Linear Fractional Transformation Modeling of Multibody Dynamics Around Parameter-Dependent Equilibrium

Ervan Kassarian, Francesco Sanfedino, Daniel Alazard, Charles-Antoine Chevrier, and Johan Montel

***Abstract*— This brief proposes a new linear fractional transformation (LFT) modeling approach for uncertain linear parameter-varying (LPV) multibody systems with parameter-dependent equilibrium. Traditional multibody approaches, which consist of building the nonlinear model of the whole structure and linearizing it around equilibrium after a numerical trimming, do not allow to isolate parametric variations with the LFT form. Although additional techniques, such as polynomial fitting or symbolic linearization, can provide an LFT model, they may be time-consuming or miss worst case configurations. The proposed approach relies on the trimming and linearization of the equations at the substructure level, before assembly of the multibody structure, which allows to only perform operations that preserve the LFT form throughout the linearization process. Since the physical origin of the parameters is retained, the linearized LFT-LPV model of the structure exactly covers all the plants, in a single parametric model, without introducing conservatism or fitting errors. An application to the LFT-LPV modeling of a robotic arm is proposed; in its nominal configuration, the model obtained with the proposed approach matches the model provided by the software Simscape Multibody, but it is enhanced with parametric variations with the LFT form; a robust LPV synthesis is performed using MATLAB robust control toolbox to illustrate the capacity of the proposed approach for control design.**

***Index Terms*— Linear fractional transformation (LFT) modeling, linear parameter-varying (LPV) system, multibody dynamics, robust control.**

NOMENCLATURE

$(^*\mathbf{u})$	Skew symmetric matrix of vector $\mathbf{u}$, such that $\mathbf{u} \times \mathbf{v} = (^*\mathbf{u})\mathbf{v}$.
$[\mathbf{X}]_{R_x}$	$\mathbf{X}$ (vector or tensor) projected in frame R_x.
$\overline{\mathbf{X}}$	$\mathbf{x}$ (scalar or vector) evaluated at equilibrium.
$\delta\mathbf{X}$	First-order variations of $\mathbf{x}$ around equilibrium.

I. INTRODUCTION

MULTIBODY systems have applications in various fields such as aeronautics, aerospace, or robotics, with numerous modeling and formulation approaches [1]. Even though multibody dynamics are inherently nonlinear, it is often useful to linearize them at equilibrium to study the stability, perform modal analysis, or apply classical linear control methods. In practical engineering problems, many parameter uncertainties impact the dynamics of the system and must be taken into account for robust analysis and control. When working on the uncertain linear model, a representation of the uncertainties with a bounded and unknown operator Δ, based on the linear fractional transformation (LFT), enables powerful tools to perform worst case robust analysis and control such as μ analysis or $\mathcal{H}_\infty$ synthesis [2]. Furthermore, the nonlinear system can often be approximated by a linear parameter-varying (LPV) model around a slowly varying equilibrium, where the varying or nonlinear terms are also represented in the operator Δ of the LFT. Finally, some mechanical parameters, such as masses of some elements, may be considered as decision variables and included in the LFT to be optimized simultaneously with the controller in multidisciplinary co-design approaches. In this brief, the uncertain, varying, and decision parameters are referred to as the parameters of interest. Classical multibody approaches, consisting of building the nonlinear model of the structure by assembly of the individual models and then linearizing this nonlinear model around equilibrium, are unable to directly provide the LFT-LPV model. This is due to the trim conditions depending on the parameters of interest: for example, consider the small angles' variations in a pendulum with an uncertain mass—the gravity introduces a stiffness which depends on the uncertain mass, and a numerical trimming preceding the linearization will only capture a single parametric configuration of this stiffness rather than a parameterized LFT model taking into account the parametric uncertainty on the mass. For a more general class of systems, the use of symbolic linearization was proposed in [3] and [4] to overcome this issue, but it is computationally costly for complex systems, especially when dealing with many parameters or high-order dynamics. Consequently, the most common practice for systems with parameter-dependent trim conditions is to perform numerical linearizations around a grid of equilibrium points corresponding to particular values of the parameters and to generate a model covering all the linear time-invariant (LTI) models of the grid using multivariable polynomial fitting techniques [5], [6]. However, this procedure may introduce conservatism or miss worst case configurations and may be time-consuming when there are many parameters or when a fine grid is required.

For the modeling of large space structures such as satellites with flexible solar panels in micro-gravity conditions, a general framework was introduced in [7] and implemented in a generic toolbox named satellite dynamics toolbox (SDT) [8], to build

This work was supported in part by Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO and in part by Centre National d'Etudes Spatiales (CNES) (French Space Agency–Grant 51/18660). Recommended by Associate Editor M. Ariola. *(Corresponding author: Ervan Kassarian.)*

Ervan Kassarian, Francesco Sanfedino, and Daniel Alazard are with ISAE-SUPAERO, 31055 Toulouse, France (e-mail: ervan.kassarian@isae.fr; francesco.sanfedino@isae.fr; daniel.alazard@isae.fr).

Charles-Antoine Chevrier and Johan Montel are with CNES, 31055 Toulouse, France (e-mail: charlesantoine.chevrier@cnes.fr; johan.montel@cnes.fr).

Digital Object Identifier 10.1109/TCST.2022.3167610

linear models of flexible multibody systems. Based on the Newton–Euler equations, this tool allows to build the dynamic model of the whole structure by assembling the individual models of each substructure based on the two-input two-output (TITOP) formalism [9]. Some assets of this approach include the compliance with various substructure models and boundary conditions, and support of the interfacing with finite element software when the model includes complex substructures [10]. The resulting model is provided under the form of a block diagram with minimal number of states, and the parameters can be isolated to obtain a minimal LFT model, allowing robust control [11] or integrated control/structure co-design with the $\mathcal{H}_\infty$ synthesis [12].

In this brief, the framework from [7] to [12] is extended to the modeling of multibody systems undergoing variations around a uniformly accelerated motion, e.g., for systems subject to gravity (robotic arms, aircrafts, civil machinery, stratospheric balloons…) or space systems during a thrust phase (launchers, spacecrafts). It was motivated by the need for robust control for stratospheric balloons, which are complex multibody systems subject to gravity with uncertain masses [13] that cannot be modeled with current multibody software due to the parameter-dependent trim conditions. Rather than linearizing the nonlinear model of the multibody structure, the proposed approach linearizes each individual substructure and kinematic joint and assembles them to build the LFT model of the structure, after an analytical computation of the parameter-dependent equilibrium. It allows to only perform operations that preserve the LFT form throughout the linearization process. The LFT model regroups all the parametric configurations in one single model, enabling modern analysis and control tools like μ analysis or $\mathcal{H}_\infty$ synthesis, and is obtained without resorting to symbolic trimming of the nonlinear model or polynomial fitting of a set of LTI plants; in particular, the LFT model exactly covers all the plants within the specified bounds without introducing conservatism or fitting error. Since the linearization procedure only relies on basic block diagram manipulations, the LFT model is obtained in a reasonable amount of time. From the control engineer's point of view, the proposed approach can be implemented in MATLAB-Simulink to build complex multibody structures by interconnecting the individual bodies. Targeted engineering applications include modeling of uncertain LPV multibody systems and lumped-parameter modeling of uncertain flexible systems, for the purpose of robust control, gain scheduling, vibrations control, or integrated control/structure co-design, along with control design tools such as MATLAB robust control toolbox. To the authors' knowledge, this approach is the first contribution addressing parametric model linearization around parameter-dependent equilibrium in the general context of uncertain multibody systems.

