Acta Astronautica 186 (2021) 549–561

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Research Paper

Forward dynamics analysis of origami-folded deployable spacecraft structures

JoAnna Fulton [*], Hanspeter Schaub

3775 Discover Drive, 429 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309-0431, United States of America

ARTICLE INFO

Keywords:
Origami space structures
Folded space structures
Free deployment
Deployment dynamics

ABSTRACT

A fast dynamics model that captures the deployment dynamics of self-actuated, origami-inspired, folded planar spacecraft structures is desired for design and verification applications. In this paper, a general simulation framework for numerically generating the equations of motion of any structure that complies with a set of pattern assumptions is presented. The framework is built through application of the articulated body forward dynamics algorithm and the tree-augmented approach for closed-chain forward dynamics. These are multi-body dynamics approaches developed in the literature for complex robotic manipulator systems. Unique adaptations are required to address the highly constrained nature of a folding structure. This solution is desirable due to the computational efficiency of the base algorithms and the ability to analyze multiple systems without reformulation of the core dynamics algorithm. Baumgarte stabilization techniques are applied to address constraint violations, but are found to be a continuing challenge for folding structures with several closed-chains.

1. Introduction

This paper develops the equations of motion of proto-typical origami folded spacecraft structures and demonstrates the numerical implementation, computational speed, and constraint stability of the approach. The dynamics model is derived using the articulated body forward dynamics (ABFD) algorithm and the augmented approach for closed-chain forward dynamics using Spatial Operator Algebra (SOA) formats. These are multi-body approaches developed in the literature for complex robotic manipulator systems [1]. Here, the applicability of this approach to folded deployable spacecraft structures is investigated. This approach is desirable due to the computational efficiency of the algorithm and the ability to implement multiple types of complex internal hinge behavior without reformulation of the dynamics algorithm. Investigations following the Lagrangian approach provide initial understanding of the problem [2], but are found to be insufficient for scaling to multiple closed chain systems due to the difficulty of implementing derivations as the number of degrees of freedom grow. The ABFD algorithm implements the mass matrix computationally without requiring specific derivation of the equations of motion, enabling changes to the system configuration without reformulation. This feature is desired for investigating systems of different size, and is not necessarily available from theoretical dynamics modeling methods alone. Additionally, the SOA format enables orderly implementation of constraint equations in the computational algorithms, regardless of the number of bodies in the system.

Origami fold patterns with repeating structure, such as the Miura [3] and Scheel patterns [4], are considered. These patterns share the common property of having no more than four panels meeting at each vertex, where a vertex refers to the shared point between panel edges as shown in Fig. 1. Therefore, the subsystem case of a four-panel set is analyzed in detail in Section 4.2 as a starting point. The scalability of the algorithm to multiple-loop systems, as would be seen in a repeating origami pattern, is a key development in this paper. A closed-loop configuration is not uncommon for robotic manipulator systems, but these typically appear as single instances in a greater open-loop chain. The unique challenges of the repeating closed-loop topology of an origami pattern has not been investigated in the literature, and therefore this development represents a novel contribution to the fields of rigid body dynamics and origami-inspired engineering. Additionally, the folded structure is deployed through self-actuation, otherwise known as a free deployment. Self-actuation is a novel deployment technique achieved through strain energy hinges, such as high strain tape springs [5]. High strain tape spring hinges are capable of storing high energy that results is fast deployments, so modeling a fast deployment is also demonstrated.

Previous research in the literature indicates active interest in this area. Several studies have modeled the kinematic and quasi-static

[*] Corresponding author.
E-mail address: joanna.fulton@colorado.edu (H. Schaub).

https://doi.org/10.1016/j.actaastro.2021.03.022
Received 28 August 2020; Received in revised form 4 March 2021; Accepted 23 March 2021
Available online 27 April 2021

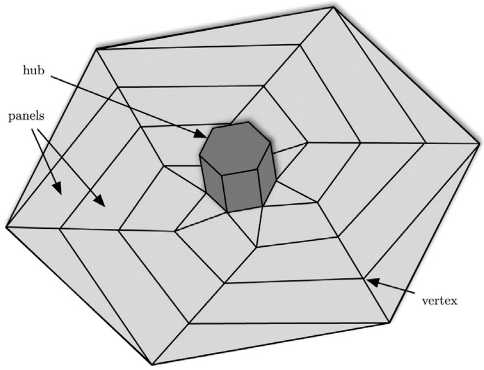

Fig. 1. Example structure concept: A spacecraft hub with a radially folding deployable structure.

behavior of various origami structures [6–9], and have investigated using dynamic excitation to control origami transformations [10]. However, modeling dynamic motion of origami structures is still in early development. Recent studies developing folding structure concepts have adapted pre-existing software tools for dynamics analysis such as MathWorks SimScape Multibody [11], or JPL's DARTs [12] simulation toolkit. However these tools are not necessarily intended for processing the high volume of closed-chain constraints presented by a folding system. This point is highlighted in a previous folded structure study where the fold pattern was designed specifically to avoid the presence of closed chains entirely for computational simplicity [12]. This motivates the need for a fast dynamics simulation approach to be developed. In this paper, an approach for modeling the dynamics using advanced multi-body techniques is developed. This method provides a fast simulation where complex hinges can be implemented at the folds to actuate the deployment in future studies.

A few key assumptions are ingrained in the construction of this approach. It is assumed that any pattern modeled using this framework will only contain four panel vertices. Additionally, the fold lines of the pattern are treated as delineations between panels that are assumed to be rigid. Therefore, this approach is only appropriate for structures where the material of the fold hinge is sufficiently more flexible than the panels, and the panels are stiff enough that this assumption is valid. It is also assumed that only loop constraints are enforced on the closed chain systems. Finally, it is assumed that the base-body of the structure is a free-flying spacecraft system, meaning the body is not rigidly attached to the ground and has six degrees of freedom. This assumption enables an algorithm shortcut in the constraint calculations and is consistent with the scope of the research.

2. Dynamics and multi-body systems fundamentals

2.1. Spatial vector kinematics

The dynamics algorithms are structured using spatial vectors for computational and mathematical efficiency. Spatial vector algebra uses six dimensional representations of rigid body properties to capture both the rotational and linear components in a single expression. For example, a rigid body's orientation and position, referred to as the spatial coordinates $\boldsymbol{q}$ of frame $\mathcal{G}$ with respect to frame $\mathcal{F}$ is expressed as

$$\boldsymbol{q}(\mathcal{F},\mathcal{G}) = \begin{bmatrix} \boldsymbol{\sigma}(\mathcal{F},\mathcal{G}) \\ \boldsymbol{l}(\mathcal{F},\mathcal{G}) \end{bmatrix} \tag{1}$$

where $\boldsymbol{\sigma}$ is a three coordinate representation of orientation and $\boldsymbol{l}$ is the position vector in 3D Euclidean space. In this application, the spacecraft orientation is represented by the standard Modified Rodriguez Parameters (MRPs) [13] with shadow set switching, where an alternative MRP set, the shadow set, is also used to avoid geometric singularities. The spatial velocity is chosen as the angular rotation rates and the linear velocities of the body

$$\boldsymbol{\beta}(\mathcal{F},\mathcal{G}) = \begin{bmatrix} \boldsymbol{\omega}(\mathcal{F},\mathcal{G}) \\ \boldsymbol{v}(\mathcal{F},\mathcal{G}) \end{bmatrix} \tag{2}$$

where the relative angular velocity is a non-integrable quasi-velocity, meaning it is not the time derivative of the spatial coordinates, and the notation $\boldsymbol{\omega}(\mathcal{F},\mathcal{G})$ denotes the angular velocity of frame $\mathcal{G}$ with respect to frame $\mathcal{F}$. The time derivative of the spatial orientations and spatial angular velocities are then related to each other using a linear transformation. For MRPs, this transformation is as follows, where the tilde operator represents the skew symmetric matrix implementing the cross product [13]

$$\dot{\boldsymbol{\sigma}} = \frac{1}{4}\left[(1-\boldsymbol{\sigma}^2)[I_{3\times 3}] + 2[\tilde{\boldsymbol{\sigma}}] + 2\boldsymbol{\sigma}\boldsymbol{\sigma}^{\intercal}\right]\boldsymbol{\omega} = [B]^{\omega}\boldsymbol{\omega} \tag{3}$$

then the full spatial transformation from spatial velocity to generalized coordinate derivatives is

$$\dot{q} = [B]\boldsymbol{\beta} = \begin{bmatrix} [B]^{\omega} & 0_3 \\ 0_3 & I_{3\times 3} \end{bmatrix}\boldsymbol{\beta} \tag{4}$$

2.2. Serial-chain ABFD framework

A prominent dynamics algorithm developed for serial chains is presented in literature as the $\mathcal{O}(\mathcal{N})$ Articulated-Body Forward Dynamics (ABFD) algorithm developed independently by Featherstone [14] and Rodriguez [15], and detailed in a unified manner by Jain [16]. Here $\mathcal{N}$ refers to the total number of velocity degrees of freedom in the system. The algorithm is developed to be appropriate for any multibody robotic system that is treated as a network of serial-chain rigid bodies. The full derivation of the algorithm can be reviewed in the literature, but key features are noted here to provide context to the adaptations developed for spacecraft and deployable structure systems. The variables referenced in this paper are consistent with those in the literature [1] and previous work [17,18]. In the articulated-body model, each of the rigid bodies down-chain of the current body being considered are treated as completely free with zero hinge force. Under this assumption, the articulated body inertia is calculated to represent those free bodies and a correction term is then developed to compensate for this assumption. Additionally, the ABFD algorithm can be expanded to handle the multiple serial-chain branches of a tree-topology case.

