l} \mathbf{w}_{\mathrm{i}}^{+} = \mathbf{w}_{\mathrm{i}} + h_{\mathcal{S}}\widetilde{\mathbf{M}}_{\mathrm{i}}^{-1}\big(\widetilde{\boldsymbol{\lambda}}_{\mathrm{i}} + \boldsymbol{\lambda}_{\mathrm{i}}^{+}\big) \\ \boldsymbol{\Phi}_{\mathrm{i}}^{+} = \boldsymbol{\Phi}_{\mathrm{i}} + h_{\mathcal{S}}\,\mathbf{w}_{\mathrm{i}}^{+} \end{array}\right\} \tag{29}$$

where $h_{\mathcal{S}}$ is the micro time-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 $\widetilde{\mathbf{M}}_{\mathrm{i}}$ and $\widetilde{\boldsymbol{\lambda}}_{\mathrm{i}}$ are kept constant in the macro time-step.

SYSTEMS WITH HYDRAULIC ACTUATORS

In this section we describe a co-simulation setup for nonsmooth multibody systems $\mathcal{M}$ with hydraulic components $\mathcal{H}$ using the interface model presented above. The model of a hydraulic crane with two actuators was used to that end. Nevertheless, the proposed methodology is general and can incorporate other subsystems of different nature.

TABLE 1. Hydraulic actuator parameters

Parameter	Units	Crane
a_p	m	$6.5 \cdot 10^{-3}$
l	m	1
c	Ns/m	$5 \cdot 10^{5}$
c_d	-	0.67
ρ	kg/m^3	850
p_P	MPa	50
p_T	MPa	0.1
a	Pa	$6.53 \cdot 10^{-10}$
b	-	$-1.19 \cdot 10^{-18}$

Hydraulic Actuator Model

A hydraulic system regulates 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. In this work, a hydraulic actuator is considered 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 this force can be expressed in terms of the hydraulic pressure difference as

$$f_\mathrm{h} = (p_2 - p_1)\, a_\mathrm{p} - c\dot{s}_1 \tag{30}$$

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 [4]

$$\dot{p}_1 = \frac{\beta_1}{a_\mathrm{p} l_1}\left[a_\mathrm{p}\dot{s}_1 + a_\mathrm{i} c_\mathrm{d}\sqrt{\frac{2(p_\mathrm{P}-p_1)}{\rho}}\,\delta_\mathrm{P1} - a_\mathrm{o} c_\mathrm{d}\sqrt{\frac{2(p_1-p_\mathrm{T})}{\rho}}\,\delta_\mathrm{T1}\right] \tag{31}$$

$$\dot{p}_2 = \frac{\beta_2}{a_\mathrm{p} l_2}\left[-a_\mathrm{p}\dot{s}_1 + a_\mathrm{o} c_\mathrm{d}\sqrt{\frac{2(p_\mathrm{P}-p_2)}{\rho}}\,\delta_\mathrm{P2} - a_\mathrm{i} c_\mathrm{d}\sqrt{\frac{2(p_2-p_\mathrm{T})}{\rho}}\,\delta_\mathrm{T2}\right] \tag{32}$$

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

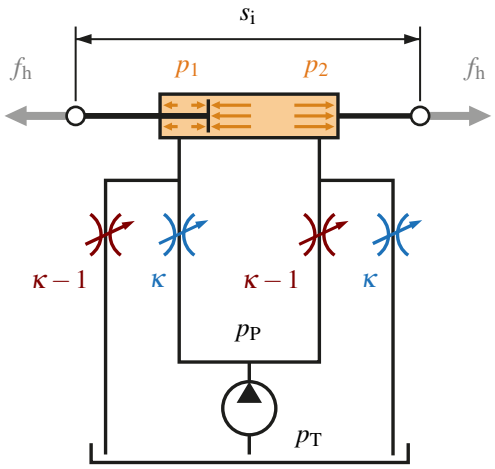

FIGURE 5. Hydraulic model of an actuator.

