AIAA SciTech Forum
4-8 January 2016, San Diego, California, USA
57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference

10.2514/6.2016-1907

A Scalable Adjoint Method for Coupled Flexible Multibody Dynamics

Graeme J. Kennedy* and Komahan Boopathy†
Georgia Institute of Technology, School of Aerospace Engineering, Atlanta, Georgia, USA

High-performance aerospace structures are light-weight, flexible, frequently made from multiple, connected components, and are subject to dynamic loads. To design structures subject to these conditions, we propose a scalable adjoint method for coupled flexible multibody dynamic simulations. Adjoint-based derivative evaluation methods have been used for design optimization in a wide-array of multidisciplinary applications. However, relatively few authors have developed adjoint methods for flexible multibody dynamic simulations. To address this issue, we develop a modular adjoint-based derivative evaluation method for flexible multibody systems that is designed for high-performance parallel computing environments. We envision that this framework will be useful for the assessment and design of flexible multibody systems that arise in many aerospace applications.

I. Introduction

A variety of aerospace vehicles, including aircraft, rotorcraft, and spacecraft, can be accurately described as flexible multibody systems. As performance goals drive lower structural weight, these aerospace systems will become more flexible, in turn leading to stronger coupling between structural and dynamic response. In the MDO community, adjoint-based derivative evaluation methods have been used to perform design optimization in a wide-array of applications [15]. However, relatively few authors have developed adjoint methods for coupled multibody dynamic simulations. In this work, we develop a modular adjoint method for the evaluation of gradients of functions of interest with respect to structural and geometric variables within the context of flexible multibody simulations. Furthermore, to be compatible with high-performance parallel computing environments, we implement methods that exhibit good parallel scalability. This framework is applicable to the design of flexible multibody systems that arise in many aerospace applications.

A number of authors have studied flexible multibody systems in the context of flexible aircraft simulations. In these simulations, coupling aerodynamic forces and moments play an integral role. Several authors have developed aeroelastic analysis tools that integrate aerodynamic and structural analysis coupled to rigid-body degrees of freedom using low or medium-fidelity methods [3, 8, 7, 18]. The simultaneous consideration of these three disciplines is essential when the rigid-body modes and lowest structural modes interact, producing a fully coupled response. In order to enhance these fully coupled aeroelastic models, several authors have employed geometrically nonlinear beam analyses that account for following forces and wing-area effects that cannot be predicted by linear theories [18, 7]. These nonlinear effects are especially pronounced for large-deflections encountered during gust or maneuver conditions. Both geometrically exact formulation [18] and co-rotational [1, 8] beam element formulations have been successfully applied to these problems. These methods are especially effective for aircraft with slender, high-aspect ratio wings, such as high-altitude long-endurance (HALE) aircraft. Most of the work performed with these tools has focused on developing novel control schemes for existing aircraft [19, 7]. Fewer studies have addressed incorporating aeroelastic design constraints within a broad aircraft design problem [6]. In addition, studies which do consider the aircraft design problem employ computationally expensive gradient-free methods [8].

In this work, we concentrate on integrating structural and multibody dynamics within a unified framework. This framework is implemented using the Toolkit for the Analysis of Composite Structure (TACS) [13], a parallel finite-element code with adjoint capabilities. The end goal of this work is to integrate structures, multibody dynamics, and aerodynamics into a unified framework, however, this is beyond the scope of this paper. Instead, we focus on the intermediate but worthwhile goal of a coupled flexible multibody dynamics adjoint implementation.

The remainder of this paper is outlined as follows: in Section II, we describe the time-dependent simulation and adjoint evaluation method that will be the basis of this work. In Section III, we describe the equations of motion for a single isolated flexible body, and then describe how holonomic constraints can be added to create a flexible multibody system. In Section IV, we outline the details of the Newton–Krylov method used to solve the time-dependent governing

*Assistant Professor, AIAA Member
†Graduate Student, AIAA Student Member

equations. Finally, in Section V, we present results that verify the accuracy of the proposed adjoint implementation on three multibody systems: a triple pendulum, a trebuchet, and a flexible double-plate pendulum.

II. Flexible multibody dynamic analysis and adjoint implementation

In this section, we present the time-integration scheme used to solve the governing equations and the associated adjoint method that is used to evaluate the discrete-consistent derivative of functions of interest. These results are independent of the underlying formulation of the governing equations, which will be described in detail in Section III.

A. Implicit time integration of the descriptor system

The equations of motion of the flexible multibody system are formulated in a descriptor system that takes the following form:

$$\mathbf{R}(\mathbf{x}, \ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}, t) = 0, \tag{1}$$

where $\mathbf{x}$ are the design variables, t is time, and $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ are the rigid-body degrees of freedom, internal elastic degrees of freedom, and constraint forces for all elements within the multibody system. We have elected to express the equations of motion in this descriptor form due to the presence of holonomic constraint equations that make it difficult to write the governing equations in the form $\dot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, t)$. Furthermore, this form simplifies the derivation of the adjoint method for gradient evaluation.

We integrate the governing equations (1) in time using a backwards-difference formula over the interval $t = [0, T]$, using $N+1$ time instances with a constant time step $\Delta t = T/N$. The integration scheme uses the initial conditions $\mathbf{q}_0$, and $\dot{\mathbf{q}}_0$, which must be specified at $t = 0$. In this integration scheme, the time derivatives at time $t_k = k\Delta t$, are approximated using m-th order accurate backwards-difference formula:

$$\begin{aligned} \dot{\mathbf{q}}_k &= \frac{1}{\Delta t} \sum_{i=0}^{m} \alpha_i \mathbf{q}_{k-i} + O(\Delta t^m), \\ \ddot{\mathbf{q}}_k &= \frac{1}{\Delta t^2} \sum_{i=0}^{2m} \beta_i \mathbf{q}_{k-i} + O(\Delta t^m). \end{aligned} \tag{2}$$

Note that the first-derivative approximation requires state variable values at $m+1$ points, while the second derivative approximation requires state variables at $2m+1$ points.

Inserting the finite-difference formula (2) into the equations of motion (1), yields the following implicit nonlinear equation for the k-th time step:

$$\mathbf{R}(\mathbf{x}, \ddot{\mathbf{q}}_k, \dot{\mathbf{q}}_k, \mathbf{q}_k, t_k) = 0. \tag{3}$$

We solve this nonlinear equation using an inexact Newton method [2, 5, 4] for a sequence of state variable $\{\mathbf{q}_k^n\}_{n=1}$, until the residuals of each discipline are each reduced below a specified tolerance. The linear system of equations for the n-th update $\Delta\mathbf{q}_k^n = \mathbf{q}_k^{n+1} - \mathbf{q}_k^n$, is

$$\frac{\partial \mathbf{R}}{\partial \mathbf{q}_k} \Delta\mathbf{q}_k^n = \left[\frac{\partial \mathbf{R}}{\partial \mathbf{q}} + \frac{\alpha_0}{\Delta t} \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{q}}} + \frac{\beta_0}{\Delta t^2} \frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{q}}} \right] \Delta\mathbf{q}_k^n = -\mathbf{R}(\mathbf{x}, \ddot{\mathbf{q}}_k^n, \dot{\mathbf{q}}_k^n, \mathbf{q}_k^n, t_k), \tag{4}$$

where we solve the resulting linear system inexactly using a Krylov-subspace method. We outline the details of the implementation of the inexact Newton–Krylov solution method below. Once the solution to the inexact Newton update, $\Delta\mathbf{q}_k$, is obtained, we apply the following updates to the approximate time-derivatives:

$$\begin{aligned} \mathbf{q}_k^{n+1} &= \mathbf{q}_k^n + \Delta\mathbf{q}_k^n, \\ \dot{\mathbf{q}}_k^{n+1} &= \dot{\mathbf{q}}_k^n + \frac{\alpha_0}{\Delta t} \Delta\mathbf{q}_k^n, \\ \ddot{\mathbf{q}}_k^{n+1} &= \ddot{\mathbf{q}}_k^n + \frac{\beta_0}{\Delta t^2} \Delta\mathbf{q}_k^n. \end{aligned}$$

The implicit BDF family of time-integration schemes are not self-starting. As a result, at time step $j < m$ there are not enough points to apply the full m-th order BDF formula (2). Instead, we use a start-up strategy where j-th order BDF formula are applied for the first $j = 1, \ldots, m$ points at where the full m-th order accurate BDF integration scheme can be applied. Note that at the first $2m$ points, the second-order approximate formula must be modified based on both the initial starting condition $\dot{\mathbf{q}}_0$ and state variable values at the previous time steps.

B. Adjoint derivation

The result of the time-dependent simulation is a time-history consisting of the approximate state variable values at all time instances, $\{\mathbf{q}_k\}_{k=1}^{k=N}$. As part of an optimization problem, or when evaluating a performance metric of interest, it may be required to evaluate a function of the time history. We write these functions in the generic form:

$$f(\mathbf{x}, \mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_N).$$

Note that this is a function only of the state variables at each time-instance. Any time derivatives that are required for the evaluation of f can be evaluated using finite-difference formula.

We have implemented an adjoint method for the efficient evaluation of the gradients of functions of interest. The cost of the adjoint method scales weakly with the number of design variables and is therefore well-suited to optimization problems with few functions of interest and many design variables. Our goal is to implement an approach which can be applied to aeroelastic design optimization with computationally expensive high-fidelity analysis methods.

We derive the adjoint equations using a Lagrange multiplier method. In this approach, the Lagrangian, $\mathscr{L}$, is defined as follows:

$$\mathscr{L} = f(\mathbf{x}, \mathbf{q}_0, \ldots, \mathbf{q}_N) + \sum_{k=0}^{N} \lambda_k^T \mathbf{R}_k(\ddot{\mathbf{q}}_k, \dot{\mathbf{q}}_k, \mathbf{q}_k), \tag{5}$$

where f is the function of interest, λ_k are the Lagrange multipliers at each time step k and $\mathbf{R}_k$ are the residuals evaluated at the k-th time step.