The dynamics of rigid bodies are modeled with the Newton–Euler equations in Section II, and the equations of the revolute joint are presented in Section III. Section IV discusses the equilibrium and linearization of the individual models of rigid body and revolute joint and the compatibility with LFT formalism. The assembly, trim, and linearization algorithm, allowing to keep the LFT dependency of the model on the parameters of interest during the linearization, is detailed in Section V. Finally, Section VI presents an application to LFT modeling of an LPV robotic arm; the model is validated with a comparison to Simscape Multibody, and an LPV control design is performed to illustrate the capacity of the proposed approach for control design.

II. Rigid Body Dynamics

A. Description of the Motion

Definition 2.1: Uniformly accelerated reference frame $\mathcal{R}$ Let $\mathcal{R} = (O, \mathbf{x}, \mathbf{y}, \mathbf{z})$ be a reference frame in uniform acceleration, represented by 3×1 vector $\mathbf{a}$, with regard to an inertial reference frame $\mathcal{R}_i$.

In this brief, the motion is described in the reference frame $\mathcal{R}$. This equilibrium condition can represent a gravity field or an acceleration during a thrust phase for a space system.

Definition 2.2: Motion in the reference frame $\mathcal{R}$ Let us define the following vectors.

1) $\mathbf{x}_P^{\mathcal{B}} = \begin{bmatrix} OP \\ \boldsymbol{\theta}^{\mathcal{B}} \end{bmatrix}$ the 6×1 pose vector of body $\mathcal{B}$ at point P, with OP the 3×1 position vector of P and $\boldsymbol{\theta}^{\mathcal{B}}$ the 3×1 vector of Euler angles of $\mathcal{B}$ with regard to $\mathcal{R}$.
2) $\mathbf{x}_P'^{\mathcal{B}} = \begin{bmatrix} \mathbf{v}_P^{\mathcal{B}} \\ \boldsymbol{\omega}^{\mathcal{B}} \end{bmatrix}$ the 6×1 dual velocity vector of body $\mathcal{B}$ at point P, with $\mathbf{v}_P^{\mathcal{B}} = ((d\overrightarrow{OP})/dt)|_{\mathcal{R}}$ and $\boldsymbol{\omega}^{\mathcal{B}}$ the angular velocity of $\mathcal{B}$ with regard to $\mathcal{R}$.
3) $\mathbf{x}_P''^{\mathcal{B}} = ((d\mathbf{x}_P'^{\mathcal{B}})/dt)|_{\mathcal{R}} = \begin{bmatrix} \dot{\mathbf{v}}_P^{\mathcal{B}} \\ \dot{\boldsymbol{\omega}}^{\mathcal{B}} \end{bmatrix}$ the 6×1 dual acceleration vector of body $\mathcal{B}$ at point P.
4) $\mathbf{m}_P^{\mathcal{B}} = [\mathbf{a}^T\ ,\ (\mathbf{x}_P''^{\mathcal{B}})^T\ ,\ (\mathbf{x}_P'^{\mathcal{B}})^T\ ,\ (\mathbf{x}_P^{\mathcal{B}})^T]^T$ is defined as the motion vector of body $\mathcal{B}$ at point P.

Noting $\mathbf{a}_6 = \begin{bmatrix} \mathbf{a} \\ \mathbf{0}_{3\times 1} \end{bmatrix}$, the linear and angular accelerations of body $\mathcal{B}$ at point P with respect to $\mathcal{R}_i$ are

$$\begin{bmatrix} \mathbf{a}_P^{\mathcal{B}}\big|_{\mathcal{R}_i} \\ \dot{\boldsymbol{\omega}}^{\mathcal{B}}\big|_{\mathcal{R}_i} \end{bmatrix} = \mathbf{x}_P''^{\mathcal{B}} + \mathbf{a}_6. \tag{1}$$

B. Newton–Euler Equations for Rigid Bodies

Let us consider a body $\mathcal{B}$ of mass $m^{\mathcal{B}}$ and matrix of inertia $\mathbf{J}_B^{\mathcal{B}}$ at center of gravity B. The Newton–Euler equations read at B

$$\underbrace{\begin{bmatrix} \mathbf{F}^{\mathcal{B}} \\ \mathbf{T}_B^{\mathcal{B}} \end{bmatrix}}_{\mathbf{W}_B^{\mathcal{B}}} = \underbrace{\begin{bmatrix} m^{\mathcal{B}}\mathbf{I}_3 & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{J}_B^{\mathcal{B}} \end{bmatrix}}_{\mathbf{D}_B^{\mathcal{B}}} \begin{bmatrix} \mathbf{a}_B^{\mathcal{B}}\big|_{\mathcal{R}_i} \\ \dot{\boldsymbol{\omega}}^{\mathcal{B}}\big|_{\mathcal{R}_i} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times 1} \\ ({}^*\boldsymbol{\omega}^{\mathcal{B}})\mathbf{J}_B^{\mathcal{B}}\boldsymbol{\omega}^{\mathcal{B}} \end{bmatrix} \tag{2}$$

where $\mathbf{W}_B^{\mathcal{B}} = \begin{bmatrix} \mathbf{F}^{\mathcal{B}} \\ \mathbf{T}_B^{\mathcal{B}} \end{bmatrix}$ is the 6×1 wrench vector (force $\mathbf{F}^{\mathcal{B}}$ and torque $\mathbf{T}_B^{\mathcal{B}}$) applied to the body $\mathcal{B}$ at point B. Definition 2.3 and Property 2.4 were introduced in [7] to transport equation (2) to any other point P of body $\mathcal{B}$.

Definition 2.3: Kinematic model [7] The 6×6 tensor $\boldsymbol{\tau}_{PC} = \begin{bmatrix} \mathbf{I}_3 & ({}^*\overrightarrow{PC}) \\ \mathbf{0}_{3\times 3} & \mathbf{I}_3 \end{bmatrix}$ is defined as the kinematic model between two points P and C.