The ABFD framework outlined by Jain [1] provides the basis of the version implemented here, with a few key adaptations that are described as needed. Here, inboard refers to being in the direction towards the base of the chain and outboard refers to towards the tip. The generalized spatial coordinates are chosen as hinge coordinates at the kth hinge, or the kth rigid body's outboard hinge frame, $\mathcal{O}_k$, orientation and position with respect to the $k+1$ rigid body's inboard hinge frame, $\mathcal{O}_k^+$, as illustrated in Fig. 2,

$$\boldsymbol{q}(k) = \begin{bmatrix} \boldsymbol{\sigma}(\mathcal{O}_k^+,\mathcal{O}_k) \\ \boldsymbol{l}(\mathcal{O}_k^+,\mathcal{O}_k) \end{bmatrix} \tag{5}$$

and the generalized velocities are chosen as the hinge spatial velocities, taken as the time derivative with respect to the k frame

$$\boldsymbol{\beta}(k) = \begin{bmatrix} \boldsymbol{\omega}(\mathcal{O}_k^+,\mathcal{O}_k) \\ \boldsymbol{v}(\mathcal{O}_k^+,\mathcal{O}_k) \end{bmatrix} \tag{6}$$

For a given set of rigid bodies, these are collected in the full coordinate and velocity sets

$$q = \begin{bmatrix} \boldsymbol{q}(1) & \dots & \boldsymbol{q}(k) & \dots & \boldsymbol{q}(n) \end{bmatrix}^{\intercal} \quad \beta = \begin{bmatrix} \boldsymbol{\beta}(1) & \dots & \boldsymbol{\beta}(k) & \dots & \boldsymbol{\beta}(n) \end{bmatrix}^{\intercal} \tag{7}$$

where the tip of the chain is denoted as body 1 and the base body is denoted as body n. This leads to system equations of motion in the form

$$\mathcal{M}(q)\dot{\beta} + C(q,\beta) = T \tag{8}$$

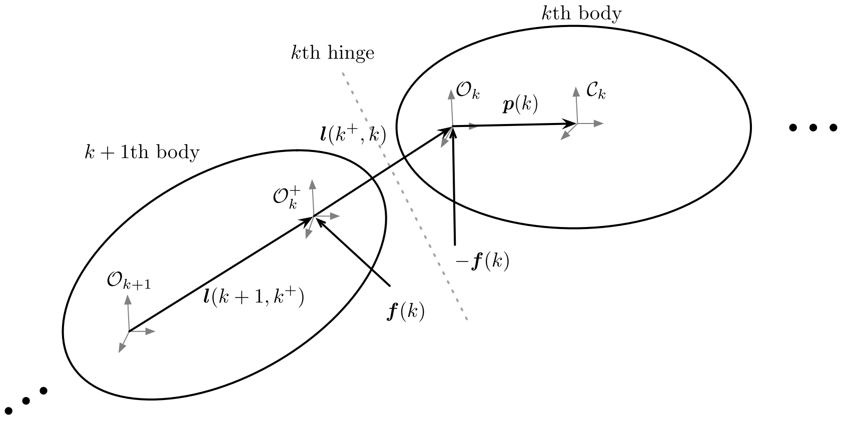

Fig. 2. Vector and frame notation between the $k+1$th and the kth body. The base of the chain is past the left of the diagram and the tip is past the right. The kth hinge refers to the interface between these two bodies, defined by the relative orientation and position, $l(\mathcal{O}_k^+, \mathcal{O}_k)$, between hinge frames $\mathcal{O}_k^+$ and $\mathcal{O}_k$.

where $\mathcal{M}(q)$ is the full system mass matrix, $C(q,\beta)$ contains the Coriolis contributions, and T is the vector of system generalized forces. The use of the quasi-velocities diverges from the assumptions implemented in Jain's text [1]. In the forward dynamics problem, q, β and T are known quantities and the time derivative $\dot{\beta}$ is the desired quantity. Direct inversion of the mass matrix $\mathcal{M}$ is typically done for small order systems, but is a computationally expensive $\mathcal{O}(\mathcal{N}^3)$ matrix operation for an $\mathcal{N}$ degree of freedom problem. This becomes prohibitively slow for large DOF multi-body systems. The computational efficiency of the ABFD algorithm is achieved by applying the Innovations Operator Factorization of the mass matrix $\mathcal{M}$ and deriving an explicit and analytical expression of the inverse, $\mathcal{M}^{-1}$. The details of this factorization are left to the literature [1]. The dynamics as implemented here are derived using body frame derivatives. The single chain algorithm is set up in the following way. First, a recursive sweep that solves the velocities and Coriolis accelerations of the chain is run from the base body to the tip. Then, the articulated body inertias and corrections are solved for in a tip to base recursion. The final step is to do a base to tip recursion to solve for the body accelerations, yielding the system equations of motion.

2.3. Framework for complex hinge behavior

2.3.1. Hinge mapping

The interaction of adjacent bodies in the chain are governed by the properties of the hinge connecting these bodies. The hinge map matrix, $H^\intercal(k)$, for a rigid body joint k defines the configuration dependence of the hinge behavior and maps the relative hinge velocities to the full six DOF spatial velocities of the body. Where $r_v(k)$ is the number of velocity degrees of freedom across the hinge, the dimension of the hinge map matrix is $H^\intercal(k) \in R^{6\times r_v(k)}$. For a free-floating rigid body in space, the hinge map matrix is a 6×6 identity matrix, I_6. Therefore, a free-floating spacecraft base-body is mapped to inertial space with $H(r) = I_6$. This mapping introduces a simple and modular way to implement various velocity constraints across the hinge of two adjacent bodies, by removing velocity degrees of freedom that are constrained by the hinge, without reformulation of the dynamics algorithm for a different number of degrees of freedom. Then for folding panels that are constrained to a single rotation along the fold axis of the pattern, where the fold axis is aligned with the first axis of the frame,

$$H(k) = [1,0,0,0,0,0] \tag{9}$$

This only applies the hinge constraint at each connected hinge of a free serial chain. Hinge properties spanning the cut edges of a closed-chain graph are captured in loop constraints, covered in Section 4.1.

2.3.2. Internal hinge forces

The spatial force acting at hinge k due to the interaction with body $k+1$ is denoted $f(k)$, where $f(k)$ acts at the O_k hinge frame and an equal but opposite force $-f(k)$ acts at the O_k^+ frame on the $k+1$ body. Then the generalized force on the kth hinge, $T(k)$, is the projection of the spatial force through the hinge degrees of freedom, defined as

$$T(k) = H(k)f(k) \tag{10}$$

This force can be defined by the components in the hinge system. Examples of simple uncontrolled hinge forces are linear and torsion springs. Due to the pursuit of a free, self-actuated deployment system, the hinges are expected to contain strain or potential energy driven forces that are a function of the general coordinates. For example, a linear torsion spring with magnitude K_1 along the first rotation θ_1 would be expressed in spatial coordinates for hinge k as

$$f(k) = K(k)q(k) = \begin{bmatrix} K_1\theta_1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^\intercal \tag{11}$$

2.4. Conserved principles for multi-body systems

The conservation of energy and the conservation of momentum provide robust verification of dynamic systems modeling such as the approach applied here. These principles are defined for this study in spatial notation as follows.

2.4.1. Energy

The energy of a body is the same regardless of the point it is measured from on the body, and therefore for a given body k, the kinetic energy of the body about its hinge frame is the same as the kinetic energy about its center of mass. In spatial coordinates, this can be expressed as

$$KE(k) = \frac{1}{2}V(k)^\intercal M(k)V(k) \tag{12}$$

where $V(k)$ is the spatial velocity and $M(k)$ is the spatial mass matrix. For a hinge with linear springs, the potential energy is

$$PE(k) = \frac{1}{2}q(k)^\intercal K(k)q(k) \tag{13}$$

where $K(k)$ is the stiffness matrix for the kth hinge, assuming a linear spring force as a function of the hinge general coordinates. The energy calculated at each body is invariant to the frame that it is calculated at, so the total system energy can be calculated, independent of frame as

$$E = \sum_{k=1}^{n} KE(k) + PE(k) \tag{14}$$

2.4.2. Angular momentum

The magnitude of the angular momentum of a single body about the body center of mass, c, is conserved, where the angular momentum can be written in spatial coordinates as

$$h(c) = \boldsymbol{J}(c)\omega(c) \tag{15}$$

where $\boldsymbol{J}(c)$ is the inertial about the center of mass and $\omega(c)$ is the angular rates of body.

For a system of rigid bodies, the angular momentum of each body must be expressed in the same frame and taken about the system's center of mass, c_{sys} to demonstrate conservation. Therefore, the angular momentum of the system is calculated as

$$h = \sum_{k=1}^{n} h_{c_{\text{sys}}}(k) \tag{16}$$

where

$$h_{c_{\text{sys}}}(k) = \boldsymbol{J}(c_k)\boldsymbol{\omega}(k) + m(k)(\boldsymbol{p}(c_{\text{sys}}, c_k) \times \boldsymbol{v}(c_k)) \tag{17}$$

Here $\boldsymbol{J}(c_k)$ is the inertia of the kth body about its center of mass, $\boldsymbol{\omega}(k)$ is the angular velocity of the kth body, $m(k)$ is the mass of the kth body, $\boldsymbol{v}(c_k)$ is the linear velocity of the kth body center of mass, and $\boldsymbol{p}(c_{\text{sys}}, c_k)$ is the position vector from the system center of mass frame to the center of mass of the kth body.