To determine the values of the Lagrange multipliers, λ_k, we evaluate the partial derivative of Lagrangian with respect to the state variables at the k-th time step:

$$\nabla_{\mathbf{q}_k} \mathscr{L} = \frac{\partial f}{\partial \mathbf{q}_k} + \lambda_k^T \frac{\partial \mathbf{R}_k}{\partial \mathbf{q}} + \sum_{i=0}^{m} \lambda_{k+i}^T \frac{\alpha_i}{\Delta t} \frac{\partial \mathbf{R}_{k+i}}{\partial \dot{\mathbf{q}}} + \sum_{i=0}^{2m} \lambda_{k+i}^T \frac{\beta_i}{\Delta t} \frac{\partial \mathbf{R}_{k+i}}{\partial \ddot{\mathbf{q}}}.$$

Rearranging terms and taking the transpose of the above expression results in the following equation for the Langrange multipliers at the k-th time step:

$$\left[\frac{\partial \mathbf{R}_k}{\partial \mathbf{q}} + \frac{\alpha_0}{\Delta t} \frac{\partial \mathbf{R}_k}{\partial \dot{\mathbf{q}}} + \frac{\beta_0}{\Delta t} \frac{\partial \mathbf{R}_k}{\partial \ddot{\mathbf{q}}} \right]^T \lambda_k = -\frac{\partial f}{\partial \mathbf{q}_k}^T - \sum_{i=1}^{m} \frac{\alpha_i}{\Delta t} \frac{\partial \mathbf{R}_{k+i}}{\partial \dot{\mathbf{q}}}^T \lambda_{k+i} - \sum_{i=1}^{2m} \frac{\beta_i}{\Delta t} \frac{\partial \mathbf{R}_{k+i}}{\partial \ddot{\mathbf{q}}}^T \lambda_{k+i} \tag{6}$$

In this expression, the right-hand-side for the Langrange multipliers at the current time step depends on future values of the Lagrange multipliers. As a result, the Lagrange multipliers must be obtained by marching backwards in time starting from the last time-instance at $k = N$. The Jacobian in the adjoint equation (6) is the same Jacobian required for the Newton update (4). In this case, it is essential that an accurate solution is obtained since adjoint errors accumulate and produce inaccurate gradients. In practice, we solve (6) using coupled Krylov method described in Section IV.

Once the values of the Lagrange multipliers have been obtained, the total derivative of the function of interest can be evaluated as follows:

$$\nabla_{\mathbf{x}} f \equiv \nabla_{\mathbf{x}} \mathscr{L} = \frac{\partial f}{\partial \mathbf{x}} + \sum_{k=0}^{N} \lambda_k^T \frac{\partial \mathbf{R}_k}{\partial \mathbf{x}}. \tag{7}$$

Note that in practice it is necessary to store the entire time-history of the simulation for the forward solve and only $2m$ right-hand sides for the backwards adjoint solve. Within the present framework, we compute the matrix-vector product contributions required by all right-hand-sides immediately following the calculation of the Lagrange multipliers at time step k. Following this update step, we also compute the contribution to the total derivative (7) from time-step k. Therefore, within this framework, we only store the Lagrange multipliers for the current time step, but accumulate the right-hand-sides for the next $2m+1$ time steps.

III. The equations of motion

In this section, we derive the equations of motion for a flexible multibody system, first examining a single isolated flexible body, then applying kinematic constraints, and forces and torques required for compatibility of the flexible multibody system. These equations of motion can then be formulated within the framework presented in Section II.

A. Dynamics of an isolated elastic body

The state vector $\mathbf{q}$ is subdivided into the rigid-body degrees of freedom, $\mathbf{q}_R$, and the elastic state variables, $\mathbf{q}_S$. We will employ both beam- and shell-element-based structural models. Therefore, we make no assumptions about the nature of the degrees of freedom present in $\mathbf{q}_S$.

In this work, we derive the equations of motion using Lagrange's equations for quasi-coordinates [17]. In this derivation, we require an expression for the kinetic energy of the elastic body and the potential energy within the system. Figure 1 shows the initial and deformed configurations of the elastic body as well as the inertial reference frame, $\vec{\mathbf{F}}_I$, and the body-fixed reference frame, $\vec{\mathbf{F}}_B$. Note that the origin of the body-fixed frame is not necessarily at the center of mass and is not required to lie within the body itself.

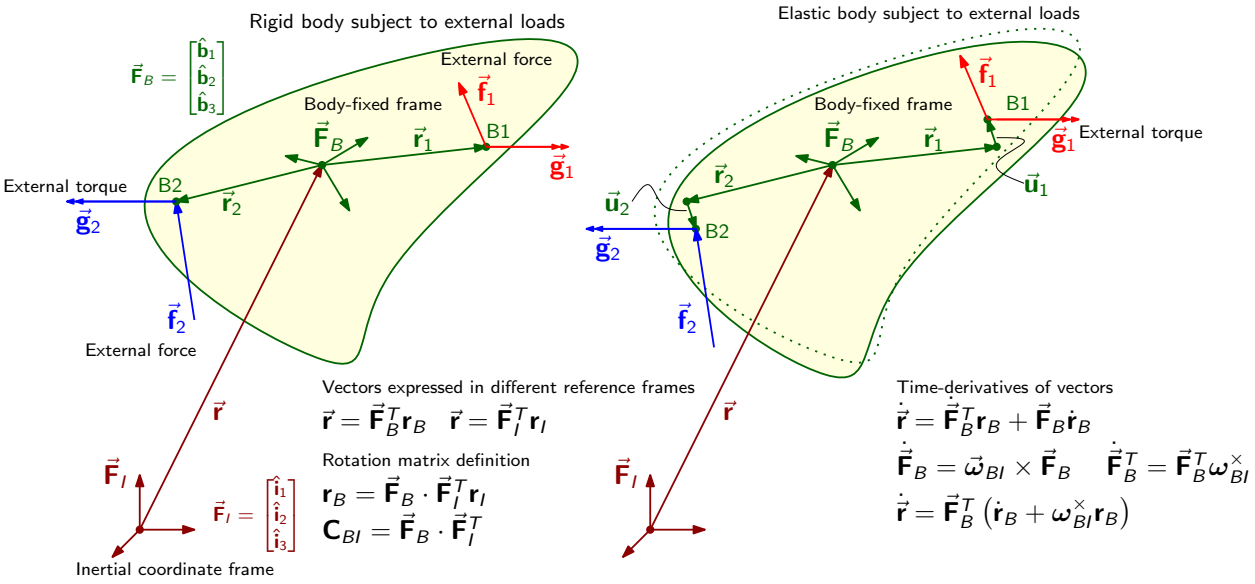


Figure 1: Illustration of the elastic body deformed under load in the body-fixed reference frame.

We express all vectors shown in Figure 1 within the body-fixed reference frame, except for the position vector, $\vec{\mathbf{r}}_0$ which is expressed in the inertial coordinate axis. The origin of the body-fixed frame has velocity $\mathbf{v}_0$ and angular velocity $\boldsymbol{\omega}$. The initial, undeformed position of a point within the elastic body is given by $\mathbf{r}^e$ and the displacement of a point in the body under load is given by $\mathbf{u}^e$. As a result, the velocity of an arbitrary point within the body can be written as follows:

$$\mathbf{v} = \mathbf{v}_0 + \dot{\mathbf{u}}^e + \boldsymbol{\omega}^\times (\mathbf{r}^e + \mathbf{u}^e) \tag{8}$$

where we have employed matrix notation for the cross product operation [9] such that:

$$\mathbf{a} \times \mathbf{b} = \mathbf{a}^\times \mathbf{b},$$

where $\mathbf{a}^\times$ is a 3×3 skew-symmetric matrix with components

$$\mathbf{a}^\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}.$$

Based on Equation (8), the kinetic energy can now be expressed as a function of the rigid-body velocity and the point-wise elastic deformation as follows:

$$\begin{aligned} T &= \frac{1}{2}\int_\Omega \mathbf{v}^T \mathbf{v} \mathrm{d}m, \\ &= \frac{1}{2}\int_\Omega \left(\mathbf{v}_0 + \dot{\mathbf{u}}^e - (\mathbf{r}^e + \mathbf{u}^e)^\times \boldsymbol{\omega}\right)^T \left(\mathbf{v}_0 + \dot{\mathbf{u}}^e - (\mathbf{r}^e + \mathbf{u}^e)^\times \boldsymbol{\omega}\right) \mathrm{d}m, \\ &= \frac{1}{2}\begin{bmatrix} \mathbf{v}_0^T & \boldsymbol{\omega}^T & \dot{\mathbf{q}}_S^T \end{bmatrix} \begin{bmatrix} m\mathbf{1} & -\hat{\mathbf{c}}^\times & \mathbf{p} \\ \hat{\mathbf{c}}^\times & \hat{\mathbf{J}} & \hat{\mathbf{h}} \\ \mathbf{p}^T & \hat{\mathbf{h}}^T & \mathbf{M} \end{bmatrix} \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega} \\ \dot{\mathbf{q}}_S \end{bmatrix}, \\ &= \frac{1}{2}\left(m\mathbf{v}_0^T\mathbf{v}_0 + \boldsymbol{\omega}^T\hat{\mathbf{J}}\boldsymbol{\omega} + \dot{\mathbf{q}}_S^T\mathbf{M}\dot{\mathbf{q}}_S\right) - \mathbf{v}_0^T\hat{\mathbf{c}}^\times\boldsymbol{\omega} + \mathbf{v}_0^T\mathbf{p}\dot{\mathbf{u}} + \boldsymbol{\omega}^T\hat{\mathbf{h}}\dot{\mathbf{q}}_S. \end{aligned} \tag{9}$$

Where we have used the following expression for the finite-element shape functions $\mathbf{u}^e = \mathbf{N}\mathbf{q}_S$. The matrices $\hat{\mathbf{c}}$, $\hat{\mathbf{J}}$, $\mathbf{p}$ and $\hat{\mathbf{h}}$ are defined as follows:

$$\hat{\mathbf{c}} = \int_\Omega (\mathbf{r}^e + \mathbf{u}^e) \mathrm{d}m \qquad \mathbf{p} = \int_\Omega \mathbf{N} \mathrm{d}m$$
$$\hat{\mathbf{J}} = -\int_\Omega (\mathbf{r}^e + \mathbf{u}^e)^\times (\mathbf{r}^e + \mathbf{u}^e)^\times \mathrm{d}m \quad \hat{\mathbf{h}} = \int_\Omega (\mathbf{r}^e + \mathbf{u}^e)^\times \mathbf{N} \mathrm{d}m$$

where the symbol $\hat{}$ denotes quantities that depend on the state of elastic deformation.