$$\beta_i = \frac{1 + a p_i + b p_i^2}{a + 2 b p_i}, \quad i = 1,2 \tag{33}$$

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

$$l_1 = 0.5l + s_{1,0} - s_1 \qquad \text{and} \qquad l_2 = 0.5l + s_1 - s_{1,0} \tag{34}$$

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

$$a_\mathrm{i} = 5 \cdot 10^{-4}\kappa \qquad \text{and} \qquad a_\mathrm{o} = 5 \cdot 10^{-4}(1-\kappa) \tag{35}$$

In Eq. (35), $\kappa \in [0,1]$ is the valve control parameter or spool displacement, i.e., the kinematic input that controls the motion of the piston. The hydraulic parameters are shown in Table 1.

Hydraulic Crane with Gripper

To test the proposed methodology in a 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

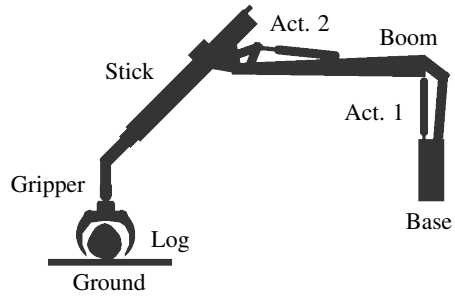

FIGURE 6. A log handling crane with two hydraulic actuators.

prismatic ones. Figure 6 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 3 non-actuated rotational degrees of freedom with respect to the stick, and two kinematically guided claws.

The inputs of the actuators are given by a velocity controller with proportional and derivative gains k^{P} and k^{D} (i.e., a PD controller). 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

$$\dot{\kappa}_j = -k_j^{\mathrm{P}}\left(w_j^* - w_j\right) - k_j^{\mathrm{D}}\left(\dot{w}_j^* - \dot{w}_j\right) \tag{36}$$

where w_j^* is the controller desired velocity, and $\dot{w}_j^*$ is its time derivative. The gains were tuned manually to damp the oscillations. Then, for actuator 1, $k_1^{\mathrm{P}} = 6.0\,\mathrm{m}^{-1}$ and $k_1^{\mathrm{D}} = 1.0\,\mathrm{s\,m}^{-1}$, and for actuator 2, $k_2^{\mathrm{P}} = 4.0\,\mathrm{m}^{-1}$ and $k_2^{\mathrm{D}} = 0.1\,\mathrm{s\,m}^{-1}$. Note that the controller has negative feedback, because, by design, an increment in the valve displacement κ would decrease the actuator force (see Fig. 5). Additionally, the controller output was limited to $\dot{\kappa} \in [-3,+3]\,\mathrm{s}^{-1}$.

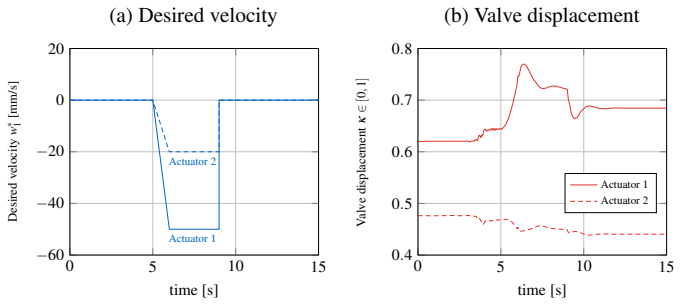

FIGURE 7. Control law of the hydraulic crane: (a) desired velocity of the controller, and (b) valve displacement as the output of the controller for the simulation using an interface model and step-size $h_{\mathcal{M}} = 16\,\mathrm{ms}$.