3. Folded structure topology processing

3.1. Graph theory applications

A system of hinge-connected rigid bodies can be described using graph theory by treating the rigid bodies as nodes and the hinges or fold lines as edges. This representation will aid in breaking down the complex system into a form that can be efficiently analyzed. The manner in which the system of nodes is connected determines the classification of the system. For a given graph, the node from which an edge leads is designated the parent node and the node at the destination of that edge is referred to as the child node. A node with no parent is the root node. This is depicted in Fig. 3, where the arrowed lines, representing edges, point from parent nodes to child nodes, and the root node is labeled with r. A parent node can have multiple child nodes, and if these nodes do not share edges within the graph, the graph is referred to as a tree topology. The basis of the dynamics algorithm discussed here is written to recursively solve for a serial chain of bodies, following the branch of a tree. At initial consideration, the closed-loop patterns of a folded spacecraft structure is a multiply connected graph where multiple child nodes span from a parent node and are interconnected, and there exist paths in the graph that lead back to a given node. The first step in modeling a folded spacecraft structure is to identify edges of the system to "cut" such that the bodies are segmented into a topology where there are no closed loops, also known as a tree topology. These cut edges must then be constrained with corrections to enforce the actual closed-chain topology. In Fig. 3, these cut edges are shown as dashed lines.

3.2. Tree topology of planar origami patterns

The development and analysis of origami-inspired fold patterns appropriate for use in spacecraft structures is an active area of interest. A select number of patterns have received more study due to the clear applicability to spacecraft needs. The Miura pattern [3], illustrated in Fig. 3, is a highly efficient folding scheme with one theoretical degree of freedom that deploys linearly in dual directions and is thoroughly studied in the literature. Similarly, the Scheel pattern [4] illustrated in Fig. 4 is a radially wrapped pattern that is commonly studied for spacecraft structure applications, where the center disk serves as the spacecraft bus.

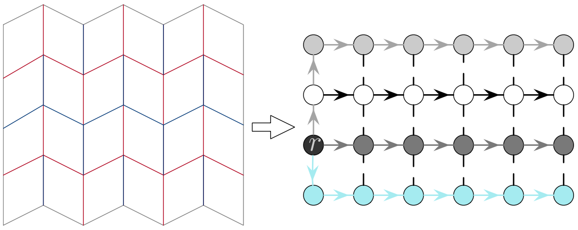

Fig. 3. Miura folding pattern and example system graph and cut edges where r denotes the root node.

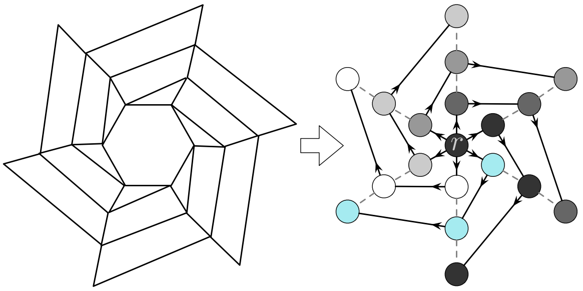

Fig. 4. Scheel folding pattern and example graph with cut edges where r denotes the root node.

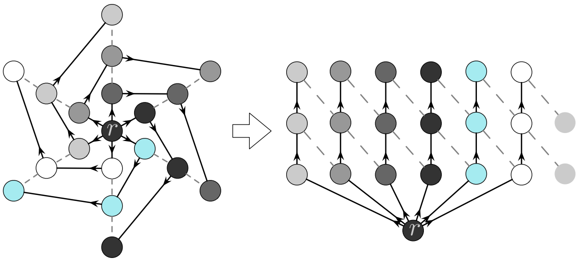

Fig. 5. Scheel folding pattern graph adapted to a grid format, where the closed grid is represented by the closure constraints on the repeated left edge chain's nodes.

Figs. 3 and 4 also display example graph patterns for their corresponding origami pattern. The patterns are segmented such that a single root parent node spawns the serial chains of the origami pattern in a manner that itself displays a repeatable and expandable pattern. These serial chains are then constrained to each other at each adjacent node of their chains. For algorithm processing, it is assumed that the root node is always the free flying spacecraft body. The pattern is then defined through declaring each chain series and defining each set of constraint nodes. For this approach, these tree topologies are assumed to be cut and defined such that they form an organized grid , as clearly seen in the graph of Fig. 3. A graph like the Scheel pattern in Fig. 4 can be adapted to mimic a grid with minimal adaptation, as demonstrated in Fig. 5. The chains of the structure are laid out like a grid, and the constraint nodes are defined as the dashed lines. This system will require an additional set of closure nodes defined between the chains on the edge of the grid (represented by a repeated set of the leftmost chain).

3.3. Constraints for grid adapted tree topologies

A given panel can have more than one constraint node, as is present where there are three or more chains in a pattern. The cut kinematic chains are defined by recording the chain sequence in terms of the named bodies in the chain from tip to base in the chain matrix κ as

$$\kappa_a = \begin{bmatrix} a(1) & \dots & a(n_a) \end{bmatrix} \tag{18}$$

for chain a with n_a bodies. For reference, n_p are the number of constraint node pairs or number of implemented constraints, n_c are the total number of constraint nodes, n_b are the number of rigid bodies in the system, and n_h are the number of chains in the cut tree topology. Then the constraints information is stored in the n_p×2 constraint node matrix, Γ, containing the constraint node pair designations. For example,

$$\Gamma = \begin{bmatrix} a(1) & b(1) \\ \dots & \end{bmatrix} \tag{19}$$

For a given set of bodies connected in a grid format that does not close onto itself, like the Miura pattern, the total number of constraints needed to adapt the set to a tree-topology system is summarized by Eq. (20) and the total number of constraint nodes on the system can be predicted by Eq. (21), assuming constraint nodes are unique to a constraint pair.

$$n_p = \left(n_h - 1\right)\left(\frac{n_b}{n_h} - 1\right) \tag{20}$$

$$n_c = n_b - n_h \tag{21}$$

These are needed for constraint generating algorithms. Similar calculations can be derived for radially closed patterns by simply including the additional closure nodes.

4. Closed-chain forward dynamics

The recursive forward dynamics algorithms for a serial chain are expanded from the literature to accommodate the generalized tree topology framework needed to handle folded structures [18]. Then, the closed-chain constraints must be developed. As discussed in Section 3.2, capturing the closed-chain behavior is achieved by cutting an edge of a closed-chain system and treating each leg of the cut as an open serial chain, emulating a tree topology. Then the cut edges are treated as motion constraints imposed on the free dynamics of the tree. There are several approaches to enforcing the closure constraints. The augmented approach compensates for the cut edge by including a correction acceleration, resulting in additional motion constraint equations and a non-minimal coordinate set [19]. When this is combined with a cut tree graph, it is referred to as the tree-augmented approach. This approach faces issues with error drift that must be compensated for with error control techniques. The direct approach uses matrix solvers and absolute coordinates, resulting in a much larger system and greater computational complexity [1]. This approach also shares similar issues as the augmented approach, and therefore is not considered, as the augmented approach is more desirable for this application. A new technique that provides a minimal coordinate set is the constraint embedding approach [20]. In this approach, the non-tree graph is transformed into a tree topology by aggregating the closed-chain structures of the topology into a representative node. This is suitable for systems with a clear tree-like structure surrounding the closed-chain elements. The folded structures of interest contain multiple dependent systems of closed loops, as demonstrated in Figs. 3 and 4, and therefore this approach is not well suited to the problems of interest and is not currently considered. Therefore, the tree-augmented approach is selected and developed for the general origami-folded spacecraft structure. Custom algorithms are adapted from the literature to handle the large number of rigid bodies subjected to closed-chain constraints across multiple serial chains.

4.1. Tree-augmented approach to closed chain structures

Implementing the correction terms to account for the motion constraints is captured in the system equations of motion by introducing the Lagrange Multipliers [1], denoted as λ, to represent the constraint forces. Additionally, a new set of equations must be considered to include the constraint expression. The generalized acceleration is then defined as

$$\dot{\beta} = \dot{\beta}_f + \dot{\beta}_c \tag{22}$$

where $\dot{\beta}_f$ are the free unconstrained accelerations and $\dot{\beta}_c$ are the correction accelerations. The correction acceleration is derived from the constraint expression, and is expressed in terms of global system spatial operators as [1]

$$\dot{\beta}_c = [I - H\phi K]D^{-1}H\phi\mathcal{B}Q^{\intercal}\lambda \tag{23}$$

where H is the global hinge map matrix defining hinge behavior, ϕ is the $6n_b \times 6n_b$ global spatial transformation matrix, K is the $6n_b \times 6n_b$ spatial operator referred to in literature as the shifted Kalman gain operator, D is the $6n_b \times 6n_b$ articulated body hinge inertia, and $\mathcal{B}$ is the $6n_b \times 6n_c$ node pick-off operator that transforms information from the body frame to the relevant constraint nodes on the body. These quantities are discussed in length in the literature [1] and together they essentially enable the equations of motion to be calculated without mass matrix inversion. Q is the $n_{c_{\mathrm{DOF}}}$×$6n_c$ constraint matrix that defines the constrained spatial degrees of freedom between nodes where $n_{c_{\mathrm{DOF}}}$ are the total number of constrained node pair degrees of freedom. For a node that is rigidly constrained to another, the corresponding entry in Q is a 6 × 6 identity matrix. Finally, λ is the $n_{c_{\mathrm{DOF}}} \times 1$ Lagrange multipliers. These are defined for loop constraints as

$$\lambda = -[Q\underline{\Lambda}Q^{\intercal}]^{-1}\ddot{\boldsymbol{\Phi}} \tag{24}$$

where $\underline{\Lambda}$ is the operation space compliance matrix

$$\underline{\Lambda} = \mathcal{B}^{\intercal}\Omega\mathcal{B} \tag{25}$$

and Ω is the extended operational space compliance matrix. Additionally, $\ddot{\boldsymbol{\Phi}}(\beta, t)$ is the derivative of a Pfaffian form constraint equation, $\dot{\boldsymbol{\Phi}}(\beta, t)$.