In the following work, we simplify the equations of motion by making the assumption that quantities within the kinetic energy expression of the form $(\mathbf{r}^e)^\times \mathbf{u}^e$ can be neglected. For relatively rigid structures, this is a reasonable assumption, however, these terms cannot be neglected when highly-flexible structures are analyzed. With this assumption, the kinetic energy (9) can now be expressed as follows:

$$T = \frac{1}{2}\left(m\mathbf{v}_0^T\mathbf{v}_0 + \boldsymbol{\omega}^T\mathbf{J}\boldsymbol{\omega} + \dot{\mathbf{q}}_S^T\mathbf{M}\dot{\mathbf{q}}_S\right) - \mathbf{v}_0^T\mathbf{c}^\times\boldsymbol{\omega} + \mathbf{v}_0^T\mathbf{p}\dot{\mathbf{u}} + \boldsymbol{\omega}^T\mathbf{h}\dot{\mathbf{q}}_S, \tag{10}$$

where the matrices, $\mathbf{c}$, $\mathbf{J}$ and $\mathbf{h}$ are defined as follows:

$$\mathbf{c} = \int_\Omega \mathbf{r}^e \mathrm{d}m, \qquad \mathbf{J} = -\int_\Omega \mathbf{r}^{e\times}\mathbf{r}^{e\times}\mathrm{d}m, \qquad \mathbf{h} = \int_\Omega \mathbf{r}^{e\times}\mathbf{N}\mathrm{d}m.$$

The potential energy of the elastic body is to the internal strain energy of the elastic deformation and can be written as follows:

$$U = \mathbf{q}_S^T\mathbf{K}\mathbf{q}_S, \tag{11}$$

where $\mathbf{K}$ is the stiffness matrix. Note that we have assumed a linear structure, however, geometric nonlinearities due to large deformations could also be taken into account within this framework without significant modification.

The equations of motion for an elastic body can be derived using Lagrange's equations expressed using quasi-coordinates [17] as follows:

$$\begin{aligned}
\frac{d}{dt}\left(\frac{\partial L}{\partial \mathbf{v}_0}^T\right) + \boldsymbol{\omega}^\times \frac{\partial L}{\partial \mathbf{v}_0}^T &= \mathbf{f}_R, \\
\frac{d}{dt}\left(\frac{\partial L}{\partial \boldsymbol{\omega}}^T\right) + \mathbf{v}_0^\times \frac{\partial L}{\partial \mathbf{v}_0}^T + \boldsymbol{\omega}^\times \frac{\partial L}{\partial \boldsymbol{\omega}}^T &= \mathbf{g}_R, \\
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}_S}^T\right) - \frac{\partial L}{\partial \mathbf{q}_S}^T &= \mathbf{f},
\end{aligned} \tag{12}$$

where L is the Lagrangian, $L = T - U$, and $\mathbf{f}_R$, and $\mathbf{g}_R$ are the total external forces and torques acting on the body expressed in the body-fixed frame. Finally, $\mathbf{f}$ is the consistent force vector acting on the elastic degrees of freedom. These terms will be described in greater detail below.

The equations of motion can now be derived using the expressions for the kinetic (10), and potential energies (11), respectively, in combination with the equations of motion in terms of quasi-coordinates (12). This results in the following two equations for the rigid-body motion:

$$\begin{aligned}
m\dot{\mathbf{v}}_0 - \mathbf{c}^\times\dot{\boldsymbol{\omega}} + \mathbf{p}\ddot{\mathbf{q}}_S + \boldsymbol{\omega}^\times(m\mathbf{v}_0 - \mathbf{c}^\times\boldsymbol{\omega} + \mathbf{p}\dot{\mathbf{q}}_S) &= \mathbf{f}_R, \\
\mathbf{c}^\times\dot{\mathbf{v}}_0 + \mathbf{J}\dot{\boldsymbol{\omega}} + \mathbf{h}\ddot{\mathbf{q}}_S + \mathbf{c}^\times\boldsymbol{\omega}^\times\mathbf{v}_0 + \boldsymbol{\omega}^\times\mathbf{J}\boldsymbol{\omega} + \mathbf{v}_0^\times\mathbf{p}\dot{\mathbf{q}}_S + \boldsymbol{\omega}^\times\mathbf{h}\dot{\mathbf{q}}_S &= \mathbf{g}_R,
\end{aligned} \tag{13}$$

while the elastic degrees of freedom are governed by the following equation of motion:

$$\mathbf{p}^T\dot{\mathbf{v}}_0 + \mathbf{h}^T\dot{\boldsymbol{\omega}} + \mathbf{M}\ddot{\mathbf{q}}_S + \mathbf{K}\mathbf{q}_S = \mathbf{f}. \tag{14}$$

In addition to the governing equations derived above, it is necessary to integrate the following kinematic equations to obtain the trajectory of the body:

$$\begin{aligned}
\dot{\mathbf{r}}_0 &= \mathbf{C}_{bi}^T\mathbf{v}_0, \\
\dot{\boldsymbol{\theta}} &= \mathbf{S}^{-1}\boldsymbol{\omega}.
\end{aligned} \tag{15}$$

The governing equations for the single elastic system are the concatenation of the system dynamics (13) and the kinematic relationships (15). These equations take the form

$$\mathbf{R}_R(\mathbf{x}, \ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}, t) = 0, \tag{16}$$

which is in the descriptor-system form required by the analysis in Section II.

B. Dynamics of a multibody system

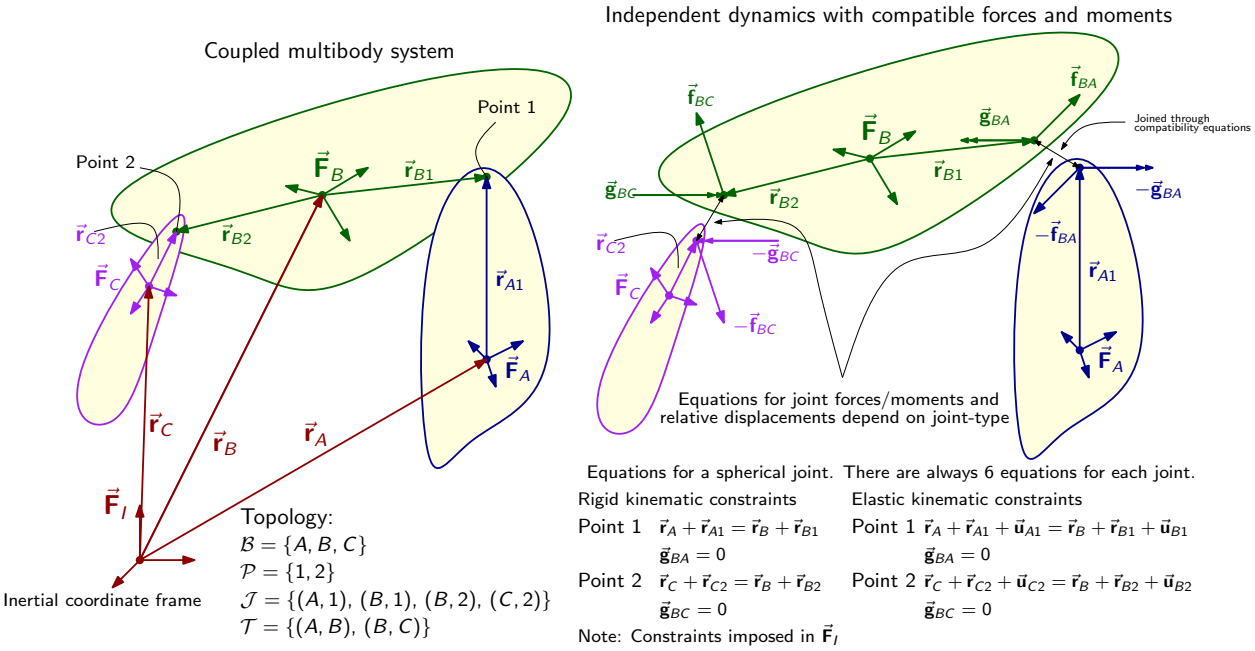


Figure 2: Illustration of the method used to couple the multibody dynamic simulation.

In this work, we obtain the governing equations for a flexible multibody system by splitting the multibody assembly into isolated components, and restoring compatibility through kinematic constraints imposed through reaction force and torque pairs. This method is illustrated in Figure 2 for a 3-body problem. In this approach, the degrees of freedom consist of both the kinematic unknowns for each body, as well as the inter-body reaction force and torque pairs. The equations for these additional force and torque degrees of freedom are obtained by imposing kinematic constraints on the relative displacement and rotation between adjacent bodies. These conditions, however, do not produce enough equations, and therefore we also impose restrictions on the reaction forces and torques. The kinematic and dynamic constraints always produce 6 equations per kinematic constraint.

Using this approach, the full set of governing equations for the multibody system can be formed by concatenating the equations of motion for each elastic body (13), and (14), together with the kinematic and dynamic equations for each kinematic constraints. We describe the details of this approach in the following section.

1. *Notation and references frames*

Before presenting the equations of motion, we first introduce the notation that we use to describe the topology of the multibody system, and the coordinate frames that are used to express the vector components.

Within the multibody system shown in Figure 2, each body is assigned an upper-case Latin letter, forming the set of bodies $\mathscr{B} = \{A, B, C \ldots\}$. Points at which kinematic constraints are created between bodies are enumerated, forming the set of points $\mathscr{P} = \{1, 2, 3, \ldots\}$. To distinguish between these two sets, we use lower-case Greek letters to identify body indices, $\alpha \in \mathscr{B}$, and lower-case Latin indices to identify points, $i \in \mathscr{P}$.