Property 2.4: Transport of the vectors [7]:

1) Dual velocity vector: $\mathbf{x}_P'^{\mathcal{B}} = \boldsymbol{\tau}_{PC}\mathbf{x}_C'^{\mathcal{B}}$;

2) Dual acceleration vector:

$$\mathbf{x}''^{\mathcal{B}}_P = \boldsymbol{\tau}_{PC}\mathbf{x}''^{\mathcal{B}}_C + \begin{bmatrix} \left({}^*\boldsymbol{\omega}^{\mathcal{B}}\right)\left({}^*\overrightarrow{PC}\right)\boldsymbol{\omega}^{\mathcal{B}} \\ \mathbf{0}_{3\times 1} \end{bmatrix};$$

3) Wrench vector: $\mathbf{W}^{\mathcal{B}}_C = \boldsymbol{\tau}^T_{PC}\mathbf{W}^{\mathcal{B}}_P$;
4) Inverse kinematic model: $\boldsymbol{\tau}^{-1}_{PC} = \boldsymbol{\tau}_{CP}$;
5) Transitivity: $\boldsymbol{\tau}_{PC}\boldsymbol{\tau}_{CP'} = \boldsymbol{\tau}_{PP'}$.

Using Property 2.4 to transport the vectors from point B to another point P of body $\mathcal{B}$, and since $\boldsymbol{\tau}_{BP}\mathbf{a}_6 = \mathbf{a}_6$, (2) is transported to P

$$\mathbf{W}^{\mathcal{B}}_P = \underbrace{\boldsymbol{\tau}^T_{BP}\mathbf{D}^{\mathcal{B}}_B\boldsymbol{\tau}_{BP}}_{\mathbf{D}^{\mathcal{B}}_P}\left(\mathbf{x}''^{\mathcal{B}}_P + \mathbf{a}_6\right) + \underbrace{\begin{bmatrix} m^{\mathcal{B}}\left({}^*\boldsymbol{\omega}^{\mathcal{B}}\right)\left({}^*\overrightarrow{BP}\right)\boldsymbol{\omega}^{\mathcal{B}} \\ \left({}^*\boldsymbol{\omega}^{\mathcal{B}}\right)\left(\mathbf{J}^{\mathcal{B}}_B - m^{\mathcal{B}}\left({}^*\overrightarrow{BP}\right)^2\right)\boldsymbol{\omega}^{\mathcal{B}} \end{bmatrix}}_{\mathrm{NL}\left(P,\boldsymbol{\omega}^{\mathcal{B}}\right)} \quad (3)$$

where $\mathrm{NL}(P,\boldsymbol{\omega}^{\mathcal{B}})$ regroups the nonlinear terms, and $\mathbf{D}^{\mathcal{B}}_P$ is defined as the direct dynamics model of body $\mathcal{B}$ at point P.

C. Projection in the Body's Frame

To describe each body independently of the others, (3) is projected in the reference frame $\mathcal{R}_b$ attached to $\mathcal{B}$

$$\left[\mathbf{W}^{\mathcal{B}}_P\right]_{\mathcal{R}_b} = \left[\mathbf{D}^{\mathcal{B}}_P\right]_{\mathcal{R}_b}\left(\left[\mathbf{x}''^{\mathcal{B}}_P\right]_{\mathcal{R}_b} + [\mathbf{a}_6]_{\mathcal{R}_b}\right) + \left[\mathrm{NL}\left(P,\boldsymbol{\omega}^{\mathcal{B}}\right)\right]_{\mathcal{R}_b}. \quad (4)$$

The kinematic and direct dynamic models are conveniently written in the body's frame. The inertial uniform acceleration vector $\mathbf{a}$ is defined in the inertial frame $\mathcal{R}_i$, or equivalently, in frame $\mathcal{R}$: $[\mathbf{a}]_{\mathcal{R}_i} = [\mathbf{a}]_{\mathcal{R}}$. With the notations of Definition 2.5, its projection in $\mathcal{R}_b$ reads

$$[\mathbf{a}]_{\mathcal{R}_b} = \mathbf{P}^T_{b/i}\left(\boldsymbol{\theta}^{\mathcal{B}}\right)[\mathbf{a}]_{\mathcal{R}_i}. \quad (5)$$

Definition 2.5 (Direction Cosine Matrix): The direction cosine matrix (DCM) between the body's frame $\mathcal{R}_b = (O, \mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b)$ and the frame $\mathcal{R}_i$, containing the coordinates of vectors $\mathbf{x}_b$, $\mathbf{y}_b$, $\mathbf{z}_b$ expressed in frame $\mathcal{R}_i$, is noted $\mathbf{P}_{b/i}(\boldsymbol{\theta}^{\mathcal{B}})$. The inverse function, which converts a DCM $\mathbf{P}_{b/i}$ into the equivalent Euler angles, is noted $\Theta(\mathbf{P}_{b/i}(\boldsymbol{\theta}^{\mathcal{B}}))$.

Definition 2.6 (From Euler Angles Rates to Angular Velocity): The relationship between the body frame angular velocity vector and the rate of change in Euler angles is

$$\left[\boldsymbol{\omega}^{\mathcal{B}}\right]_{\mathcal{R}_b} = \boldsymbol{\Gamma}\left(\boldsymbol{\theta}^{\mathcal{B}}\right)\dot{\boldsymbol{\theta}}^{\mathcal{B}} \quad (6)$$

where $\boldsymbol{\Gamma}(\boldsymbol{\theta}^{\mathcal{B}})$ depends on the chosen Euler sequence and expresses the relationship between the angular velocity vector and the rate of change in Euler angles [14].

III. Connection With a Revolute Joint

In this section, we consider two bodies $\mathcal{A}$ and $\mathcal{B}$ interconnected with a revolute joint (one degree of freedom in rotation).

A. Change in Frame

Since the equations describing the motion of $\mathcal{B}$ (respectively $\mathcal{A}$) are projected in the reference frame $\mathcal{R}_b$ (respectively $\mathcal{R}_a$), the change in frame operation is necessary to write the interconnection of $\mathcal{A}$ and $\mathcal{B}$.

Property 3.1 (Change of Frame): Given the DCM $\mathbf{P}_{a/b}$ between two frames $\mathcal{R}_a$ and $\mathcal{R}_b$ according to Definition 2.5, let us define $\mathbf{P}^{\times 2}_{a/b} = \mathrm{diag}(\mathbf{P}_{a/b}, \mathbf{P}_{a/b})$. Then:

1) For X, a dual velocity, acceleration, or wrench vector: $[\mathbf{X}]_{\mathcal{R}_b} = \mathbf{P}^{\times 2}_{a/b}[\mathbf{X}]_{\mathcal{R}_a}$;
2) Direct dynamics model: $\mathbf{P}^{\times 2}_{a/b}[\mathbf{D}^{\mathcal{A}}_P]_{\mathcal{R}_a}{\mathbf{P}^{\times 2}_{a/b}}^T = [\mathbf{D}^{\mathcal{A}}_P]_{\mathcal{R}_b}$;
3) $\mathbf{P}_{b/a} = \mathbf{P}^{-1}_{a/b} = \mathbf{P}^T_{a/b}$.