4.1.1. Spatial constraint equations for folded structures

For any set of two closure nodes for a single closed loop in the system, up to six constraint equations can be written in the global spatial coordinates, three for position and three for rotations. For a rigorous derivation of the constraints, these will be considered separately and then interpreted to a general spatial format for implementation. First considering the position of the closure nodes, a constraint equation that defines the two nodes must be in the same place is written as

$$\boldsymbol{\Phi}^{\boldsymbol{v}} = \boldsymbol{l}(1_{nd}) - \boldsymbol{l}(2_{nd}) = \boldsymbol{0} \tag{26}$$

where the location of the nodes can be written in terms of the hinge frames as

$$\boldsymbol{\Phi}^{\boldsymbol{v}} = (\boldsymbol{l}(1) + \boldsymbol{l}(1, 1_{nd})) - (\boldsymbol{l}(2) + \boldsymbol{l}(2, 2_{nd})) = \boldsymbol{0} \tag{27}$$

Then taking the first time derivative

$$\dot{\boldsymbol{\Phi}}^{\boldsymbol{v}} = \frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{l}(1) + \boldsymbol{l}(1, 1_{nd})) - \frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{l}(2) + \boldsymbol{l}(2, 2_{nd})) = \boldsymbol{0} \tag{28}$$

and using the Transport theorem,

$$\dot{\boldsymbol{\Phi}}^{\boldsymbol{v}} = (\boldsymbol{v}(1) + \boldsymbol{\omega}(1) \times \boldsymbol{l}(1, 1_{nd})) - (\boldsymbol{v}(2) + \boldsymbol{\omega}(2) \times \boldsymbol{l}(2, 2_{nd})) = \boldsymbol{0} \tag{29}$$

Recognizing the relationship with the node pick-off operator, the velocity form of the position loop constraint is

$$\dot{\boldsymbol{\Phi}}^{\boldsymbol{v}} = \boldsymbol{v}(1_{nd}) - \boldsymbol{v}(2_{nd}) = \boldsymbol{0} \tag{30}$$

Then taking the second derivative,

$$\ddot{\boldsymbol{\Phi}}^{\boldsymbol{v}} = \frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{v}(1) + \boldsymbol{\omega}(1) \times \boldsymbol{l}(1, 1_{nd})) - \frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{v}(2) + \boldsymbol{\omega}(2) \times \boldsymbol{l}(2, 2_{nd})) = \boldsymbol{0} \tag{31}$$

$$\begin{aligned}\ddot{\boldsymbol{\Phi}}^{\boldsymbol{v}} = &\left(\boldsymbol{\alpha}^{\boldsymbol{v}}(1) + \left(\dot{\boldsymbol{\omega}}(1) \times \boldsymbol{l}(1, 1_{nd})\right) + \boldsymbol{\omega}(1) \times \left(\boldsymbol{\omega}(1) \times \boldsymbol{l}(1, 1_{nd})\right)\right) - \\ &\left(\boldsymbol{\alpha}^{\boldsymbol{v}}(2) + \left(\dot{\boldsymbol{\omega}}(2) \times \boldsymbol{l}(2, 2_{nd})\right) + \boldsymbol{\omega}(2) \times \left(\boldsymbol{\omega}(2) \times \boldsymbol{l}(2, 2_{nd})\right)\right) = \boldsymbol{0}\end{aligned} \tag{32}$$

and rewriting in terms of node information,

$$\begin{aligned}\ddot{\boldsymbol{\Phi}}^{\boldsymbol{v}} = &\left(\boldsymbol{\alpha}^{\boldsymbol{v}}(1_{nd}) + \boldsymbol{\omega}(1) \times \left(\boldsymbol{\omega}(1) \times \boldsymbol{l}(1, 1_{nd})\right)\right) - \\ &\left(\boldsymbol{\alpha}^{\boldsymbol{v}}(2_{nd}) + \boldsymbol{\omega}(2) \times \left(\boldsymbol{\omega}(2) \times \boldsymbol{l}(2, 2_{nd})\right)\right) = \boldsymbol{0}\end{aligned} \tag{33}$$

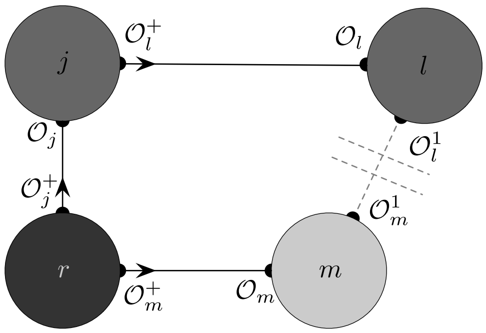

Fig. 6. System graph and frame notation of 4 body closed-chain structure.

Table 1
Mass properties of the rigid root body and panel bodies.

Body	$m(k)$ (kg)	$K(k)$ (N m/rad)	Length (m)	Inertia (kg m^2)
r	1	0	2	diag $(0.3342, 0.3342, 0.0017)$
m	1	0.1	2	diag $(0.3342, 0.3342, 0.0017)$
l	1	0.1	2	diag $(0.3342, 0.3342, 0.0017)$
j	1	0.1	2	diag $(0.3342, 0.3342, 0.0017)$

Table 2
Geometry properties of the rigid bodies. $\boldsymbol{p}(i,j)$ is a position vector from frame i to frame j.

Body	$\theta(c, k-1^+)$ (rad)	$\boldsymbol{p}(c, k-1^+)$ (m)	$\boldsymbol{p}(k, c)$ (m)
r_1	$[0,0,0]$	$[0,1,0]$	$[0,0,0]$
r_2	$[0,0,\pi/2]$	$[1,0,0]$	$[0,0,0]$
m	$[0,0,0]$	$[0,1,0]$	$[0,0,0]$
l	$[0,0,0]$	$[0,1,0]$	$[0,0,0]$
j	$[0,0,-\pi/2]$	$[1,0,0]$	$[0,-1,0]$

Table 3
Geometry properties of the constraint nodes.

i	$\theta(k, \mathcal{N}_{k_i})$ (rad)	$\boldsymbol{p}(k, \mathcal{N}_{k_i})$ (m)
1	$[0,0,\pi/2]$	$[0,-2,0]$
2	$[0,0,\pi/2]$	$[0,2,0]$

Table 4
Initial conditions of the numerical simulation.

Body	q	$\dot{\beta}$
r	$[0,0,0,0,0,0]$	$[0,0,0,0,0,0]$
m	$-\pi$	0
l	$-\pi$	0
j	π	0

where $\boldsymbol{\alpha}^{\boldsymbol{v}}(k)$ is the free, unconstrained linear acceleration of body k, and $\boldsymbol{\alpha}^{\boldsymbol{v}}(k_{nd})$ is the free, unconstrained linear acceleration of the constraint node points on body k.

Now considering the constraint derivation for the rotational components of the nodes, in order to avoid complications due to the choice of non-integrable rates $\boldsymbol{\omega}(k)$ for the generalized coordinates, a non-holonomic Pfaffian constraint is written as a function of those rates. This use of rate-based constraints will introduce error control concerns, where correcting on rates introduces greater errors in the coordinates due to numerical integration error.

$$\dot{\boldsymbol{\Phi}}^{\boldsymbol{\omega}} = \boldsymbol{\omega}(1_{nd}) - \boldsymbol{\omega}(2_{nd}) = \boldsymbol{\omega}(1) - \boldsymbol{\omega}(2) = \mathbf{0} \tag{34}$$

Then taking the time derivative

$$\ddot{\boldsymbol{\Phi}}^{\boldsymbol{\omega}} = \dot{\boldsymbol{\omega}}(1) - \dot{\boldsymbol{\omega}}(2) = \mathbf{0} \tag{35}$$

$$\ddot{\boldsymbol{\Phi}}^{\boldsymbol{\omega}} = \boldsymbol{\alpha}_{\boldsymbol{\omega}}(1) - \boldsymbol{\alpha}_{\boldsymbol{\omega}}(2) = \mathbf{0} \tag{36}$$

where $\boldsymbol{\alpha}^{\boldsymbol{\omega}}(k)$ is the free, unconstrained rotational acceleration of body k, and $\boldsymbol{\omega}(k_{nd}) = \boldsymbol{\omega}(k)$ because these points are on the same rigid body.

Now considering the spatial operator format of the algorithms discussed thus far, the constraint formulation is restructured into a single spatial expression as

$$\begin{aligned}\ddot{\boldsymbol{\Phi}} = {} & \phi^{\intercal}(1, 1_{nd})\boldsymbol{\alpha}(1) - \phi^{\intercal}(2, 2_{nd})\boldsymbol{\alpha}(2) + \\ & \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\omega}(1) \times \left(\boldsymbol{\omega}(1) \times \boldsymbol{l}(1, 1_{nd})\right) - \boldsymbol{\omega}(2) \times \left(\boldsymbol{\omega}(2) \times \boldsymbol{l}(2, 2_{nd})\right) \end{bmatrix} = \mathbf{0}\end{aligned} \tag{37}$$

Expanding to include the constraint matrix formulation for constraint design flexibility,

$$\ddot{\boldsymbol{\Phi}} = [Q] \begin{bmatrix} \phi^{\intercal}(1, 1_{nd})\boldsymbol{\alpha}(1) + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\omega}(1) \times \left(\boldsymbol{\omega}(1) \times \boldsymbol{l}(1, 1_{nd})\right) \end{bmatrix} \\ \phi^{\intercal}(2, 2_{nd})\boldsymbol{\alpha}(2) + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\omega}(2) \times \left(\boldsymbol{\omega}(2) \times \boldsymbol{l}(2, 2_{nd})\right) \end{bmatrix} \end{bmatrix} = \mathbf{0} \tag{38}$$

which can be written in a general compact form as

$$\ddot{\boldsymbol{\Phi}} = \mathcal{Q}\left(\mathcal{B}^{\intercal}\boldsymbol{\alpha} + \mathcal{U}\right) \tag{39}$$

This two node derivation is applicable to multiple closed chain constraints within a system by simply considering the formatting of the $\mathcal{Q}$ and $\mathcal{B}$ matrices in those cases. Special care must be applied to generating the $\mathcal{U}$ vector appropriately for this expanded case.