The manner in which the bodies within the system are joined together directly impacts the structure of the governing equations. To define the topological structure of the multibody system, we introduce the set of body-point pairs $\mathscr{J}$, such that for each member of the set $(\alpha, i) \in \mathscr{J}$, the body α contains the point i. For instance, in Figure 2, this set would be $\mathscr{J} = \{(A,1), (B,1), (B,2), (C,2)\}$. Adjacent bodies can now be defined using the set $\mathscr{J}$. The topology of the multibody system is reflected in the set of body-body pairs $(\alpha, \beta) \in \mathscr{T}$, such that $\mathscr{T} = \{\, (\alpha, \beta) \mid \alpha \neq \beta,\ (\alpha, i) \in \mathscr{J} \text{ and } (\beta, i) \in \mathscr{J} \text{ for some } i \,\}$. The set $\mathscr{T}$ is the set of pairs of bodies that share a connecting kinematic constraint.

The equations of motion must be expressed in component form in order to integrate them in time. We define an inertial reference frame, given by $\vec{\mathbf{F}}_I$, as well as body-fixed reference frames for each body in the system, $\vec{\mathbf{F}}_\alpha \forall \alpha \in \mathscr{B}$. The position of a body-fixed point within each body in the system is expressed in the inertial coordinate system as follows:

$$\vec{\mathbf{r}}_\alpha = \vec{\mathbf{F}}_I^T \mathbf{r}_\alpha.$$

In addition, the reaction forces and torques between bodies are expressed in the global frame as follows:

$$\left[\vec{\mathbf{f}}_{\alpha\beta} \quad \vec{\mathbf{g}}_{\alpha\beta}\right] = \vec{\mathbf{F}}_I^T \left[\mathbf{f}_{\alpha\beta} \quad \mathbf{g}_{\alpha\beta}\right], \qquad \forall \alpha, \beta \in \mathscr{B},$$

where $\vec{\mathbf{f}}_{\alpha\beta}$ and $\vec{\mathbf{g}}_{\alpha\beta}$ are the force and torque acting on body α as a result of a reaction from body β, All remaining vector quantities are expressed in their corresponding body-fixed reference frame as follows:

$$\left[\vec{\mathbf{v}}_\alpha \quad \vec{\omega}_\alpha \quad \vec{\mathbf{r}}_{\alpha i}\right] = \vec{\mathbf{F}}_\alpha^T \left[\mathbf{v}_\alpha \quad \omega_\alpha \quad \mathbf{r}_{\alpha i}\right], \qquad \forall \alpha \in \mathscr{B},$$

where $\vec{\mathbf{v}}_\alpha$ and $\vec{\omega}_\alpha$ are the velocity and angular velocity of the body α, and $\vec{\mathbf{r}}_{\alpha i}$ is the position vector from the fixed-point within the body to a point $i \in \mathscr{P}$. We also write the mass for each body as m_α, and express the inertial properties $\mathbf{c}_\alpha$ and $\mathbf{J}_\alpha$ in each local body-fixed reference frame. Furthermore, we use the vectors $\mathbf{q}_R^\alpha$ and $\mathbf{q}_S^\alpha$ to denote all rigid-body and elastic degrees-of-freedom, respectively, associated with the body α. The rotation matrix associated with the α-body is defined as follows:

$$\mathbf{C}_\alpha = \vec{\mathbf{F}}_\alpha \cdot \vec{\mathbf{F}}_I^T,$$

where $\mathbf{C}_\alpha$, transforms components from the inertial reference frame to the body-fixed frame.

Note that by Newton's third law, the reaction forces, $\vec{\mathbf{f}}_{\alpha\beta}$, and torques, $\vec{\mathbf{g}}_{\alpha\beta}$, satisfy the relationships, $\vec{\mathbf{f}}_{\alpha\beta} = -\vec{\mathbf{f}}_{\beta\alpha}$, and $\vec{\mathbf{g}}_{\alpha\beta} = -\vec{\mathbf{g}}_{\beta\alpha}$, respectively. Since both reaction torques and moments are expressed in the inertial reference frame, $\vec{\mathbf{F}}_I$, the reaction force vector components the relationship:

$$\mathbf{f}_{\beta\alpha} = -\mathbf{f}_{\alpha\beta}, \tag{17}$$

and likewise, the internal torque components satisfy the relationship:

$$\mathbf{g}_{\beta\alpha} = -\mathbf{g}_{\alpha\beta}. \tag{18}$$

For ease of notation, we define $\mathbf{f}_{\alpha\alpha} = 0$ and $\mathbf{g}_{\alpha\alpha} = 0$.

To reduce the number of degrees of freedom in the governing equations, we select $\mathbf{f}_{\alpha\beta}$, $\mathbf{g}_{\alpha\beta}$ for $\alpha < \beta$ as the independent unknowns, and compute the forces, $\mathbf{f}_{\alpha\beta}$, and torques, $\mathbf{g}_{\alpha\beta}$, for $\alpha > \beta$ from (17) and (18).

2. *Joint compatibility equations*

In this section, we present four types of joint that will be implemented within the multibody dynamics code. The four joint types that we consider are

1. a spherical joint that permits arbitrary rotation about a fixed point;
2. a revolute joint that permits rotation about a single axis at a fixed point.
3. a planar joint that permits arbitrary rotation and translation within a fixed plane; and
4. a prismatic joint that permits deformation along a single axis.

In each case, we formulate six equations for each joint. Some of these equations are trivial, such as zero torque, $\mathbf{g}_{\alpha\beta} = 0$, and could be eliminated from the system of equations. However, treating the joint equations in a uniform manner has two benefits:

1. the structure of the equations is not affected by complicated joint expressions and relationships, simplifying internal bookkeeping in the code; and
2. the structure in the equations enables us to use optimized block-specific direct sparse matrix factorization code.

For each joint within the multibody system, we take a body-pair $(\alpha, \beta) \in \mathscr{T}$, and choose an point $i \in \mathscr{P}$ such that $(\alpha, i) \in \mathscr{J}$ and $(\beta, i) \in \mathscr{J}$, and we form a set of equations written as follows:

$$\mathbf{j}_{\alpha\beta}(\mathbf{q}_R^\alpha, \mathbf{q}_R^\beta, \mathbf{q}_S^\alpha, \mathbf{q}_S^\beta, \mathbf{f}_{\alpha\beta}, \mathbf{g}_{\alpha\beta}) = 0, \tag{19}$$

that describe the kinematics and dynamic constraints that hold at the joint. While the details of the joint equations depend on the specific joint-type, the general linearization the constraint equations (19) takes the following form:

$$\mathbf{J}_{\alpha\beta}^\alpha \Delta\mathbf{q}_R^\alpha + \mathbf{J}_{\alpha\beta}^\beta \Delta\mathbf{q}_R^\beta + \mathbf{L}_{\alpha\beta} \begin{bmatrix} \Delta\mathbf{f}_{\alpha\beta} \\ \Delta\mathbf{g}_{\alpha\beta} \end{bmatrix} + \mathbf{M}_{\alpha\beta}^\alpha \Delta\mathbf{q}_S^\alpha + \mathbf{M}_{\alpha\beta}^\beta \Delta\mathbf{q}_S^\beta. \tag{20}$$

We now present the governing equations for each of the four types of joints listed above. Without loss of generality, we assume that the joint under consideration holds between bodies A and B. Within these equations, we use $\mathbf{e}_k \in \mathbb{R}^3$, for $k = 1,\ldots,3$ to denote the Cartesian basis. Furthermore, to simplify the presentation, we do not include the effect of elastic deformation.

Spherical joint The equations for a spherical joint impose the conditions that 1) the bodies are coincident at the joint location, and 2) the joint cannot transmit a reaction torque. These conditions can be written as follows:

$$\begin{aligned} \mathbf{r}_A + \mathbf{C}_A^T \mathbf{r}_{A1} - \mathbf{r}_B - \mathbf{C}_B^T \mathbf{r}_{B1} &= 0, \\ \mathbf{g}_{AB} &= 0. \end{aligned}$$

Revolute joint For the revolute joint, we impose conditions that 1) the 3-axis for both coordinate frames must be aligned, that 2) no reaction torque is transmitted about the 3-axis, and 3) the plane of rotation must be equal in both axes. This can be expressed in the following manner:

$$\begin{aligned} \mathbf{r}_A + \mathbf{C}_A^T \mathbf{r}_{A1} - \mathbf{r}_B - \mathbf{C}_B^T \mathbf{r}_{B1} &= 0, \\ \mathbf{e}_3^T \mathbf{g}_{AB} &= 0, \\ \mathbf{e}_3^T \mathbf{C}_A \mathbf{C}_B^T \mathbf{e}_1 &= 0, \\ \mathbf{e}_3^T \mathbf{C}_A \mathbf{C}_B^T \mathbf{e}_2 &= 0. \end{aligned}$$

Prismatic joint For the prismatic joint, the two bodies may slide along a common axis without rotation relative to one another. The constraints are that 1) the rotation matrices $\mathbf{C}_A$ and $\mathbf{C}_B$ are equivalent, 2) the displacement in directions transverse to the 3-axis is zero, and 3) the reaction force along the 3-axis is zero. These criteria can be written as follows:

$$\begin{aligned} \mathbf{e}_k^T \mathbf{C}_A \mathbf{C}_B^T \mathbf{e}_k &= 1, \quad k = 1,\ldots,3, \\ \mathbf{e}_k^T \mathbf{C}_A (\mathbf{r}_A + \mathbf{C}_A^T \mathbf{r}_{A1} - \mathbf{r}_B - \mathbf{C}_B^T \mathbf{r}_{B1}) &= 0, \quad k = 1,2, \\ \mathbf{e}_3^T \mathbf{f}_{AB} &= 0. \end{aligned}$$

Planar joint The planar joint constrains the two bodies to slide along the plane perpendicular to the 3-axis, passing through the origin of the fixed point of the body $\alpha = A$. The constraints are that 1) the relative displacement along the 3-axis of A is zero, 2) the A and B frames share the same 3-axis, 3) the in-plane reaction force in the 1- and 2-directions is zero, and 4) the reaction torque is zero about the 3-axis. The equations for the planar joint can be written as follows:

$$\begin{aligned} \mathbf{e}_3^T \mathbf{C}_A (\mathbf{r}_A + \mathbf{C}_A^T \mathbf{r}_{A1} - \mathbf{r}_B - \mathbf{C}_B^T \mathbf{r}_{B1}) &= 0, \\ \mathbf{e}_3^T \mathbf{C}_A \mathbf{C}_B^T \mathbf{e}_1 &= 0, \\ \mathbf{e}_3^T \mathbf{C}_A \mathbf{C}_B^T \mathbf{e}_2 &= 0, \\ \mathbf{e}_1^T \mathbf{f}_{AB} &= 0, \\ \mathbf{e}_2^T \mathbf{f}_{AB} &= 0, \\ \mathbf{e}_3^T \mathbf{g}_{AB} &= 0. \end{aligned}$$

Note that these equations are presented in a simplified form, where the joints are constrained to lie in the 3-axis. These constraints can be easily generalized to arbitrary points and axes within the multibody system.