B. Model of the Revolute Joint

Let θ, $\dot{\theta}$, and $\ddot{\theta}$ be the angular configuration, rate, and acceleration, respectively, inside the revolute joint between bodies $\mathcal{B}$ and $\mathcal{A}$ at the connection point P, $\mathbf{r}$ be the vector of unit norm aligned with the joint's axis, and T_r be the driving torque along $\mathbf{r}$. The revolute joint $\mathcal{J}$ is modeled as a body with two ports (it is connected to $\mathcal{A}$ and $\mathcal{B}$), to which are added an input $\ddot{\theta}$ and an output T_r. It is assumed that $\mathcal{J}$ is a mass-less body attached to the body $\mathcal{A}$, with a matrix of inertia $\mathbf{J}^{\mathcal{J}}_P = J^{\mathcal{J}}\mathbf{r}\mathbf{r}^T$. From (2), the dynamic model of $\mathcal{J}$ reads

$$\begin{aligned}\mathbf{W}_{\mathcal{B}/\mathcal{J},P} + \mathbf{W}_{\mathcal{A}/\mathcal{J},P} &= \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{J}^{\mathcal{J}}_P \end{bmatrix}\begin{bmatrix} \dot{\mathbf{v}}^{\mathcal{A}}_P + \mathbf{a} \\ \dot{\boldsymbol{\omega}}^{\mathcal{A}} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \left({}^*\boldsymbol{\omega}^{\mathcal{A}}\right)\mathbf{J}^{\mathcal{J}}\boldsymbol{\omega}^{\mathcal{A}} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \mathbf{J}^{\mathcal{J}}_P\left(\dot{\boldsymbol{\omega}}^{\mathcal{B}} + \ddot{\theta}\mathbf{r}\right) + \left({}^*\boldsymbol{\omega}^{\mathcal{A}}\right)\mathbf{J}^{\mathcal{J}}_P\boldsymbol{\omega}^{\mathcal{A}} \end{bmatrix}. \end{aligned} \quad (7)$$

The driving torque T_r is the projection of the torque $\mathbf{T}_{B/\mathcal{J},P}$ applied by $\mathcal{B}$ on $\mathcal{J}$ at P along $\mathbf{r}$

$$\begin{aligned} T_r &= \mathbf{r}^T_6\mathbf{W}_{B/\mathcal{J},P} \\ &= J^{\mathcal{J}}\left(\mathbf{r}^T\dot{\boldsymbol{\omega}}^{\mathcal{B}} + \ddot{\theta}\right) - \mathbf{r}^T_6\mathbf{W}_{\mathcal{A}/\mathcal{J},P} + \underbrace{\mathbf{r}^T\left({}^*\boldsymbol{\omega}^{\mathcal{A}}\right)\mathbf{J}^{\mathcal{J}}_P\boldsymbol{\omega}^{\mathcal{A}}}_{=0} \end{aligned} \quad (8)$$

where $\mathbf{r}_6 = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \mathbf{r} \end{bmatrix}$. In most applications, it is preferred to invert the channel from $(\theta, \dot{\theta}, \ddot{\theta})$ to T_r to take into account a driving mechanism actuating the revolute joint

$$\ddot{\theta} = \frac{1}{J^{\mathcal{J}}}\left(T_r + \mathbf{r}^T_6\mathbf{W}_{\mathcal{A}/\mathcal{J},P}\right) - \mathbf{r}^T\dot{\boldsymbol{\omega}}^{\mathcal{B}}. \quad (9)$$

The motion vector, projected in each body's frame, is transformed through the revolute joint as follows.

Property 3.2: Transformation of the motion vector through a revolute joint between two bodies: The motion vector at point P can be expressed from body $\mathcal{B}$ to body $\mathcal{A}$, connected at point P with a revolute joint

$$\left[\mathbf{m}^{\mathcal{A}}_P\right]_{\mathcal{R}_a} = \begin{bmatrix} [\mathbf{a}]_{\mathcal{R}_a} \\ \left[\mathbf{x}''^{\mathcal{A}}_P\right]_{\mathcal{R}_a} \\ \left[\mathbf{x}'^{\mathcal{A}}_P\right]_{\mathcal{R}_a} \\ \left[\mathbf{x}^{\mathcal{A}}_P\right]_{\mathcal{R}_a} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{P}_{b/a}(\theta)[\mathbf{a}]_{\mathcal{R}_b} \\ \mathbf{P}^{\times 2}_{b/a}(\theta)\left[\mathbf{x}''^{\mathcal{B}}_P\right]_{\mathcal{R}_a} + \ddot{\theta}[\mathbf{r}_6]_{\mathcal{R}_a} \\ \mathbf{P}^{\times 2}_{b/a}(\theta)\left[\mathbf{x}'^{\mathcal{B}}_P\right]_{\mathcal{R}_a} + \dot{\theta}[\mathbf{r}_6]_{\mathcal{R}_a} \\ \begin{bmatrix} \mathbf{P}_{b/a}(\theta)\left[\overrightarrow{OP}\right]_{\mathcal{R}_a} \\ \Theta^{\mathcal{J}}_{b/a}\left(\boldsymbol{\theta}^{\mathcal{B}}, \theta\right) \end{bmatrix} \end{bmatrix}$$

where $\boldsymbol{\theta}^{\mathcal{A}} = \Theta^{\mathcal{J}}_{b/a}(\boldsymbol{\theta}^{\mathcal{B}}, \theta)$ is defined as $\Theta^{\mathcal{J}}_{b/a}(\boldsymbol{\theta}^{\mathcal{B}}, \theta) = \Theta(\mathbf{P}_{b/i}(\boldsymbol{\theta}^{\mathcal{B}})\mathbf{P}_{b/a}(\theta))$.

IV. Linearization of the Individual Models

In this section, the equations describing the equilibrium and the linear variations around the equilibrium are derived individually for each model of rigid body and revolute joint obtained in Sections II and III. This way, the parametric dependencies are analytically derived on simple models; it is shown why this step is necessary to capture the LFT dependency on certain parameters of interest.