4.1.2. Baumgarte stabilization of constraint enforcement

The forward dynamics problem for rigid bodies subject to closure constraints is unstable in numerical computation due to numerical error in the standard integration of the equations of motion [21]. Initial numerical demonstrations were found to be very unstable on the time scales of interest and under the actuation of internal forces of self-actuated deployable space structures. The most popular approach to removing this instability is to use Baumgarte Stabilization [22], and this method is implemented here. A correction gain is included to the constraint equation, $\mathfrak{f}$, such that instead of writing

$$\mathfrak{f} = \ddot{\boldsymbol{\Phi}} = \mathbf{0} \tag{40}$$

the constraint equation is defined as

$$\mathfrak{f} = \ddot{\boldsymbol{\Phi}} + 2\mathfrak{a}\dot{\boldsymbol{\Phi}} + \mathfrak{b}^2\boldsymbol{\Phi} = \mathbf{0} \tag{41}$$

where $\mathfrak{f}$ is stable for any positive values of $\mathfrak{a}$ and $\mathfrak{b}$, which are tuned uniquely for each system they are implemented in. Then the Lagrange multipliers for loop constraints are

$$\lambda = -[\mathcal{Q}\underline{\Lambda}\mathcal{Q}^{\intercal}]^{-1}\ddot{\mathfrak{f}} \tag{42}$$

and the constraint force that is applied to the rigid body system is defined as

$$f_c = -\mathcal{Q}^{\intercal}\lambda \tag{43}$$

4.2. Four-body closed loop map folded structure

The closed-chain theory is now demonstrated in detail for the four-body structure case. In previous work [23], the author evaluates and validates this approach by comparing results of matching case studies with a commercial software, Abaqus. Using the notation displayed in Fig. 6, the cut edge is selected at the internal edge connecting nodes l and m, where the root parent node is selected as node r. Due to the non-integrable spatial velocities, the closure constraint is better expressed as a non-holonomic constraint, expressed in the Pfaffian form, as derived in Section 4.1. The points on the body where the constraint is to be applied must first be defined. Where frame $\mathcal{O}_m$ denotes the mth link hinge frame connecting to the body's predecessor in the chain, the single outboard frame where the closure is connected is denoted as $\mathcal{O}_m^1$. Similarly, the outboard frame of the lth link's closure point is denoted as $\mathcal{O}_l^1$, as illustrated in Fig. 6. Then the spatial velocity at these closure

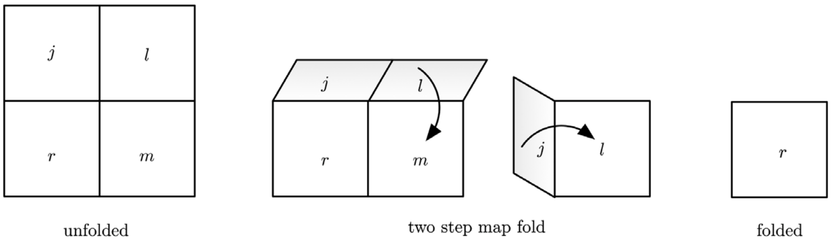

Fig. 7. Stages of a four body planar map from unfolded to folded configurations.

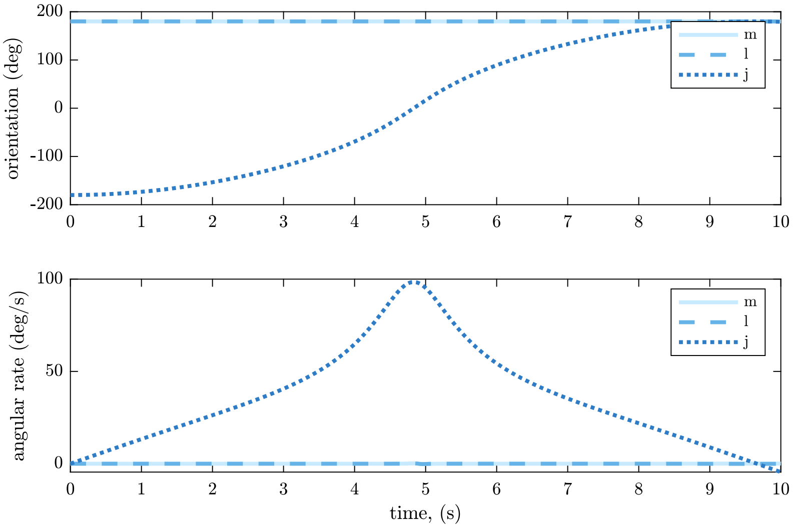

Fig. 8. Angular orientation and rates of the three panel bodies.

nodes, V_{nd}, can be mapped from the global spatial velocity at the body hinges using the pick-off spatial operator, $\mathcal{B}$.

The numerical demonstration considers a folded structure with planar panels and three dimensional spatial motions. This structure is representative of a map-folded four body structure, or a single base unit of the Miura pattern where the pattern angle is 90 deg, and a diagram of how this structure folds is displayed in Fig. 7. The structure is flat folded and assembled with only a single rotational degree of freedom on the hinge joints, captured in the initial conditions in Table 4. The Baumgarte stabilization parameters are set to $\mathfrak{a} = 100$ and $\mathfrak{b} = 0$. These values are found from hand-tuning and selecting a value that minimized the peak of the constraint violation at flat deployment without introducing an unexceptable energy or momentum violation. The second term $\mathfrak{b}$, is found to have no observable effect and is set to 0. This is because the violation is driven by the rotational behavior of the system. The rotational constraint was written in the Pfaffian form, and therefore the $\mathfrak{b}^2\boldsymbol{\Phi}$ term is only providing feedback on the translational coordinates, which have no significant contribution.

The algorithms are written in Matlab, run on a machine with a 2.5 GHz Quad-Core Intel Core i7 processor and 16 GB of memory, and a fixed step 4th order Runge–Kutta (RK4) integrator is implemented and a simulation of 10 seconds is run with a time increment of 0.001 s, resulting in a computational clock time of 281.5 s for the integration. Clock time is reported for relative evaluation between simulations of this paper. Each panel body is defined with identical geometric and mass properties as defined in Tables 1, 2, and 3. The panel motions in Fig. 8 show the unfolding of the hinge at panel j, but because of the geometry of the map fold, the hinges at l and m do not unfold, where the first hinge would need to stabilize in a flat configuration before these hinges could unfold. The free-flying host body motion is shown in Fig. 9, and is seen to rotate in the opposing direction of the panels. The constraint violations are captured in Fig. 11, where $\dot{\boldsymbol{\Phi}}^{\intercal}\dot{\boldsymbol{\Phi}}$ is the square of the spatial magnitude of the applied constraint equations and does not have physical units, and are shown to behave nonlinearly around $\theta_j = 0$. The internal constraint forces and torques are seen in Fig. 12 to be highly nonlinear around this value as well. This point is suspected to be a numerically singular configuration of the system due to the kinematic singularity of a flattened state, and special consideration must be taken for the constraint violations around these points. The numerical accuracy of the constraint enforcement for this case, with fast spatial motion, is notable. Additionally, the performance of the conservation of energy and system angular momentum shown in Fig. 10 further quantify the violations. Taking the physical system into account, the constraint violations are sub-millimeter and are considered acceptable for capturing the bulk deployment motion for the system here. However these results highlight the need for special consideration of acceptable numerical performance of the dynamics approach for deployable systems with fast self-actuated deployments. An application of this modeling approach to a physical Miura structure with high strain hinges with validation to experimental data is demonstrated by the authors is prior work [23].

4.3. Multiple constraint enforcements

Numerical test cases are developed to investigate the algorithm performance as it has been adapted for enforcing multiple constraints across multiple chains in a cut tree topology. Two cases in particular are of interest. The first is the case where there is more than one pair of constrained bodies between two chains of the cut tree topology for a system graph. The second case of interest is when there are more than two chains in the cut tree, and a single body is subjected to more than one constrained node pair. Fig. 13 displays the graphs of the two example cases designed for this study. The goal of this section is to assess the performance of the current approach as it is scaled up for larger folding space structures.

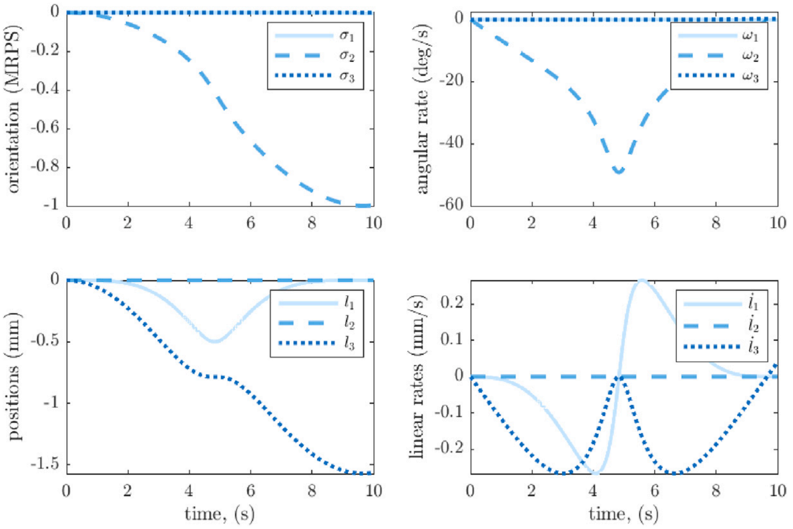

Fig. 9. States and rates of the spacecraft body in three dimensional space of the four body case.