The desired velocity of the actuators is displayed in Fig 7-a, and it can be written for both of them as

$$w_j^* = \begin{cases} 0 & \text{if} \quad t \leqslant t_{\mathrm{ini}} \\ w_j^{\max}\dfrac{t - t_{\mathrm{ini}}}{t_{\max} - t_{\mathrm{ini}}} & \text{if } t_{\mathrm{ini}} < t \leqslant t_{\max} \\ w_j^{\max} & \text{if } t_{\max} < t \leqslant t_{\mathrm{end}} \\ 0 & \text{if} \quad t > t_{\mathrm{end}} \end{cases} \tag{37}$$

Grasping of the log occurs for $t \in [3,5]\,\mathrm{s}$. The arm remains motionless until $t_{\mathrm{ini}} = 5\,\mathrm{s}$. The maximum desired velocity $w_j^{\max}$ is applied at $t_{\max} = 6\,\mathrm{s}$ and is kept constant until $t_{\mathrm{end}} = 9\,\mathrm{s}$. The 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 (see Fig. 7).

Friction is considered in all contact interfaces with $\mu = 0.5$. Due to constraint redundancy at the contact between gripper and log, the contact constraints are regularized with a stiffness and damping coefficients of $K_{\mathrm{n}} = 10^6\,\mathrm{N/m}$ and $B_{\mathrm{n}} = 10^5\,\mathrm{Ns/m}$.

RESULTS

The multibody model of the log handler was created using the Vortex simulation software package; the co-simulation manager as well as the hydraulic model were written in C++ and embedded into the software package. The simulations were performed on an Intel Core i7-4720HQ machine with a 4-core CPU@2.60 GHz and 8 GB of RAM, and running Windows 10.

The step-size of the hydraulics was fixed to $h_{\mathcal{H}} = 0.2\,\mathrm{ms}$, and different step-sizes $h_{\mathcal{M}}$ were used for the multibody system, which matched the macro step-size H in all simulations. The results obtained using the interface model (IM) were compared to those delivered by direct co-simulation with zero-order hold (ZOH).

The displacement of actuator 1 is shown in Fig. 8 for both approaches and several values of $h_{\mathcal{M}}$. As can be seen, the simulation becomes unstable for large step-sizes when using ZOH direct co-simulation. On the other hand, it is possible to take

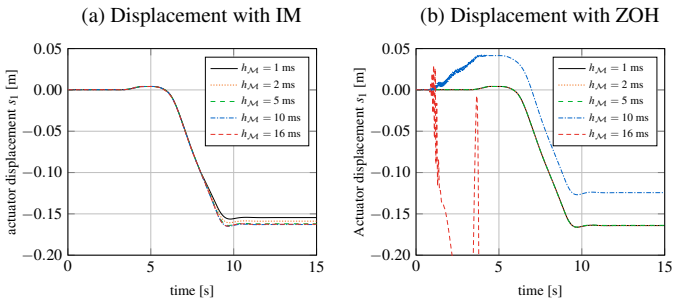

FIGURE 8. Displacement of actuator 1 in the hydraulic crane using (a) an interface model, and (b) a zero-order hold. $h_{\mathcal{H}} = 0.2$ ms.

larger step-sizes with an interface model without losing stability. Figure 9 shows the velocity and force of actuator 1 during the manoeuvre. Significant oscillations can be observed in the velocity plots, e.g., in Fig. 9-c). This takes place during the grasping phase, when contact and friction forces have the most significant effect on the system dynamics. The interface model alleviated the severity of these oscillations. It is also noteworthy that ZOH co-simulation with $h_{\mathcal{M}} = 10$ ms recovers stability after grasping the log ($t = 5$ s), in spite of the severe oscillations in force and velocity during the grasping stage. When the claws completely grasp the log, its mass becomes supported by the arm, which reduces the natural frequency of the system and makes its numerical integration more stable.

CONCLUSIONS

Multirate co-simulation approaches make it possible to predict the behaviour of complex multiphysics systems in a modular and efficient way. The nature of some applications and simulation environments, however, requires the use of non-iterative schemes that may suffer from accuracy and instability issues. In particular, when conventional Jacobi co-simulation schemes are used, this may impose limitations on the maximum macro step-size that can be used to communicate the subsystems, as well as on the integration step-sizes in the subsystems. This, in turn, is detrimental for code execution efficiency, which is a major concern in computationally demanding applications, e.g., those that require real-time performance. The causes of these stability and accuracy issues can be traced back to discontinuities and delays introduced by the discrete-time coupling at the co-simulation interface. In multirate setups, the lack of updated inputs between communication points for fast subsystems makes it necessary to formulate some assumption about their behaviour to proceed with the numerical integration until the next coupling instant.