3. Equations of motion

With these definitions, the equations of motion can now be expressed in the following descriptor form:

$$\mathbf{R}(\mathbf{x}, \ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}, t) = \begin{bmatrix} \begin{bmatrix} \mathbf{C}_\alpha \dot{\mathbf{r}}_\alpha - \mathbf{v}_\alpha \\ \mathbf{S}(\theta_\alpha)\dot{\theta}_\alpha - \omega_\alpha \\ m_\alpha \dot{\mathbf{v}}_\alpha - \mathbf{c}_\alpha^\times \dot{\omega}_\alpha + \mathbf{p}_\alpha \ddot{\mathbf{q}}_S^\alpha + \omega_\alpha^\times (m_\alpha \mathbf{v}_\alpha - \mathbf{c}_\alpha^\times \omega_\alpha + \mathbf{p}_\alpha \dot{\mathbf{q}}_S^\alpha) - \sum_{\beta \in \mathscr{B}} \mathbf{f}_{\alpha\beta} \\ \mathbf{c}_\alpha^\times \dot{\mathbf{v}}_\alpha + \mathbf{J}_\alpha \dot{\omega}_\alpha + \mathbf{h}_\alpha \ddot{\mathbf{q}}_S^\alpha + \mathbf{c}_\alpha^\times \omega_\alpha {}^\times \mathbf{v}_\alpha + \omega_\alpha^\times \mathbf{J}_\alpha \omega_\alpha + \mathbf{v}_\alpha^\times \mathbf{p}_\alpha \dot{\mathbf{q}}_S^\alpha + \omega_\alpha^\times \mathbf{h}_\alpha \dot{\mathbf{q}}_S^\alpha - \sum_{\beta \in \mathscr{B}} \mathbf{g}_{\alpha\beta} \end{bmatrix} & \forall \alpha \in \mathscr{B} \\ \mathbf{j}_{\alpha\beta}(\mathbf{q}_R^\alpha, \mathbf{q}_R^\beta, \mathbf{q}_S^\alpha, \mathbf{q}_S^\beta, \mathbf{f}_{\alpha\beta}, \mathbf{g}_{\alpha\beta}) & \forall (\alpha,\beta) \in \mathscr{T} \\ \mathbf{p}_\alpha^T \dot{\mathbf{v}}_\alpha + \mathbf{h}_\alpha^T \dot{\omega}_\alpha + \mathbf{M}_\alpha \ddot{\mathbf{q}}_S^\alpha + \mathbf{K}_\alpha \mathbf{q}_S^\alpha - \sum_{\beta \in \mathscr{B}} \mathbf{Q}_{\alpha\beta} \begin{bmatrix} \mathbf{f}_{\alpha\beta} \\ \mathbf{g}_{\alpha\beta} \end{bmatrix} & \forall \alpha \in \mathscr{B} \end{bmatrix} = 0. \tag{21}$$

IV. The Krylov-based solution methods

In this section, we outline the solution methods used within the analysis and adjoint-evaluation framework. In particular, we describe the inexact Newton–Krylov solution method applied at each time-integration step, and the fully-coupled Krylov solution method used to obtain the adjoint variables. In both cases, we use Krylov subspace methods to solve the linear systems that are obtained as part of the solution procedure. Implementing an efficient and effective preconditioning technique is essential when using Krylov methods. In this work, we use a Schur-complement-based preconditioner for the combined structural and rigid-body degrees of freedom.

The overall efficiency of the simulation and the adjoint method are directly related to the efficiency of the Krylov methods used to solve the coupled linear systems at each iteration. However, we wish to solve a variety of flexible multibody problems that utilize different structural models, therefore modularity is a key consideration within this framework. To adhere to these requirements, we have used an abstract vector class that defines all the vector operations necessary to implement a Krylov subspace method. This vector class has both structural and rigid-body degrees of freedom. Since the vectors for structures and dynamics are stored in a native format, no conversion is necessary for any individual vector operation. This enables a fast and clear implementation within the existing code.

A. The Jacobian of the equations of motion

At each time-step within the analysis, we solve the equations for the k-th time step (3), where the descriptor system is given by the governing equations (21). To achieve good performance, we use an inexact Newton–Krylov method [2, 5]. To obtain a good starting point at each new iteration, we first extrapolate the solution to the next time based on the values of the state variables and their time derivatives from the previous step. This extrapolation takes the form:

$$\mathbf{q}_k^0 = \mathbf{q}_{k-1} + \Delta t \dot{\mathbf{q}}_{k-1} + \frac{\Delta t^2}{2}\ddot{\mathbf{q}}_{k-1}. \tag{22}$$

At each iteration of Newton's method, we solve the linear system (4) inexactly, based on a relative reduction of the residual [4]. The Jacobian in the Newton system takes the following form:

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{q}} + a\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{q}}} + b\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{q}}}\right]\mathbf{y} = \mathbf{b}.$$

If the contributions to this equation are separated between rigid and elastic components, this equation becomes:

$$\begin{bmatrix} \mathbf{D}_R + \mathbf{S}_R & \mathbf{S}_{RS} \\ \mathbf{S}_{SR} & \mathbf{S}_S \end{bmatrix}\begin{bmatrix} \mathbf{y}_R \\ \mathbf{y}_S \end{bmatrix} = \begin{bmatrix} \mathbf{b}_R \\ \mathbf{b}_S \end{bmatrix}. \tag{23}$$

Here the letters **D** and **S**, indicate terms from the dynamics or structural degrees of freedom, respectively.

To define all the terms in this matrix, we first define the terms for each body in the multibody system. The Jacobian of the rigid-body dynamics, $\mathbf{D}_R$, is a linear combination of the derivatives of the rigid state variables $\mathbf{q}_R$ and their first time derivatives, as follows:

$$\mathbf{D}_R^\alpha = \begin{bmatrix} a\mathbf{C}_\alpha & (\mathbf{C}_\alpha \dot{\mathbf{r}}_\alpha)^\times \mathbf{S}_\alpha & -\mathbf{1} & 0 \\ 0 & (\dot{\mathbf{S}}_\alpha + (\mathbf{S}_\alpha \dot{\boldsymbol{\theta}}_\alpha)^\times \mathbf{S}_\alpha) + a\mathbf{S}_\alpha & 0 & -\mathbf{1} \\ 0 & 0 & m_\alpha(a\mathbf{1} + \boldsymbol{\omega}_\alpha{}^\times) & -a\mathbf{c}_\alpha^\times + ((\mathbf{c}_\alpha^\times \boldsymbol{\omega}_\alpha)^\times - m_\alpha \mathbf{v}_\alpha^\times - \boldsymbol{\omega}_\alpha^\times \mathbf{c}_\alpha^\times) \\ 0 & 0 & a\mathbf{c}_\alpha^\times + \mathbf{c}_\alpha^\times \boldsymbol{\omega}_\alpha^\times & a\mathbf{J}_\alpha + (\boldsymbol{\omega}_\alpha^\times \mathbf{J}_\alpha - (\mathbf{J}_\alpha \boldsymbol{\omega}_\alpha)^\times - \mathbf{c}_\alpha^\times \mathbf{v}_\alpha^\times) \end{bmatrix},$$

where $\mathbf{C}_\alpha$ and $\mathbf{S}_\alpha$ are the rotation matrix, and the angular rate matrix, respectively. Each diagonal block is coupled to adjacent blocks in the multibody system through the force-coupling matrices defined as follows:

$$\mathbf{D}_R^{\alpha\beta} = 0 \text{ if} (\alpha,\beta) \notin \mathscr{T} \qquad \mathbf{D}_R^{\alpha\beta} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -\mathbf{C}_\alpha & 0 \\ 0 & -\mathbf{C}_\alpha \end{bmatrix} \text{ if } \alpha < \beta \qquad \mathbf{D}_R^{\alpha\beta} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \mathbf{C}_\alpha & 0 \\ 0 & \mathbf{C}_\alpha \end{bmatrix} \text{ otherwise}$$

The structural matrices in Equation (23), are only calculated on the structural processors. The four matrices required from each elastic body can be written as follows:

$$\mathbf{S}_R^\alpha = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -(\mathbf{p}_\alpha \dot{\mathbf{q}}_\alpha)^\times \\ 0 & 0 & -(\mathbf{p}_\alpha \dot{\mathbf{q}}_\alpha)^\times & -(\mathbf{h}_\alpha \dot{\mathbf{q}}_\alpha)^\times \end{bmatrix}, \quad \mathbf{S}_{RS}^\alpha = \begin{bmatrix} 0 \\ 0 \\ b\mathbf{p}_\alpha + a\boldsymbol{\omega}_\alpha^\times \mathbf{p}_\alpha \\ b\mathbf{h}_\alpha + a(\mathbf{v}_\alpha^\times \mathbf{p}_\alpha + \boldsymbol{\omega}_\alpha^\times \mathbf{h}_\alpha) \end{bmatrix},$$

$$\mathbf{S}_{SR}^\alpha = \begin{bmatrix} 0 & 0 & a\mathbf{p}_\alpha^T & a\mathbf{h}_\alpha^T \end{bmatrix}, \qquad \mathbf{S}_S^\alpha = \mathbf{K}_\alpha + b\mathbf{M}_\alpha.$$

In our implementation, the matrices $\mathbf{S}_{RS}$ and $\mathbf{S}_{SR}$ are stored as a series of vectors where only the non-zero vectors are computed and stored. Note that the matrix $\mathbf{S}_S$ is a linear combination of the stiffness matrix and the mass matrix.