A. Equilibrium

The system is said to be at equilibrium when it has no motion in the reference frame $\mathcal{R}$. For a body $\mathcal{B}$ and a point P, it corresponds to: $\{\mathbf{x}^{\mathcal{B}}_P = \overline{\mathbf{x}}^{\mathcal{B}}_P,\ \mathbf{x}'^{\mathcal{B}}_P = \mathbf{0},\ \mathbf{x}''^{\mathcal{B}}_P = \mathbf{0}\}$. For a revolute joint $\mathcal{J}$, it corresponds to $\{\theta = \bar{\theta},\ \dot{\theta} = 0,\ \ddot{\theta} = 0\}$. The Euler angles at equilibrium are noted $\bar{\boldsymbol{\theta}}^{\mathcal{B}}$. Around the equilibrium, the vectors defined in Section II-A, projected in $\mathcal{R}_b$, verify to the first-order

$$\begin{cases} \delta\left[\mathbf{x}''^{\mathcal{B}}_P\right]_{\mathcal{R}_b} = \dfrac{d\left(\delta\left[\mathbf{x}'^{\mathcal{B}}_P\right]_{\mathcal{R}_b}\right)}{dt} \\ \delta\left[\mathbf{x}'^{\mathcal{B}}_P\right]_{\mathcal{R}_b} = \mathrm{diag}\left(\mathbf{I}_3, \boldsymbol{\Gamma}\left(\bar{\boldsymbol{\theta}}^{\mathcal{B}}\right)\right)\dfrac{d\left(\delta\left[\mathbf{x}^{\mathcal{B}}_P\right]_{\mathcal{R}_b}\right)}{dt} \end{cases} \tag{10}$$

and the linearized motion vector projected in $\mathcal{R}_b$ is

$$\delta\mathbf{m}^{\mathcal{B}}_P = \left[\delta[\mathbf{a}]^T_{\mathcal{R}_b}, \delta\left[\mathbf{x}''^{\mathcal{B}}_P\right]^T_{\mathcal{R}_b}, \delta\left[\mathbf{x}'^{\mathcal{B}}_P\right]^T_{\mathcal{R}_b}, \delta\left[\mathbf{x}^{\mathcal{B}}_P\right]^T_{\mathcal{R}_b}\right]^T. \tag{11}$$

B. Linearized Model of the Rigid Body

Equation (4) is evaluated at equilibrium

$$\left[\overline{\mathbf{W}}^{\mathcal{B}}_P\right]_{\mathcal{R}_b} = \left[\mathbf{D}^{\mathcal{B}}_P\right]_{\mathcal{R}_b}[\overline{\mathbf{a}}_6]_{\mathcal{R}_b} \tag{12}$$

with

$$[\overline{\mathbf{a}}_6]_{\mathcal{R}_b} = \begin{bmatrix} \mathbf{P}^T_{b/i}\left(\bar{\boldsymbol{\theta}}^{\mathcal{B}}\right)[\mathbf{a}]_{R_i} \\ \mathbf{0}_{3\times 1} \end{bmatrix}. \tag{13}$$

Equation (12) shows that the wrench applied to $\mathcal{B}$ at equilibrium depends on the DCM $\mathbf{P}_{b/i}(\bar{\boldsymbol{\theta}}^{\mathcal{B}})$ and on the direct dynamics model $[\mathbf{D}^{\mathcal{B}}_P]_{\mathcal{R}_b}$ of body $\mathcal{B}$, which can be an LFT of the following parameters: mass, matrix of inertia, and position of the center of gravity B relatively to point P. This observation induces that the internal wrenches of the multibody system may be LFTs of these parameters, which will be of importance when linearizing the model of revolute joint in Section IV-C.

Equation (4) is linearized around the equilibrium

$$\delta\left[\mathbf{W}^{\mathcal{B}}_P\right]_{\mathcal{R}_b} = \left[\mathbf{D}^{\mathcal{B}}_P\right]_{\mathcal{R}_b}\left(\delta\left[\mathbf{x}''^{\mathcal{B}}_P\right]_{\mathcal{R}_b} + \delta[\mathbf{a}_6]_{\mathcal{R}_b}\right). \tag{14}$$

Remark: including the acceleration vector $\mathbf{a}$ in the motion vector $\mathbf{m}^{\mathcal{B}}_P$ allows (14) to be linear in the linearized motion vector $\delta[\mathbf{m}^{\mathcal{B}}_P]_{\mathcal{R}_b}$, and thus to be compliant with LFT formalism. This is possible because $\delta[\mathbf{a}]_{\mathcal{R}_b}$ can be propagated through the revolute joints (see the linearization of Property 3.2, in Section IV-C). If it was not the case, we should instead write $\delta[\mathbf{a}]_{\mathcal{R}_b}$ as

$$\delta[\mathbf{a}]_{\mathcal{R}_b} = \left.\frac{d[\mathbf{a}]_{\mathcal{R}_b}}{d\boldsymbol{\theta}^{\mathcal{B}}}\right|_{\mathrm{eq}} \delta\boldsymbol{\theta}^{\mathcal{B}} \tag{15}$$

but the matrix $((d[\mathbf{a}]_{\mathcal{R}_b})/(d\boldsymbol{\theta}^{\mathcal{B}}))|_{\mathrm{eq}}$ cannot generally be obtained as an LFT because it depends on Euler angles (see Appendix).

The transport of the motion vector (following Property 2.4) can be linearized around the equilibrium:

$$\begin{cases} \delta\left[\mathbf{m}^{\mathcal{B}}_P\right]_{\mathcal{R}_b} = [\boldsymbol{\Upsilon}_{PC}]_{\mathcal{R}_b}\delta\left[\mathbf{m}^{\mathcal{B}}_C\right]_{\mathcal{R}_b} \\ [\boldsymbol{\Upsilon}_{PC}]_{\mathcal{R}_b} = \mathrm{diag}\left(\mathbf{I}_3, [\boldsymbol{\tau}_{PC}]_{\mathcal{R}_b}, [\boldsymbol{\tau}_{PC}]_{\mathcal{R}_b}, \mathbf{I}_6\right). \end{cases} \tag{16}$$

Finally, consider a body $\mathcal{B}$ where the motion is imposed at parent port P, and external wrenches $\mathbf{W}_{./\mathcal{B},C_i}$ are applied at N child ports C_i. From (14) and (16), the linearized inverse dynamics LFT model is represented by the block diagram in Fig. 1(a). Using additionally (10), the linearized 12th-order forward dynamics LFT model is represented by the block diagram in Fig. 1(b) for a body with N ports C_i where only external wrenches are applied (no imposed motion). The green and blue blocks represent the nominal models and the $\boldsymbol{\Delta}$ operators, respectively.

Remark: a multibody system has a base which is either the ground (imposed motion) or a body with forward dynamics (no imposed motion, the equilibrium is determined by the wrenches). In the latter case, the orientation of the base at equilibrium is explicitly defined. Then, the matrix $((d[\mathbf{a}]_{\mathcal{R}_b})/(d\boldsymbol{\theta}^{\mathcal{B}}))|_{\mathrm{eq}}$ can be obtained as an LFT of the Euler angles. However, for inverse dynamics, the Euler angles are propagated from the base to the body through the joints, and since this transformation is not compliant with the LFT (see the discussion in Appendix), it is necessary to use the acceleration propagated with the motion vector.