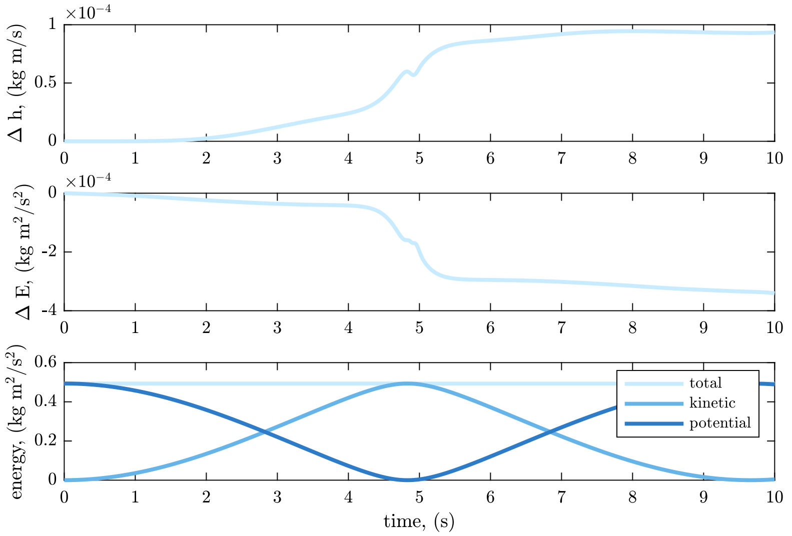

Fig. 10. Change in total system energy and total system angular momentum of the four body case.

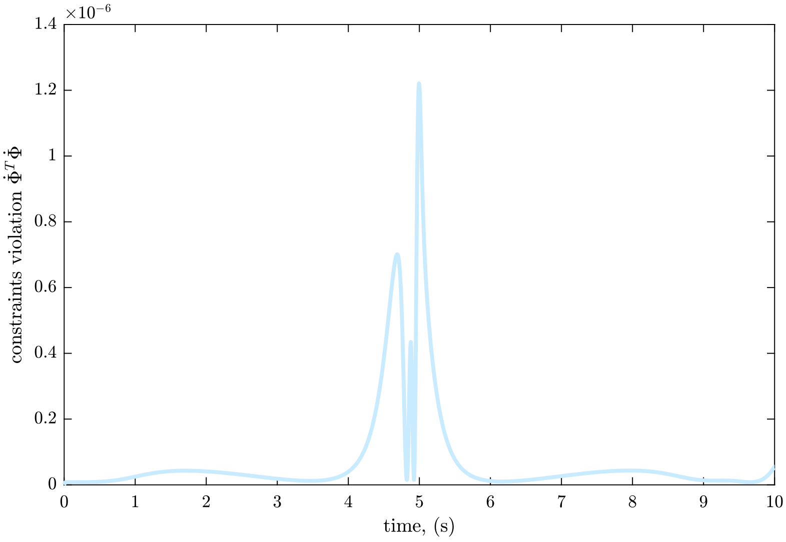

Fig. 11. Change in the velocity form of the constraint equation of the four body case.

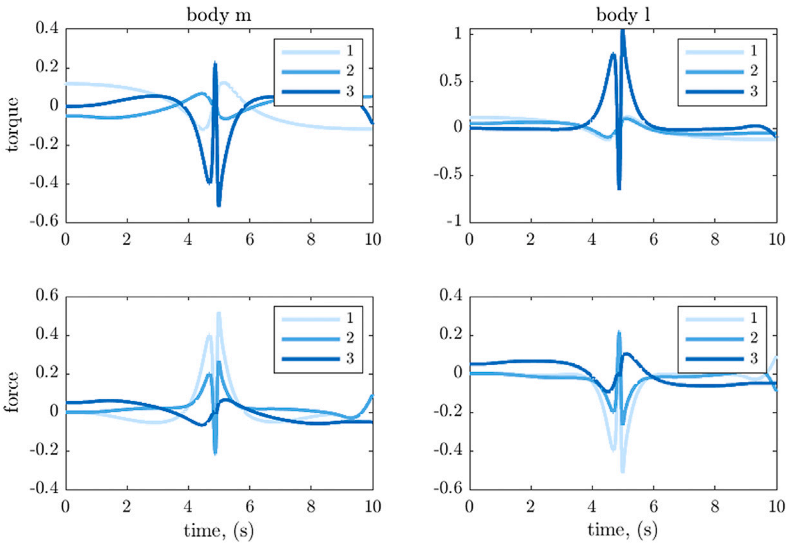

Fig. 12. Internal forces and torques to enforce the constraint equations of the four body case.

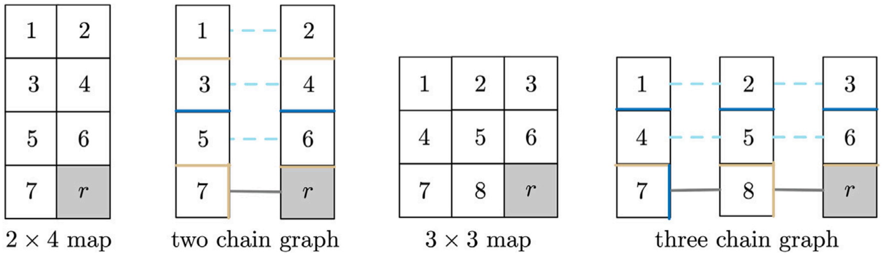

Fig. 13. Diagrams for the two and three chains in a cut tree topology.

Table 5
Mass properties of the rigid root body and panel bodies.

Body	$m(k)$ (kg)	$K(k)$ (N m/rad)	Width (m)	Thickness (m)	Inertia (kg m^2)
r	100	0	2	1	diag $(66.7, 66.7, 66.7)$
1–6	1	0.01	2	0.01	diag $(0.3342, 0.3342, 0.0017)$
7	1	0.03	2	0.01	diag $(0.3342, 0.3342, 0.0017)$
8	1	0.03	2	0.01	diag $(0.3342, 0.3342, 0.0017)$

Table 6
Geometry properties of the rigid bodies.

Body	$\theta(c_k, k-1^+)$ (rad)	$\boldsymbol{p}(c_k, k-1^+)$ (m)	$\boldsymbol{p}(k, c_k)$ (m)
r	[π/2, 0, 0]	[1, 0, 0]	[0, 0, 0]
1–6	[0, 0, 0]	[1, 0, 0]	[1, 0, 0]
7	[0, π/2, 0]	[1, 0, 0]	[0, 0, −1]
8	[0, π/2, 0]	[1, 0, 0]	[0, 0, −1]

Table 7
Geometry properties of the constraint nodes.

i	$\theta(k, \mathcal{N}_{k_i})$ (rad)	$\boldsymbol{p}(k, \mathcal{N}_{k_i})$ (m)
1	[0, π/2, 0]	[1, 0, −1]
2	[0, −π/2, 0]	[1, 0, 1]

Table 8
Initial conditions of the numerical simulation.

Body	q	$\dot{\beta}$
r	[0, 0, 0, 0, 0, 0]	[0, 0, 0, 0, 0, 0]
Mountain folds, 1, 2, 5, 6, 7	π	0
Valley folds, 3, 4	$-\pi$	0

4.3.1. *Numerical demonstration of multiple constrained bodies*

The first test case considers a system of eight bodies in two chains, with bodies referenced by number in the illustration of this dynamical system in Fig. 13. The configuration properties of the bodies are listed in Tables 5 and 6. Each fold representative hinge is constrained to a single rotation θ about the 3rd axis, is given the same linear torsional spring stiffness, and the geometry is designed such that each panel is identical. The root node, r, does not have a force acting between it and the six degree of freedom hinge to inertial space, representing a free-flying spacecraft root body. The inertia is calculated from the height, width, and thickness assuming all bodies are square. The orthogonal rotations of the $\theta(c_k, k-1^+)$ orientations represents the fold lines of a square fold and are reported in 321 Euler angles for quick physical interpretation. Changing this value and the inertia definitions would adapt the simulation from a square map fold to Miura map folds or other desired patterns. The initial conditions of the numerical demonstration are listed in Table 8 and are designed to mimic a flat folded map fold with no initial rates. Positive angle folds are representative of mountain folds and negative angle folds of valley folds. A 2 × 4 grid of rigid bodies is set up where the root body r is the branching node of the chains as shown in Fig. 3. Table 7 contains the relative position and orientation of the constraint nodes, where the frames are rotated such that the unconstrained axis is along the fold axis. The Baumgarte stabilization coefficients are determined from hand tuning and are set to $a = 40$ and $b = 0$. Using a time step of $\mathrm{d}t = 0.001$ seconds, the 30 second simulation of 14 degrees of freedom takes 11 minutes to compute using an RK4 integrator.