Finally, the full Jacobian can be assembled by including the linearization of the joint constraint equations (20). The full Jacobian of the multibody system shown in Figure 2, has the following structure:

$$\mathbf{D}_R = \begin{bmatrix} \mathbf{D}_R^A & & & \mathbf{D}_R^{AB} & \\ & \mathbf{D}_R^B & & \mathbf{D}_R^{BA} & \mathbf{D}_R^{BC} \\ & & \mathbf{D}_R^C & & \mathbf{D}_R^{CB} \\ \mathbf{J}_{AB}^A & \mathbf{J}_{AB}^B & & \mathbf{L}_{AB} & \\ & \mathbf{J}_{BC}^B & \mathbf{J}_{BC}^C & & \mathbf{L}_{BC} \end{bmatrix} \quad \mathbf{S}_{RS} = \begin{bmatrix} \mathbf{S}_{RS}^A & & \\ & \mathbf{S}_{RS}^B & \\ & & \mathbf{S}_{RS}^C \\ \mathbf{M}_{AB}^A & \mathbf{M}_{AB}^B & \\ & \mathbf{M}_{BC}^B & \mathbf{M}_{BC}^C \end{bmatrix}$$

$$\mathbf{S}_{SR} = \begin{bmatrix} \mathbf{S}_{SR}^A & & & \mathbf{Q}_{AB} & \\ & \mathbf{S}_{SR}^B & & \mathbf{Q}_{BA} & \mathbf{Q}_{BC} \\ & & \mathbf{S}_{SR}^C & & \mathbf{Q}_{CB} \end{bmatrix} \quad \mathbf{S}_S = \begin{bmatrix} \mathbf{S}_S^A & & \\ & \mathbf{S}_S^B & \\ & & \mathbf{S}_S^C \end{bmatrix}$$

Note that these matrices exhibit a regular sparsity pattern that can be used to achieve good parallel performance. In particular, the elastic degrees of freedom are coupled through a block-diagonal matrix, $\mathbf{S}_S$, where each block can be factored independently on different processor groups.

B. Preconditioning methods

The preconditioner for the inexact Newton–Krylov method is obtained from a lagged Schur-complement factorization of the matrix (23). To compute the Schur-complement, it is necessary to first factor the block matrix associated with the structural equations, $\mathbf{S}_S$. Once the matrix $\mathbf{S}_S$ is factored, the Schur complement, $\mathbf{G}$, of the structural and rigid-body degrees of freedom can be computed as follows:

$$\mathbf{G} = \mathbf{D}_R + \mathbf{S}_R - \mathbf{S}_{RS}\mathbf{S}_S^{-1}\mathbf{S}_{RS}. \tag{24}$$

After the Schur-complement, $\mathbf{G}$, has been obtained from (24), the action of the preconditioner, $\mathbf{y} = \mathbf{M}_p^{-1}\mathbf{b}$, can be computed as follows:

$$\begin{aligned} \mathbf{t}_S &= \mathbf{S}_S^{-1}\mathbf{b}_S, \\ \mathbf{y}_R &= \mathbf{G}^{-1}(\mathbf{b}_R - \mathbf{S}_{RS}\mathbf{t}_S), \\ \mathbf{y}_S &= \mathbf{S}_S^{-1}(\mathbf{b}_S - \mathbf{S}_{SR}\mathbf{b}_R), \end{aligned}$$

where $\mathbf{t}_S$ is a temporary structural vector. The preconditioner for the transpose system can also be obtained using a similar approach. For the transpose system, the action of the preconditioner, $\mathbf{y} = \mathbf{M}_p^{-T}\mathbf{x}$, can obtained as follows:

$$\begin{aligned} \mathbf{t}_S &= \mathbf{S}_S^{-T}\mathbf{x}_S, \\ \mathbf{y}_R &= \mathbf{G}^{-T}(\mathbf{x}_R - \mathbf{S}_{SR}^T\mathbf{t}_S), \\ \mathbf{y}_S &= \mathbf{S}_S^{-T}(\mathbf{x}_S - \mathbf{S}_{RS}^T\mathbf{y}_R), \end{aligned} \tag{25}$$

where the same Schur complement factorization can be used for both the forward and transpose preconditioning operations.

C. Additional adjoint terms

In addition to solving the adjoint equation (6), it is also necessary to compute the inner product of the adjoint solution vector with the derivative of the residuals with respect to the design variables. In this section, we outline the computation of these additional terms that are essential for the total derivative evaluation.

In practice, we subdivide the design variables into two groups: geometric design variables, which modify the geometry of the problem, and material design variables, which only directly affect the constitutive relationships within the structural finite-element discretization. We write this partition as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_G \\ \mathbf{x}_M \end{bmatrix} \tag{26}$$

where $\mathbf{x}_G$, and $\mathbf{x}_M$ are the geometric and material, respectively.

The geometric design variables modify the structural nodal locations $\mathbf{X}_S$. As a result, the product of the adjoint variables with the derivative of the residuals with respect to the geometric design variables has contributions both the rigid-body degrees of freedom and the elastic degrees of freedom as follows:

$$\lambda^T \frac{\partial \mathbf{R}}{\partial \mathbf{x}_G} = \lambda_R^T \frac{\partial \mathbf{R}_R}{\partial \mathbf{x}_G} + \lambda_S^T \frac{\partial \mathbf{R}_S}{\partial \mathbf{X}_S} \frac{\partial \mathbf{X}_S}{\partial \mathbf{x}_G} \tag{27}$$

The material design variables, modify both the structural design variables and the inertial constants m, $\mathbf{c}$, and $\mathbf{J}$. As a result, the derivative with respect to the material design variables can be written as follows:

$$\lambda^T \frac{\partial \mathbf{R}}{\partial \mathbf{x}_M} = \lambda_R^T \frac{\partial \mathbf{R}_R}{\partial \mathbf{x}_M} + \lambda_S^T \frac{\partial \mathbf{R}_S}{\partial \mathbf{x}_M} \tag{28}$$

The total derivative (7) must be evaluated with the inner product of the derivative of the residuals with respect to the geometric (27), material (27) and initial-condition (27) design variables to obtain an accurate gradient.

V. Results

In the following section, we describe the application of the multibody dynamics framework to a triple pendulum system, a trebuchet, and a flexible double-plate pendulum. These cases illustrate the proposed framework on simple multibody dynamical systems.

A. Triple pendulum

In this section, we describe the application of the proposed multibody dynamics framework to a triple pendulum system. The triple pendulum system has three bodies which are denoted $\mathscr{B} = \{A, B, C\}$, and three kinematic constraints at the points $\mathscr{P} = \{1, 2, 3\}$, as shown in Figure 3. The body axes are chosen such that one of the orthogonal axes are aligned with the geometrical dimensions of the body to simplify the calculation of the inertial properties. The bodies are assumed to have a uniform density. The properties of the bodies and the kinematic constraints are listed in Tables 1 and 2, respectively.

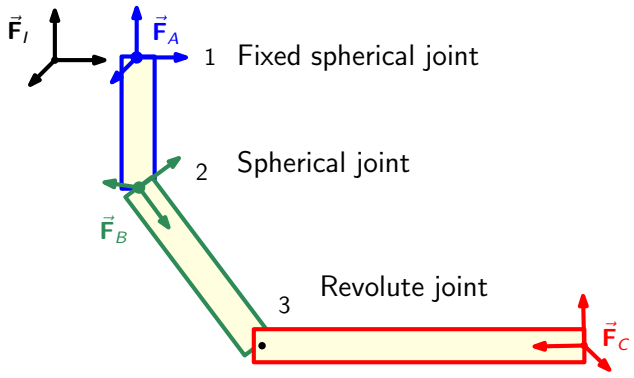


Figure 3: The schematic of the triple pendulum system.

Table 1: List of bodies in the pendulum system and their properties.

Body	Type	Mass	Length	Width	Thickness
A	Bar	1	1	0.1	0.1
B	Bar	2	2	0.1	0.1
C	Bar	3	3	0.1	0.1

Each body contributes twelve degrees of freedom to the multibody system resulting from the position vector, velocity, angular velocity and Euler angles. Each kinematic constraint contributes six degrees of freedom corresponding to the reaction forces and torques. As a result, there are 54 state variables associated with the rigid-body motion of the triple pendulum.

Table 2: List of kinematic constraints in the pendulum system and their properties.

Joint	Type	Components
1	Spherical	F_I and body A
2	Revolute (hinge)	Bodies A and B
3	Revolute (hinge)	Bodies B and C

Figure 4 shows the position and orientation of the bodies in the system over the first 3 seconds of motion. The effect of the revolute joint can be seen where the adjacent bodies in the joint are constrained to rotate about a locally-aligned axis.