Interface models (IM) provide a reduced representation of the dynamics of slower subsystems, to obtain a dynamics-based prediction of the evolution of their inputs between communication points. IM for mechanical systems 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 generalized force terms that describe its dynamics in the subspace defined by these interface velocities. In this paper, the expression of the effective mass and force terms for mechanical systems with nonsmooth dynamics, 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 mixed linear complementarity problem (MLCP). Moreover, the proposed IM was used in the multirate co-simulation of a mechanical system with hydraulic actuators. Results showed that the use of an IM

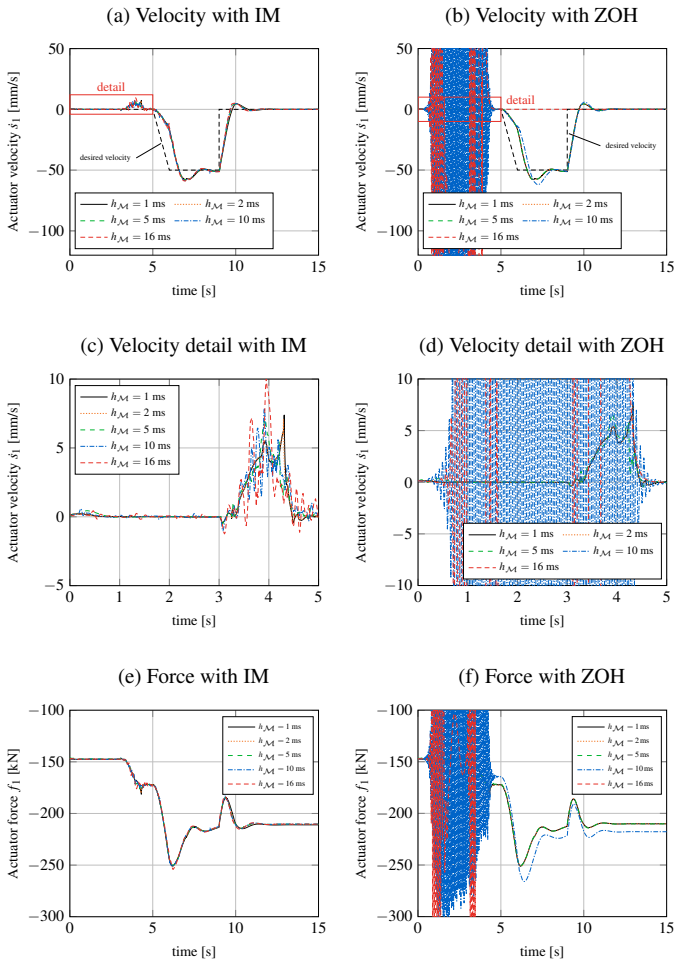

FIGURE 9. Velocity of actuator 1 in the hydraulic crane using (a) an interface model, and (b) a zero-order hold. Detail plots of the velocity during the grasping phase of the log $t \in [3,5]$ s are shown in (c) and (d). Actuator force with (e) interface model and (f) zero-order hold. Different step-sizes $h_{\mathcal{M}}$ were used for the multibody system; $h_{\mathcal{H}} = 0.2$ ms.

to predict the inputs of the fast subsystem between communication steps delivered more stable and accurate results than conventional polynomial extrapolation techniques.

ACKNOWLEDGMENT

This research was supported by the Natural Sciences and Engineering Research Council Canada (NSERC) and CMLabs Simulations, Inc. F. González was supported by the Ramón y Cajal program of the Ministry of Economy of Spain, contract no. RYC-2016-20222. The support is gratefully acknowledged.