From Fig. 14 the states of the folded panels are seen to begin the unfolding process, where the fold between Panel 7 and the spacecraft

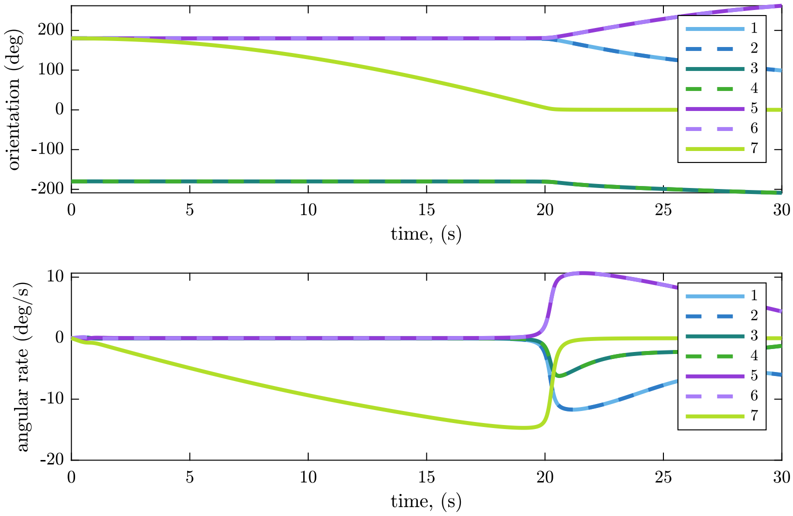

Fig. 14. Angular orientation and rates of the seven folding panel bodies.

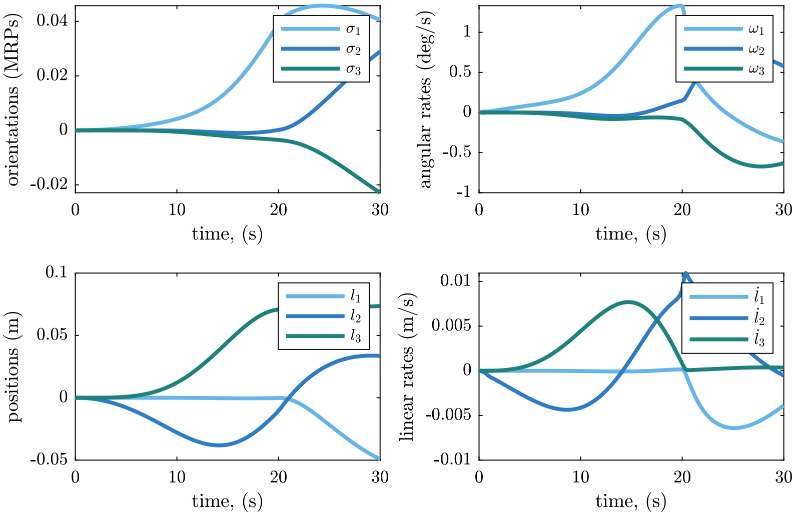

Fig. 15. Angular orientation, linear position and rates of the spacecraft body in three dimensional space of the eight body case.

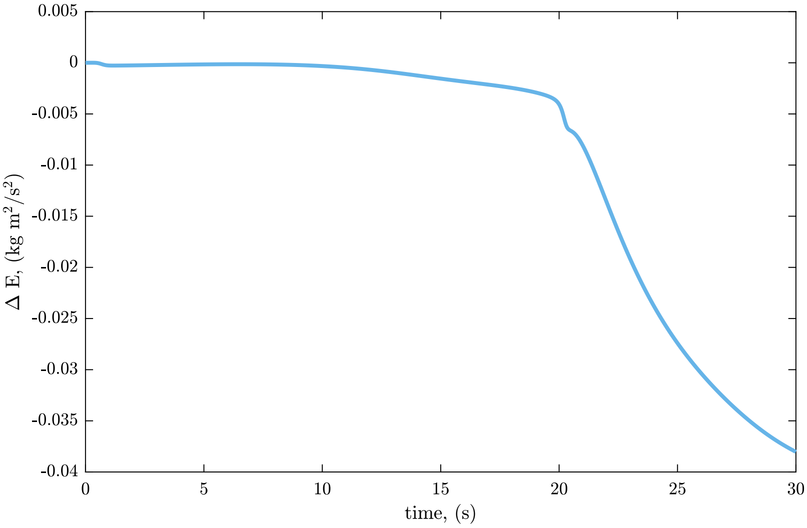

Fig. 16. Change in total system energy of the eight body case.

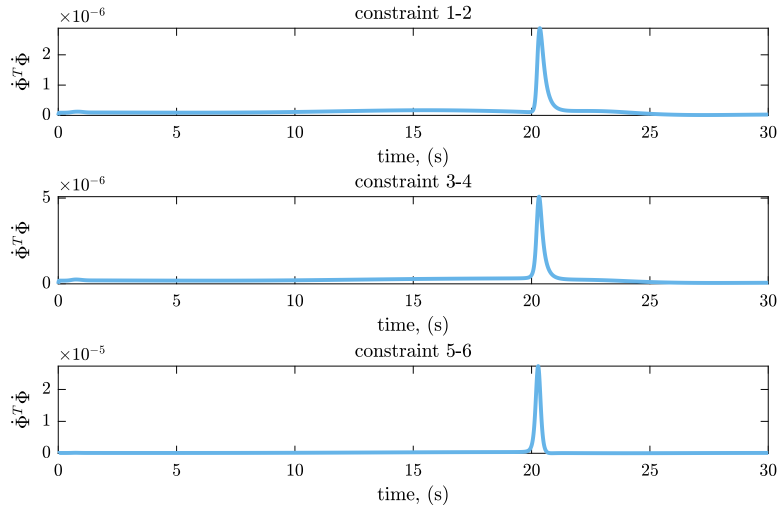

Fig. 17. Change in the velocity form of the constraint equation of the eight body case.

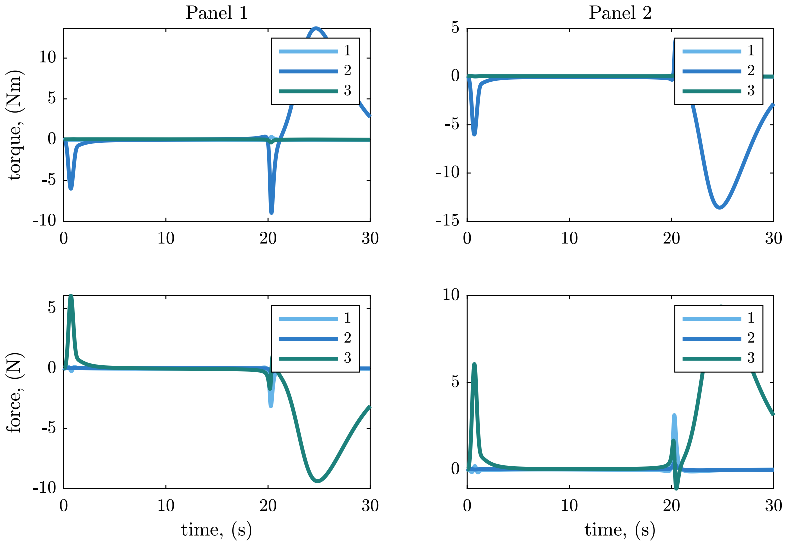

Fig. 18. Internal forces and torques to enforce the constraint equations of hinge 1–2 of the eight body case.

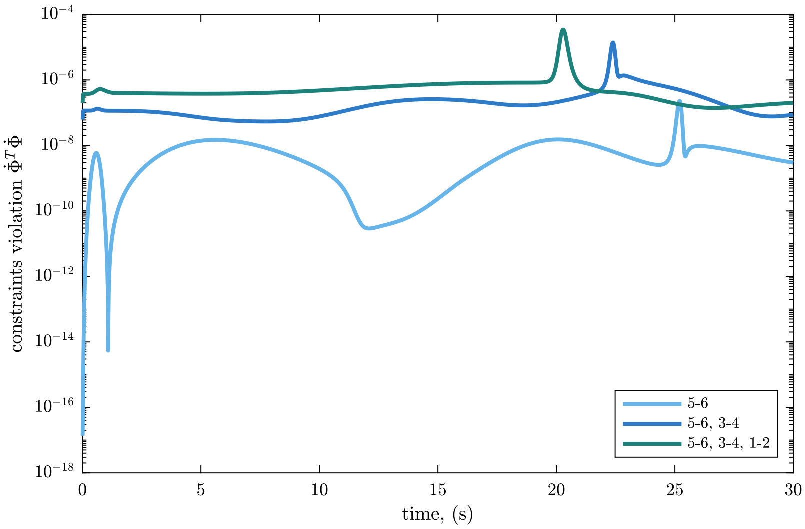

Fig. 19. Constraint violations of cases of multiple constraints for the two chain graph.