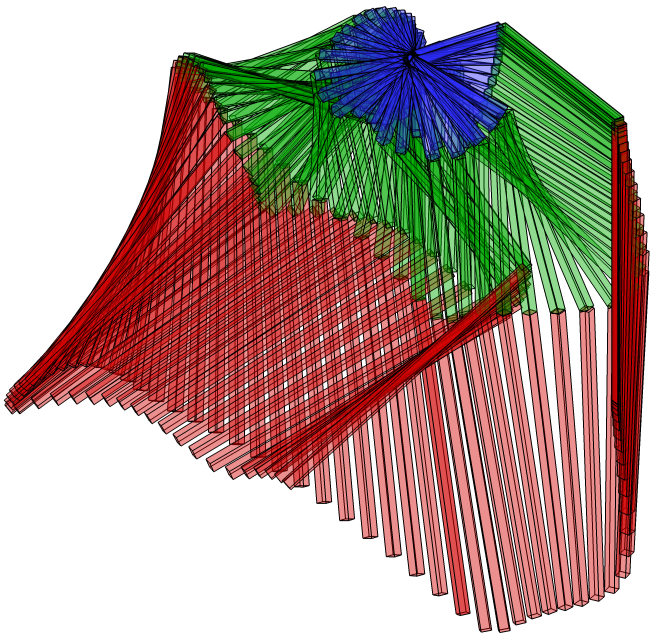


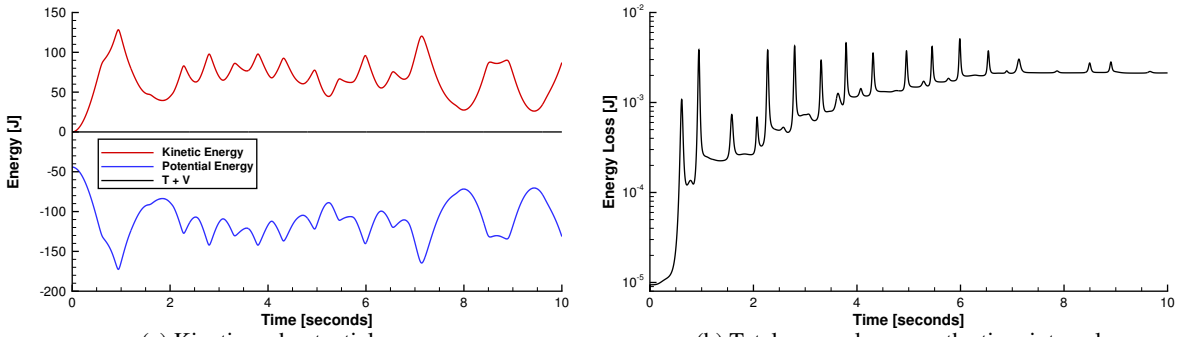
Figure 4: Motion of the triple pendulum over the first 3 seconds.

Figure 5 shows the changes in the potential and kinetic energies of the system over a 10 second time interval. Since non-conservative forces are not modeled, such as joint friction, the sum of the potential and kinetic energy should remain constant. Figure 5a clearly illustrates the complementary trend of energy transfer between kinetic and potential energies. However, the limited numerical accuracy of the time integration scheme introduces an energy defect that can grow over time. To assess this error, Figure 5 shows the energy loss over the same time period. Note that over the entire simulation, the energy loss is about 2×10^{-3} J.

Figure 6 shows a complex-step verification of the adjoint-based gradient evaluation method described above [20, 16]. The verification study compares the derivative of the Kreisselmeier–Steinhauser (KS) function of velocity [14, 12, 10] with respect to a series of design variables consisting of initial configuration variables and inertial properties. The KS function approximates the maximum velocity achieved over the time interval of the simulation. The complex-step method does not suffer from subtractive cancellation, which enables the use of very small step sizes, producing highly accurate gradient estimates. Each component of the gradient exhibits a relative accuracy on the order of 10^{-12}, illustrating near machine precision accuracy of all gradients.

B. Trebuchet

In the following section, the dynamics of a trebuchet are studied using the multibody analysis framework presented above.



(a) Kinetic and potential energy

(b) Total energy loss over the time interval

Figure 5: Plot of the potential, kinetic and total energies with time for the pendulum system.

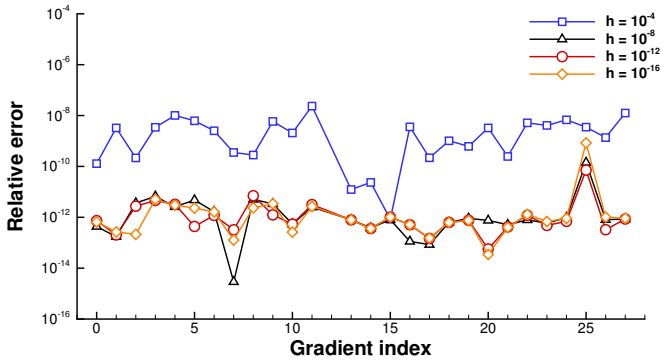


Figure 6: Gradient verification study with the complex-step method using step sizes of 10^{-4}, 10^{-8}, 10^{-12}, and 10^{-16}.

1. Geometry

The trebuchet is made up of five bodies given by $\mathscr{B} = \{A, B, C, D, E\}$, and five kinematic constraints labeled $\mathscr{P} = \{1, 2, 3, 4, 5\}$ as shown in Figure 7. The kinematic constraints at points 1, 2 and 3 are chosen to be revolute, while the other joints at points 4 and 5 chosen to be spherical. Note that the whole trebuchet assembly rotates about the axle at point 5, which lies to the left of the inertial reference frame $\vec{\mathbf{F}}_I$ and the axis of body B. The body axes are again chosen to enable convenient calculation of the inertial properties. Table 3 contains the list of bodies and their properties. The trebuchet arm, body C, is not a uniform geometry and representative values are tabulated. The geometry of the bodies and their corresponding materials are picked to facilitate the transfer of potential energy of the counterweight to kinetic energy of the projectile mass. The rigid part of the multibody dynamic system contains 90 unknowns, 60 kinematic and dynamic variables and 30 unknown internal reaction forces and torques.

Table 3: List of bodies in the trebuchet system and their properties.

Body	Name	Density	Length	Width	Thickness
A	Counter weight	25	4	4	1
B	Connecting link	10	0.5	0.5	2
C	Arm	2	20	0.5	2
D	Projectile link	10^{-2}	0.2	0.5	6
E	Projectile	10^{-2}	1	1	1

Figure 8 depicts the motion of the trebuchet system. The trebuchet arm starts from a horizontal orientation and reaches a near vertical position as it rotates about the axle. Note that the axle is not shown explicitly. The angular momentum of the swinging motion generated by the counterweight is transferred to the projectile mass through the arm and projectile link.

Figure 9 shows the kinetic and potential energy in the trebuchet system over the time history of the simulation.

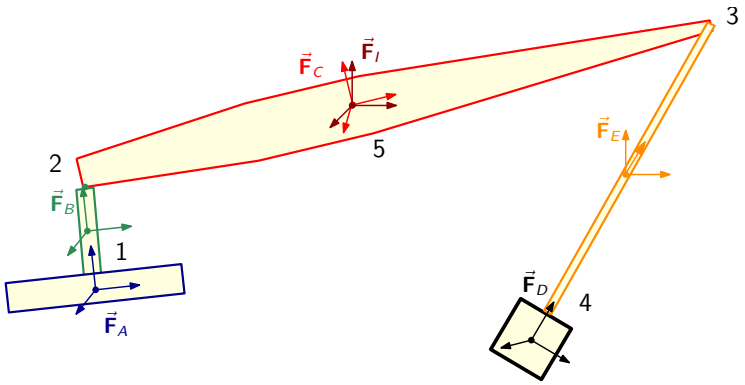


Figure 7: The schematic of the trebuchet system.

During the motion, the potential energy of the system, stored primarily in the counterweight is transferred to kinetic energy. The total energy of the system is conserved, since no non-conservative forces are modeled. The integration error produces a small change of less than 3×10^{-2} J in the total energy in the system, as shown in Figure 9. This energy loss can be reduced by utilizing a smaller time integration step size.

2. *Trebuchet optimization*

In this section, we present the results from the optimization of the trebuchet system described above. The objective of the optimization problem is to maximize projectile range, estimated using the kinematics of the projectile motion under gravity. The precise release point of the projectile is not calculated. Instead, we use the optimal release point by taking the maximum of the projectile range if it were released at any time during the entire trebuchet motion. We estimate this maximum range using the KS function, in a similar manner to the maximum velocity function described above. We also impose a constraint that the projectile must clear a barrier of specified height at a location down range along the path of the projectile.

This optimization problem is solved using ParOpt [11], an in-house interior-point optimization algorithm developed for large-scale optimization problems. The present trebuchet problem consists of six design variables and one constraint. The six design variables consist of the mass of the different components within the trebuchet system and an initial condition variable governing the release height of the counterweight.

Figure 10 show the initial and optimized trebuchet designs. Note that the counterweight release height is unconstrained at the final design point. The release height is selected such that the motion of the counterweight is synchronized with the arm and projectile motion to achieve maximum velocity at the release point.

C. Flexible double-plate pendulum

In this section, we illustrate the flexible multibody dynamics capabilities of the framework with the simulation of a flexible double-plate pendulum. This double pendulum system consists of two flexible plates which are free to rotate about a common connected edge. The double-plate system is also free to rotate about an edge of the root plate that is fixed in space. In this simulation, both plates consist of a 4×16 mesh of 3^{rd}-order MITC9 elements.

The time history of the motion of the flexible multibody system is shown in Figure 11. The contours illustrate the bending strain along the axial direction of each plate. The plates are given an initial angular velocity which propels them upwards from their initial starting configuration before gravity overcomes this initial inertia. During the motion, the joint between the two plates is evident. No moment is transmitted across this edge, producing an interesting motion.

VI. Conclusions

Aerospace structures are increasingly lightweight, are subject to dynamic loads, and are often composed of multiple connected components. To design structures subject to these conditions, we have proposed an analysis method for coupled flexible multibody dynamic simulations. Our proposed framework provides adjoint-based derivative evaluation capabilities that can be used in conjunction with gradient-based design optimization methods. This framework will be useful for the assessment and design of flexible multibody systems that arise in many aerospace applications.