C. Linearized Model of the Revolute Joint

The linearization of the revolute joint must take into account the dependency of the DCM $\mathbf{P}_{a/b}$ on the variable θ: if $\mathbf{X}$ is a vector such that $[\mathbf{X}]_{\mathcal{R}_a} = \mathbf{P}_{b/a}(\theta)[\mathbf{X}]_{\mathcal{R}_b}$

$$\delta[\mathbf{X}]_{\mathcal{R}_a} = \mathbf{P}_{b/a}\left(\bar{\theta}\right)\delta[\mathbf{X}]_{\mathcal{R}_b} + \underbrace{\left.\frac{d\mathbf{P}_{b/a}}{d\theta}\right|_{\mathrm{eq}}}_{=(\mathbf{r}^*)\mathbf{P}_{b/a}\left(\bar{\theta}\right)} \delta\theta\left[\overline{\mathbf{X}}\right]_{\mathcal{R}_b}. \tag{17}$$

Remark: the DCM $\mathbf{P}_{b/a}(\bar{\theta})$ can be expressed as an LFT of the parameter $t = \tan(\bar{\theta}/2)$ (respectively $t = \tan(\bar{\theta}/4)$) with 2 (respectively 4) occurrences (cf. [15, p. 191 to 195]).

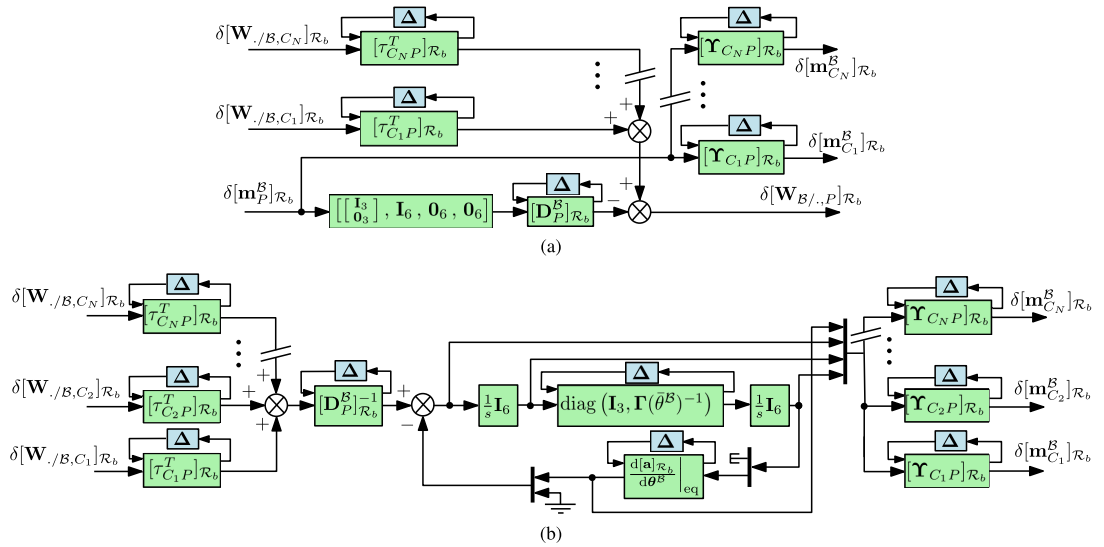

Fig. 1. Linearized model of a rigid body $\mathcal{B}$: (a) inverse dynamics and (b) forward dynamics.

The projections of equation (7) in the frame $\mathcal{R}_b$ and of equation (9) along $\mathbf{r}$ read

$$\begin{cases} \left[\mathbf{W}_{\mathcal{B}/\mathcal{J},P}\right]_{\mathcal{R}_b} + \mathbf{P}^{\times 2}_{a/b}(\theta)\left[\mathbf{W}_{\mathcal{A}/\mathcal{J},P}\right]_{\mathcal{R}_a} \\ \quad = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \left[\mathbf{J}^{\mathcal{J}}_P\right]_{\mathcal{R}_b}\left(\left[\dot{\boldsymbol{\omega}}^{\mathcal{B}}\right]_{\mathcal{R}_b} + \ddot{\theta}[\mathbf{r}]_{\mathcal{R}_b}\right) + \left[({}^{*}\boldsymbol{\omega}^{\mathcal{A}})\mathbf{J}^{\mathcal{J}}_P\boldsymbol{\omega}^{\mathcal{A}}\right]_{\mathcal{R}_b} \end{bmatrix} \\ \ddot{\theta} = \dfrac{1}{J^{\mathcal{J}}}\left(T_r + \left[\mathbf{r}^T_6\right]_{\mathcal{R}_a}\left[\mathbf{W}_{\mathcal{A}/\mathcal{J},P}\right]_{\mathcal{R}_a}\right) - \left[\mathbf{r}^T\right]_{\mathcal{R}_b}\left[\dot{\boldsymbol{\omega}}^{\mathcal{B}}\right]_{\mathcal{R}_b}. \end{cases} \tag{18}$$

It can be noted that $[\mathbf{r}]_{\mathcal{R}_b}$ and $[\mathbf{J}^{\mathcal{J}}_P]_{\mathcal{R}_b}$ are independent of θ. Indeed, noting $\mathbf{R}(\theta)$ the rotation matrix around $\mathbf{r}$, the DCM reads $\mathbf{P}_{b/a}(\theta) = \mathbf{R}(\theta)\mathbf{P}_{b/a}(0)$. Then

$$[\mathbf{r}]_{\mathcal{R}_b} = \mathbf{P}^T_{b/a}(0)\underbrace{\mathbf{R}^T(\theta)[\mathbf{r}]_{\mathcal{R}_a}}_{=[\mathbf{r}]_{\mathcal{R}_a}} \tag{19}$$

and

$$\begin{aligned} \left[\mathbf{J}^{\mathcal{J}}_P\right]_{\mathcal{R}_b} &= \mathbf{P}^T_{b/a}(\theta)\left[\mathbf{J}^{\mathcal{J}}_P\right]_{\mathcal{R}_a}\mathbf{P}_{b/a}(\theta) \\ &= J^{\mathcal{J}}\mathbf{P}^T_{b/a}(0)\underbrace{\mathbf{R}(\theta)^T[\mathbf{r}]_{\mathcal{R}_a}}_{=[\mathbf{r}]_{\mathcal{R}_a}}\underbrace{\left[\mathbf{r}^T\right]_{\mathcal{R}_a}\mathbf{R}(\theta)}_{=[\mathbf{r}^T]_{\mathcal{R}_a}}\mathbf{P}_{b/a}(0) \\ &= \mathbf{P}^T_{b/a}(0)\left[\mathbf{J}^{\mathcal{J}}_P\right]_{\mathcal{R}_a}\mathbf{P}_{b/a}(0). \end{aligned} \tag{20}$$