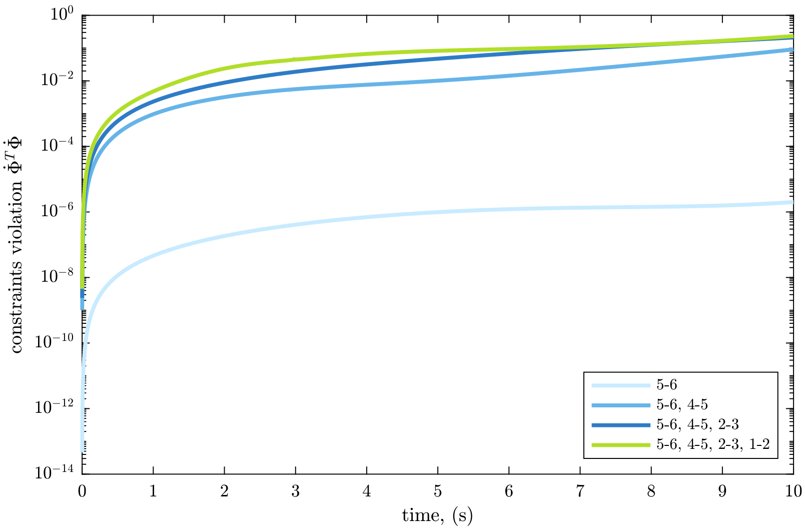

Fig. 20. Constraint violations of multiple constraints for the three chain graph.

body r must unfold first to then allow the z-folded pairs (Panels 1–2, 3–4, and 5–6) to then release near the 20 second mark. Due to the design of the system, the z-folded pairs are expected to unfold with identical states, and this kinematic behavior is verified in the dynamics simulation by the states in Fig. 14. The unfolding process of the z-folded pairs is not representative of a physical structure however, as the model does not include contact. Therefore, the internal force from sets 1–2 unfolding causes the sets 3–4 and 5–6 to fold beyond 180 degrees, as seen from Fig. 14. The host spacecraft states during the deployment, shown in Fig. 15, show a general tumble and small linear perturbations are created. Most notable is the change in the system total energy over the course of the simulation, shown in Fig. 16, which grows significantly around 20 seconds, the point where the deployment transitions to the second stage. This non-conservative energy behavior is due to the Baumgarte constraint stabilization method implemented and is a compromise for using this modeling approach. The constraint violations are plotted as the square of the magnitude of all constraint violations at a node in Fig. 17, and it is noted that the violations have significant peaks at the deployment transition point. These violations are 4 orders of magnitude smaller than the state motions they are applied to, which can be considered acceptable for this demonstration but may need further improvement for future implementation. The magnitude of the constraint violations is suspected to be a function of the number of system constraints, and this relationship is explored in the next section. For a single demonstration of the internal forces applied at the panel body frames due to the constraint, the constraint forces and torques between Panels 1 and 2 is reported in Fig. 18. These are shown to be complex internal behaviors of significant magnitude.

4.3.2. *Constraint violations for multiple constraints*

Comparing the results of Section 4.3.1 to those obtained for a single constraint node in Section 4.2, the magnitude of the constraint violations is suspected to be influenced by the number of constraints in a system. A numerical demonstration for the second test case of the three chain graph is also conducted, and the constraint violation errors are significant enough to invalidate the results of the demonstration. To investigate the constraint violation trends, numerous runs of the two test cases are conducted with different constraints enforced. The results of the two chain graph case are shown in Fig. 19, which shows the squared magnitude of all the constraint violations for instances of 1, 2, and 3 constraints applied on a log scale. There is clearly an increase in violations as the number of constraints are increased, and the violations are shown to stabilize over the simulation with a peak at the point where the deployment stage transitions as discussed previously.

For the second test case where there are instances of a single rigid body subjected to two constraint node pairs, the jump in constraint violation behavior is much more significant. Additionally, the simulation is not stable for long integration times, and tuning the Baumgarte stabilization parameters, $\mathfrak{a}$ and $\mathfrak{b}$, is difficult. In Fig. 20, the square of the magnitude of all constraint violations in the system is shown for just the closure between 5–6 and for closures on body 5 between 5–6 and 4–5, as well as the additional two constraints in the 3×3 grid structure. The cases of more than one constraint do not contain stabilization corrections as the tuning did not yield good results for this case. This study demonstrates that the Baumgarte Stabilization technique is not sufficient when applying multiple constraints to a single rigid body. For a large scale folding structure architecture, there are many instances of a single rigid body subject to two constraint nodes as seen in the pattern graphs of Figs. 3 and 4. Therefore, more advanced constraint stabilization techniques are required for scaling the multi-body dynamics approach to larger folding structure architectures. This point is left for future work in the field.

5. Conclusions and future work

A self-actuated folded deployable spacecraft structure presents a novel modeling challenge due to free-flying spacecraft dynamics coupled with a complexly constrained multibody system. An approach that blends several SOA articulated body-derived robotics dynamics algorithms together is presented to address the multibody folded structure problems. The articulated body forward dynamics algorithm is outlined as the basis for the approach, and adaptations that generalize the ABFD algorithm to the spacecraft folded deployable structure scenario are discussed. The tree augmented approach is developed for any grid formatted spacecraft structure. Origami-folded structure topology is studied and interpreted for dynamics analysis using graph theory, and a 4 body map fold architecture is analyzed for demonstration of the approach. Numerical demonstrations are presented for cases with several folds and multiple closed loops. Origami-inspired folding topologies with large number of bodies are known to have algorithm gains for recursively calculated loop constraints, however constraint violations are a significant concern, as demonstrated on two cases of multiple constraint configurations. The numerically instable deployed configuration of a structure, due to the kinematic singularity of the flattened state, drives this behavior in the model and should be given additional consideration in future work. Future work in the field should focus on developing robust constraint correction and stabilization tools for systems with a large number of constraints as well as multiple constraints applied to a given body in the system. An additional consideration is the influence of the integration scheme in the stability and numerical performance of the model. A variable time step integrator may improve numerical performance around nonlinear deployment behavior and kinematic singularities. Future work should also investigate this.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was conducted under support from the NASA, United States of America Space Technology Research Fellowship, Grant 80NSSC17K0136.

References

[1] A. Jain, Robot and Multibody Dynamics, Springer Science+Business Media, LLC, 2011.

[2] J. Fulton, H. Schaub, Dynamic modeling of folded deployable space structures with flexible hinges, in: 2017 AAS/AIAA Astrodynamics Specialist Conference, Stevenson, WA, 2017.

[3] K. Miura, Folding a plane- scenes from nature technology and art, symmetry, in: Symmetry of structure, interdisciplinary symposium, Budapest, Hungary, 1989.

[4] S. Guest, S. Pellegrino, Inextensional wrapping of flat membranes, in: Proceedings of the First International Seminar on Structural Morphology, 1992.

[5] T.W. Murphey, S. Pellegrino, A novel actuated composite tape-spring for deployable structures, in: 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, 2004.

[6] sarah-marie. belcastro, T.C. Hull, sarah-marie belcastro T. C. Hull Modelling the folding of paper into three dimensions using affine transformations, Linear Algebra Appl. 348 (1) (2002) 273–282, http://dx.doi.org/10.1016/S0024-3795(01)00608-5, URL http://www.sciencedirect.com/science/article/pii/S0024379501006085.

[7] W. Wu, Z. You, Modelling rigid origami with quaternions and dual quaternions, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 466 (2010) 2155–2174, http://dx.doi.org/10.1098/rspa.2009.0625.

[8] M. Schenk, S. Guest, Geometry of miura-folded metamaterials, Proc. Natl. Acad. Sci. USA 110 (2013) 3276–3281, http://dx.doi.org/10.1073/pnas.1217998110.

[9] E. Filipov, K. Liu, T. Tachi, M. Schenk, G. Paulino, Bar and hinge models for scalable analysis of origami, Int. J. Solids Struct. 124 (2017) 26–45, http://dx.doi.org/10.1016/j.ijsolstr.2017.05.028, URL http://www.sciencedirect.com/science/article/pii/S0020768317302408.

[10] C. Liu, S.M. Felton, Transformation dynamics in origami, Phys. Rev. Lett. 121 (2018) 254101, http://dx.doi.org/10.1103/PhysRevLett.121.254101, URL https://link.aps.org/doi/10.1103/PhysRevLett.121.254101.

[11] N.A. Pehrson, S.P. Smith, D.C. Ames, S.P. Magleby, M. Arya, Self-deployable, self-stiffening, and retractable origami-based arrays for spacecraft, in: AIAA SciTech Spacecraft Structures Conference, San Diego, CA, 2019.

[12] M.B. Quadrelli, A. Stoica, M. Ingham, A. Thakur, Flexible electronics-based transformers for extreme environments, in: AIAA SPACE 2015 Conference and Exposition, 2015.

[13] H. Schaub, J.L. Junkins, Analytical Mechanics of Space Systems, third ed., American Institute of Aeronautics and Astronautics, Inc., 1801 Alexander Bell Drive, Reston, Virginia, 20191-4344, 2014.

[14] R. Featherstone, The calculation of robot dynamics using articulated-body inertias, Int. J. Robot. Res. 2 (1) (1983) 13–30.

[15] G. Rodriguez, Recursive forward dynamics for multiple robot arms moving a common task object, 1989.

[16] A. Jain, Unified formulation of dynamics for serial rigid multibody sytems, J. Guid. Control Dyn. 14 (3) (1991) 531–542.

[17] J. Fulton, H. Schaub, Closed-chain forward dynamics modeling of a four-panel folding spacecraft structure, in: International Astronautical Congress, Bremen, Germany, 2018.

[18] J. Fulton, H. Schaub, Forward dynamics algorithm for origami-folded deployable spacecraft structures, in: International Astronautical Congress, Washington, D.C., 2019.

[19] A. Jain, C. Crean, C. Kuo, M. Quadrelli, Efficient constraint modeling for closed-chain dynamics, 2012.

[20] A. Jain, Multibody graph transformations and analysis part ii: Closed-chain constraint embedding, Nonlinear Dynam. 67 (2012) 2153–2170, http://dx.doi.org/10.1007/s11071-011-0136-x.

[21] P. Flores, R. Pereira, M. Machado, E. Seabra, Investigation on the baumgarte stabilization method for dynamic analysis of constrained multibody systems, in: M. Ceccarelli (Ed.), Proc. EUCOMES 08 (2009) 305–312.

[22] J. Baumgarte, Stabilization of constraints and integrals of motion in dynamical systems, Comput. Methods Appl. Mech. Engrg. 1 (1972) 1–16.

[23] J. Fulton, Deployment Dynamics Analysis of Origami-Folded Spacecraft Structures with Elastic Hinges (Ph.D. thesis), University of Colorado at Boulder, 2020.