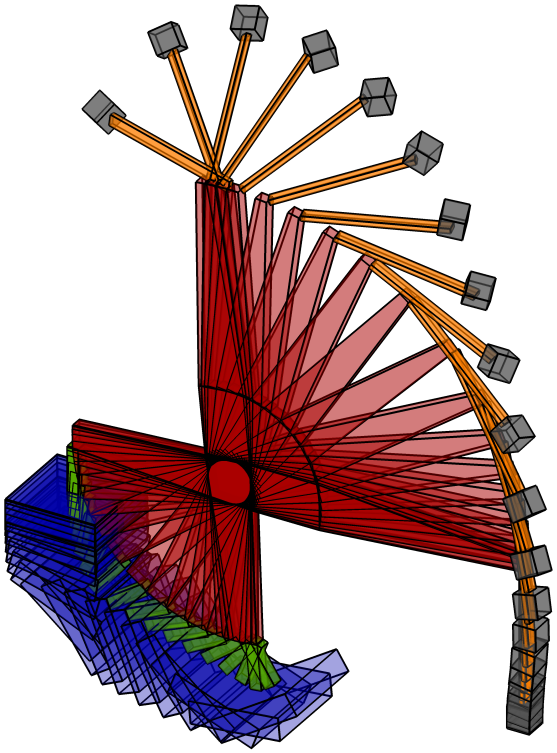


Figure 8: Motion of the trebuchet

References

[1] S. K. Chimakurthi, B. K. Stanford, C. E. S. Cesnik, and W. Shyy. Flapping wing CFD/CSD aeroelastic formulation based on a corotational shell finite element. In *50th AIAA / ASME / ASCE / AHS / ASC Structures, Structural Dynamics, and Materials Conference*, May 2009. AIAA-2009-2412.

[2] R. S. Dembo, S. C. Eisenstat, and T. Steihaug. Inexact Newton methods. *SIAM Journal on Numerical Analysis*, 19(2): 400–408, 1982. doi:10.1137/0719025.

[3] M. Drela. Integrated simulation model for preliminary aerodynamic, structural and control-law design of aircraft. In *Proceedings of the 40th AIAA Structures Dynamics and Materials Conference*, St. Louis, MO, 1999. AIAA 99-1394.

[4] S. Eisenstat and H. Walker. Choosing the forcing terms in an inexact Newton method. *SIAM Journal on Scientific Computing*, 17(1):16–32, 1996. doi:10.1137/0917003.

[5] S. C. Eisenstat and H. F. Walker. Globally convergent inexact Newton methods. *SIAM Journal on Optimization*, 4(2):393–422, 1994. doi:10.1137/0804022.

[6] K. Fidkowski, F. Engelsen, K. E. Willcox, and I. Kroo. Stochastic gust analysis techniques for aircraft conceptual design. In *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Victoria BC, Canada*, September 2008. AIAA Paper Number 2008-5848.

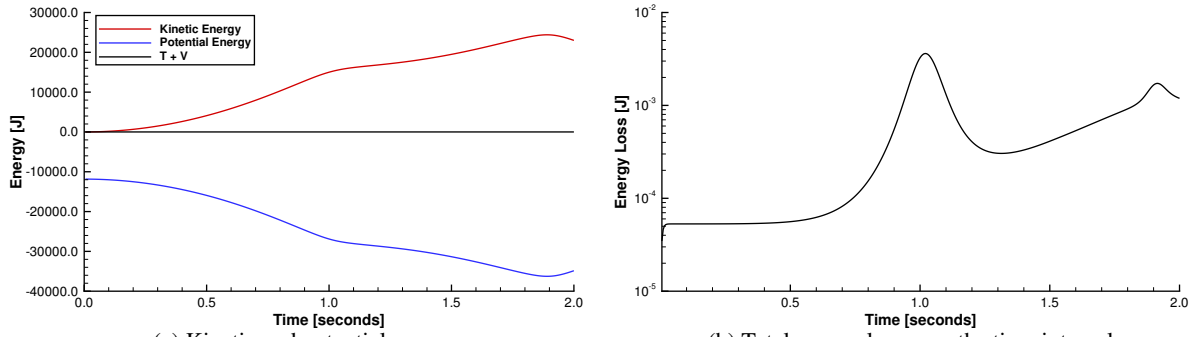
[7] S. Haghighat, H. H. T. Liu, and J. R. R. A. Martins. A model predictive gust load alleviation controller for a highly flexible aircraft. *Journal of Guidance, Control and Dynamics*, 36:1751–1766, 2012. doi:10.2514/1.57013.

[8] S. Haghighat, J. R. R. A. Martins, and H. H. T. Liu. Aeroservoelastic design optimization of a flexible wing. *Journal of Aircraft*, 49(2):432–443, 2012. doi:10.2514/1.C031344.

[9] P. C. Hughes. *Spacecraft attitude dynamics*. Dover books on engineering. Dover Publications, 2004. ISBN 9780486439259.

[10] G. J. Kennedy. Strategies for adaptive optimization with aggregation constraints using interior-point methods. *Computers & Structures*, 153(0):217 – 229, 2015. ISSN 0045-7949. doi:10.1016/j.compstruc.2015.02.024.

[11] G. J. Kennedy. Large-scale multimaterial topology optimization for additive manufacturing. In *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Kissimmee, FL, January 2015. doi:10.2514/6.2015-1799.



(a) Kinetic and potential energy

(b) Total energy loss over the time interval

Figure 9: Plot of the potential, kinetic and total energies with time for the trebuchet system.

[12] G. J. Kennedy and J. E. Hicken. Improved constraint-aggregation methods. *Computer Methods in Applied Mechanics and Engineering*, 289(0):332 – 354, 2015. ISSN 0045-7825. doi:10.1016/j.cma.2015.02.017.

[13] G. J. Kennedy and J. R. R. A. Martins. A parallel finite-element framework for large-scale gradient-based design optimization of high-performance structures. *Finite Elements in Analysis and Design*, 87(0):56 – 73, 2014. ISSN 0168-874X. doi:10.1016/j.finel.2014.04.011.

[14] G. Kreisselmeier and R. Steinhauser. Systematic control design by optimizing a vector performance index. In *International Federation of Active Controls Symposium on Computer-Aided Design of Control Systems*, Zurich, Switzerland, 1979.

[15] J. R. R. A. Martins and J. T. Hwang. Review and unification of methods for computing derivatives of multidisciplinary computational models. *AIAA Journal*, 51(11):2582–2599, November 2013. doi:10.2514/1.J052184.

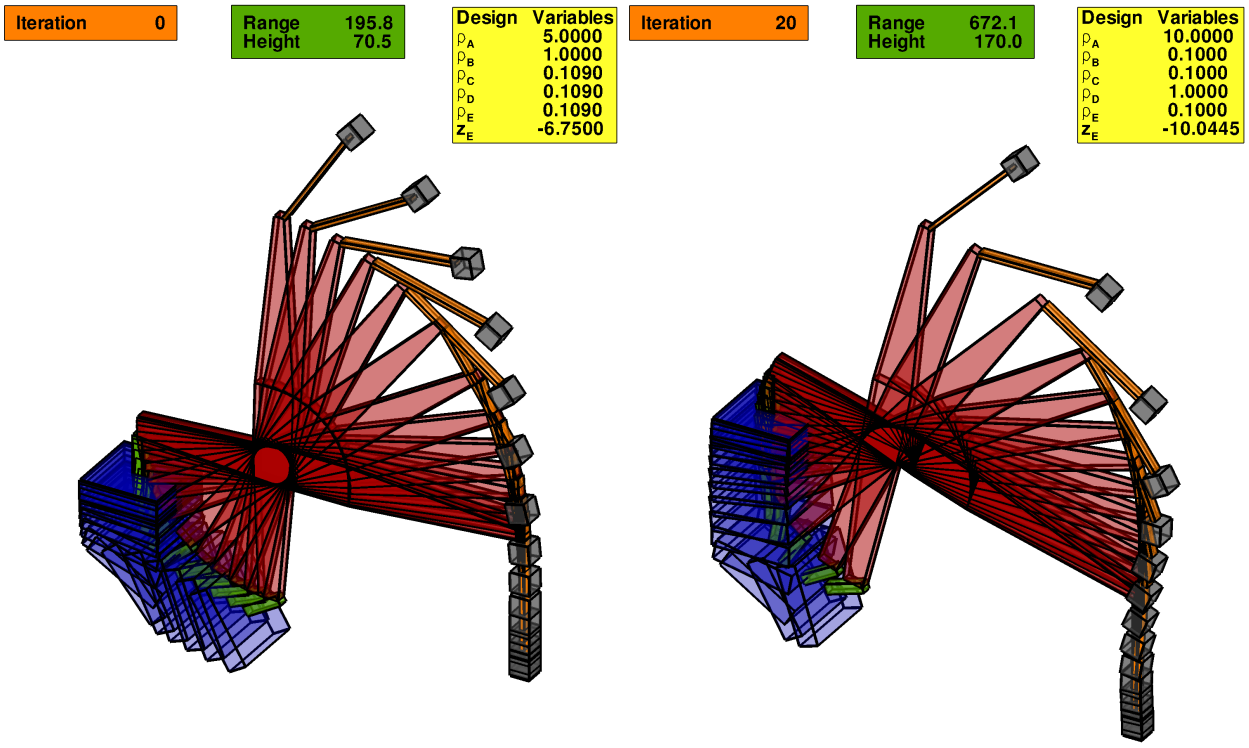
[16] J. R. R. A. Martins, P. Sturdza, and J. J. Alonso. The complex-step derivative approximation. *ACM Transactions on Mathematical Software*, 29(3):245–262, Sept. 2003. doi:10.1145/838250.838251.

[17] L. Meirovitch and T. Stemple. Hybrid equations of motion for flexible multibody systems using quasicoordinates. *Journal of Guidance, Control and Dynamics*, 18(4):678–688, 1995.

[18] C. M. Shearer and C. E. Cesnik. Nonlinear flight dynamics of very flexible aircraft. *Journal of Aircraft*, 44:1528 –1545, 2007. doi:10.2514/1.27606.

[19] C. M. Shearer and C. E. S. Cesnik. Trajectory control for very flexible aircraft. *Journal of Guidance, Control and Dynamics*, 31(2), 2008. doi:10.2514/1.29335.

[20] W. Squire and G. Trapp. Using complex variables to estimate derivatives of real functions. *SIAM Review*, 40(1):110–112, 1998. doi:10.1137/S003614459631241X.



(a) Initial design

(b) Optimized design

Figure 10: Figure illustrating the initial and final trebuchet designs.

Figure 11: Time lapse of the flexible double-plate pendulum over the time history of the motion.