Equation (18) is evaluated at equilibrium

$$\begin{cases} \left[\overline{\mathbf{W}}_{\mathcal{J}/\mathcal{B},P}\right]_{\mathcal{R}_b} = \mathbf{P}^{\times 2}_{a/b}(\bar{\theta})\left[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}\right]_{\mathcal{R}_a} \\ 0 = \overline{T}_r + \left[\mathbf{r}^T_6\right]_{\mathcal{R}_a}\left[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}\right]_{\mathcal{R}_a} \end{cases} \tag{21}$$

and linearized around the equilibrium

$$\begin{cases} \delta\left[\mathbf{W}_{\mathcal{B}/\mathcal{J},P}\right]_{\mathcal{R}_b} + \begin{bmatrix} (\mathbf{r}^*) & \mathbf{0} \\ \mathbf{0} & (\mathbf{r}^*) \end{bmatrix}\mathbf{P}^{\times 2}_{a/b}(\bar{\theta})\left[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}\right]_{\mathcal{R}_a}\delta\theta \\ \quad + \mathbf{P}^{\times 2}_{a/b}(\bar{\theta})\delta\left[\mathbf{W}_{\mathcal{A}/\mathcal{J},P}\right]_{\mathcal{R}_a} \\ \quad = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \left[\mathbf{J}^{\mathcal{J}}_P\right]_{\mathcal{R}_b}\left(\delta\left[\dot{\boldsymbol{\omega}}^{\mathcal{B}}\right]_{\mathcal{R}_b} + \delta\ddot{\theta}[\mathbf{r}]_{\mathcal{R}_b}\right) \end{bmatrix} \\ \delta\ddot{\theta} = \dfrac{1}{J^{\mathcal{J}}}\left(\delta T_r + \left[\mathbf{r}^T_6\right]_{\mathcal{R}_a}\delta\left[\mathbf{W}_{\mathcal{A}/\mathcal{J},P}\right]_{\mathcal{R}_a}\right) \\ \quad -\left[\mathbf{r}^T\right]_{\mathcal{R}_b}\delta\left[\dot{\boldsymbol{\omega}}^{\mathcal{B}}\right]_{\mathcal{R}_b}. \end{cases} \tag{22}$$

Equation (22) shows that the wrench applied to the joint at equilibrium, represented by the vector $[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}]_{\mathcal{R}_a}$, introduces a stiffness in the motion of the revolute joint (factor multiplying $\delta\theta$). In addition to some possible wrenches applied to the system and defining the equilibrium (such as a buoyant force compensating for the gravity acceleration in the case of a stratospheric balloon, or a thrust providing the acceleration in the case of a launcher), the wrench $[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}]_{\mathcal{R}_a}$ results from the wrenches applied by rigid bodies given by (12), and it must be evaluated at equilibrium before linearization. Therefore, following the discussion on (12), $[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}]_{\mathcal{R}_a}$ may depend on some parameters of interest (masses, lengths, etc.). A numerical evaluation of the trim point, as it is done with current available software, is not adequate to capture it as an LFT (it will only evaluate a single, nominal configuration); on the contrary, evaluating $[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}]_{\mathcal{R}_a}$ while preserving its LFT structure allows to correctly re-inject it in the linearized model of the revolute joint. This observation justifies the need for the analytical derivation of the trim conditions and analytical linearization presented in this section, as well as for the dedicated procedure presented in Section V.

Example 1: Pendulum–Consider a pendulum around its stable equilibrium, composed of a revolute joint, a mass-less link, and an point mass, and assume that the mass is uncertain and represented by an LFT. The stiffness is proportional to the mass and must be computed as an LFT to be re-injected in the linearized model of the revolute joint. It is worth emphasizing that a system as simple as the pendulum has a parameter-dependent equilibrium in the sense of this brief, even though the equilibrium angle is fixed, and must be treated with the proposed approach to derive a multibody LFT model.

Example 2: Robotic arm–Consider a robotic arm with several bodies and revolute joints. It is sought to derive an LPV model where the scheduling parameters, whose variations are isolated with LFT formalism, are the equilibrium angle of each joint. In addition to rigid bodies' parameters (masses, etc.), the internal wrenches also depend on the equilibrium angles of the bodies. Therefore, it is first necessary to derive the DCM $\mathbf{P}_{b/i}(\bar{\boldsymbol{\theta}}^{\mathcal{B}})$ of each body as the product of the individual DCMs

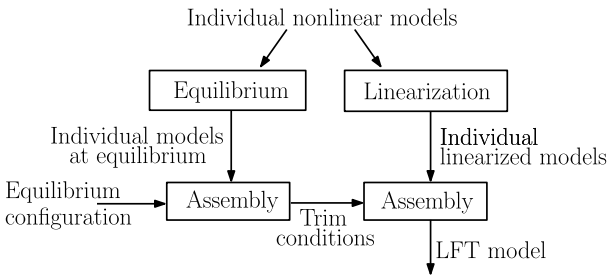

Fig. 2. Assembly, trim, and linearization algorithm.

of the revolute joints, which are LFT-LPV rotation matrices. Then, the wrenches are evaluated as LFTs of the equilibrium angles and rigid bodies' parameters, and finally re-injected in the linearized models of the revolute joints.

The transformation of the motion vector (Property 3.2) is also linearized

$$\begin{cases} \delta[\mathbf{a}]_{\mathcal{R}_a} = (\mathbf{r}^*)\mathbf{P}_{b/a}(\bar{\theta})[\bar{\mathbf{a}}]_{\mathcal{R}_b}\delta\theta + \mathbf{P}_{b/a}(\bar{\theta})\delta[\mathbf{a}]_{\mathcal{R}_b} \\ \delta\left[\mathbf{x}_P''^{\mathcal{A}}\right]_{\mathcal{R}_a} = \mathbf{P}^{\times 2}_{b/a}(\bar{\theta})\delta\left[\mathbf{x}_P''^{\mathcal{B}}\right]_{\mathcal{R}_b} + \delta\ddot{\theta}[\mathbf{r}_6]_{\mathcal{R}_a} \\ \delta\left[\mathbf{x}_P'^{\mathcal{A}}\right]_{\mathcal{R}_a} = \mathbf{P}^{\times 2}_{b/a}(\bar{\theta})\left[\mathbf{x}_P'^{\mathcal{B}}\right]_{\mathcal{R}_b} + \delta\dot{\theta}[\mathbf{r}_6]_{\mathcal{R}_a} \\ \delta\left[\mathbf{x}_P^{\mathcal{A}}\right]_{\mathcal{R}_a} = \begin{pmatrix} \mathbf{P}_{b/a}(\bar{\theta}) & \mathbf{0} \\ \mathbf{0} & \left.\frac{\partial\Theta^{\mathcal{J}}_{a/b}}{\partial\boldsymbol{\theta}^{\mathcal{B}}}\right|_{\text{eq}} \end{pmatrix}\delta\left[\mathbf{x}_P^{\mathcal{B}}\right]_{\mathcal{R}_b} \\ \qquad\qquad + \begin{pmatrix} (\mathbf{r}^*)\mathbf{P}_{b/a}(\bar{\theta})\left[\overline{OP}\right]_{\mathcal{R}_b} \\ \left.\frac{\partial\Theta^{\mathcal{J}}_{a/b}}{\partial\theta}\right|_{\text{eq}} \end{pmatrix}\delta\theta. \end{cases} \tag{23}$$

Once again, the position vector at equilibrium $[\overline{OP}]_{\mathcal{R}_b}$ must be computed analytically because it may have LFT dependency on the lengths or angles at equilibrium.