REFERENCES

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

[2] Samin, J. C., Brüls, O., Collard, J. F., Sass, L., and Fisette, P., 2007. "Multiphysics modeling and optimization of mechatronic multibody systems". *Multibody System Dynamics,* ***18***(3), pp. 345–373.

[3] Rahikainen, J., Kiani, M., Sopanen, J., Jalali, P., and Mikkola, A., 2018. "Computationally efficient approach for simulation of multibody and hydraulic dynamics". *Mechanism and Machine Theory,* ***130***, pp. 435–446.

[4] 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". *Archive of Mechanical Engineering,* ***58***(2), pp. 223–243.

[5] Gomes, C., Thule, C., Broman, D., Larsen, P. G., and Vangheluwe, H., 2018. "Co-simulation: A survey". *ACM Computing Surveys,* ***51***(3), pp. 1–33.

[6] Benedikt, M., and Holzinger, F. R., 2016. "Automated configuration for non-iterative co-simulation". In Proceedings of the 17th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE).

[7] Kübler, R., and Schiehlen, W., 2000. "Modular simulation in multibody system dynamics". *Multibody System Dynamics,* ***4***(2), pp. 107–127.

[8] 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 System Dynamics,* ***25***(4), pp. 461–483.

[9] Arnold, M., 2009. *Simulation Techniques for Applied Dynamics*. Springer, ch. Numerical methods for simulation in applied dynamics, pp. 191–246.

[10] 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.

[11] Benedikt, M., Watzenig, D., Zehetner, J., and Hofer, A., 2013. "NEPCE - a nearly energy-preserving coupling element for weak-coupled problems and co-simulation". In International Conference on Computational Methods for Coupled Problems in Science and Engineering.

[12] Schweizer, B., and Lu, D., 2014. "Semi-implicit co-simulation approach for solver coupling". *Archive of Applied Mechanics,* ***84***(12), pp. 1739–1769.

[13] Haid, T., Stettinger, G., Watzenig, D., and Benedikt, M., 2018. "A model-based corrector approach for explicit co-simulation using subspace identification". In Proceedings of the 5th Joint International Conference on Multibody System Dynamics.

[14] 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". *Mechanism and Machine Theory,* ***131***, pp. 172–188.

[15] Peiret, A., González, F., Kövecses, J., and Teichmann, M., 2018. "Multibody system dynamics interface modelling for stable multirate co-simulation of multiphysics systems". *Mechanism and Machine Theory,* ***127***, pp. 52–72.

[16] Kövecses, J., 2008. "Dynamics of mechanical systems and the generalized free–body diagram – part I: General formulation". *Journal of Applied Mechanics,* ***75***(6, paper 061012), pp. 1 – 12.

[17] Stewart, D. E., and Trinkle, J. C., 1996. "An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and Coulomb friction". *International Journal for Numerical Methods in Engineering,* ***39***(15), pp. 2673–2691.

[18] Erleben, K., 2007. "Velocity-based shock propagation for multibody dynamics animation". *ACM Transactions on Graphics,* ***26***(2), p. article 12.

[19] Glocker, C., 2001. *Set-Valued Force Laws*. Springer, Troy, New York, USA.

[20] Stewart, D. E., 1998. "Convergence of a time-stepping scheme for rigid-body dynamics and resolution of Painlevé's problem". *Archive for Rational Mechanics and Analysis,* ***145***(3), pp. 215–260.

[21] Moreau, J., 1966. "Quadratic programming in mechanics: Dynamics of one sided constraints". *SIAM Journal on Control,* ***4***(1), pp. 153–158.

[22] Anitescu, M., and Potra, F. A., 1997. "Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems". *Nonlinear Dynamics,* ***14***(3), pp. 231–247.

[23] Lacoursière, C., 2006. A regularized time stepper for multibody systems. Tech. Rep. 04, UMINF.

[24] Júdice, J. J., 1994. "Algorithms for linear complementarity problems". *Algorithms for continuous optimization,* ***434***, pp. 435–474.