From (22) and (23), the linearized model of the revolute joint takes $\delta[\mathbf{W}_{\mathcal{A}/\mathcal{J},P}]_{\mathcal{R}_a}$, $\delta[\mathbf{m}_P^{\mathcal{B}}]_{\mathcal{R}_b}$ and δT_r as inputs and returns $\delta[\mathbf{m}_P^{\mathcal{A}}]_{\mathcal{R}_a}$, $\delta[\mathbf{W}_{\mathcal{J}/\mathcal{B},P}]_{\mathcal{R}_b}$ and $(\delta\ddot{\theta}, \delta\dot{\theta}, \delta\theta)$ as outputs. With the proposed evaluation of the vectors $[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}]_{\mathcal{R}_a}$ and $[\overline{OP}]_{\mathcal{R}_b}$ as LFT models, the linearized revolute joint is also an LFT model except for gains $((\partial\Theta^{\mathcal{J}}_{a/b})/\partial\theta)|_{\text{eq}}$ and $((\partial\Theta^{\mathcal{J}}_{a/b})/(\partial\boldsymbol{\theta}^{\mathcal{B}}))|_{\text{eq}}$ which are used to propagate Euler angles (see the discussion in Appendix).

V. Assembly, Trim, and Linearization Algorithm

As discussed in Section IV, it is necessary to perform an analytical trimming to preserve LFT dependencies. This is possible by assembling the model of the structure at equilibrium from the individual models (12) and (21). Then, the trim conditions, expressed as LFTs, are re-injected in the assembly of the individual linearized models [see Fig. 1, (22) and (23)]. This procedure is schematized in Fig. 2.

More precisely, let us consider a tree-like structure composed of: 1) a base, which is either a parent body described by its forward dynamics (six DOF) or the ground (no DOF); 2) children bodies described by their inverse dynamics (no additional DOF); and 3) n revolute joints (n DOF). Each body may be connected to any number of other bodies or joints, as long as there is no closed kinematic loop.

Step 1 *(Geometry at Equilibrium, or Forward Recurrence):* This step aims at computing the geometrical trim conditions as LFTs of the parameters of interest: the DCM $\mathbf{P}_{b/i}(\bar{\boldsymbol{\theta}}^{\mathcal{B}})$ for each body and the position vector $[\overline{OP}]_{\mathcal{R}_b}$ at each revolute joint. These quantities are initially defined at the base (either a parent body or ground) and are propagated from the base to the other bodies and joints. The DCM is transformed at each revolute joint: $\mathbf{P}_{b/i}(\bar{\boldsymbol{\theta}}^{\mathcal{B}}) = \mathbf{P}_{a/i}(\bar{\boldsymbol{\theta}}^{\mathcal{A}})\mathbf{P}_{b/a}(\bar{\theta})$; and the position vector is transformed at each revolute joint: $[\overline{OP}]_{\mathcal{R}_b} = \mathbf{P}_{a/b}(\bar{\theta})[\overline{OP}]_{\mathcal{R}_a}$, and at each rigid body: from a port P to a port C: $[\overline{OC}]_{\mathcal{R}_b} = [\overline{OP}]_{\mathcal{R}_b} + [CP]_{\mathcal{R}_b}$. The DCMs $\mathbf{P}_{b/a}(\bar{\theta})$ and the positions $[CP]_{\mathcal{R}_b}$ can be LFT models, and these operations preserve the LFT form; hence, all $\mathbf{P}_{b/i}(\bar{\boldsymbol{\theta}}^{\mathcal{B}})$ and $[\overline{OP}]_{\mathcal{R}_b}$ are finally obtained as LFT models.

Step 2 *(Wrenches at Equilibrium, or Backward Recurrence):* This step aims at computing the wrenches at equilibrium $[\overline{\mathbf{W}}_{\mathcal{A}/\mathcal{J},P}]_{\mathcal{R}_a}$ in the revolute joints as LFTs of the parameters of interest. For this, the wrenches are propagated from the outer bodies (end of the open kinematic chain) to the base using the models (12) and (21), which are also compliant with the LFT formalism (and where the matrices $[\mathbf{D}_P^{\mathcal{B}}]_{\mathcal{R}_b}$ can be LFT models as well). Note that step 2 requires the DCMs $\mathbf{P}_{b/i}(\bar{\boldsymbol{\theta}}^{\mathcal{B}})$ computed at step 1 [see (13)].

Step 3 *(Linearized Model):* Finally, the individual linearized models [see Fig. 1, and (22) and (23)] are assembled while re-injecting the trim conditions obtained as LFT models in steps 1 and 2. Therefore, the resulting model is a fully parameterized LFT model accounting for the parameter-dependent equilibrium.

Since the physical origin of all the parameters has been preserved during the whole procedure, the LFT model exactly covers all the plants without introducing conservatism or fitting error. In practice, the procedure can be implemented on MATLAB-Simulink; in this case, the trim conditions computed as LFT models in steps 1 and 2 are evaluated as input-output transfers after implementation of models (12) and (21) as static LFT models. Since only basic block diagram manipulations are applied, the procedure can be executed in reasonable time even for complex systems. However, although the trim conditions calculated in steps 1 and 2 can be expressed with minimal parametric dependency on the parameters of interest, since they are in turn re-injected at step 3, there can be redundant occurrences in the linearized LFT model; reduction techniques can be used to reduce the order of the block $\boldsymbol{\Delta}$ [16].

VI. Application Example

A. Presentation of the System

The two-link robotic arm presented in Fig. 3 is subject to the gravity represented by the vector $\mathbf{g}$, which is equivalent to an acceleration $\mathbf{a} = -\mathbf{g}$ in the proposed approach. The reference frame in acceleration is noted $\mathcal{R} = (O, \mathbf{x}, \mathbf{y}, \mathbf{z})$. The arm is composed of three bodies $\mathcal{B}_1$, $\mathcal{B}_2$, and $\mathcal{B}_3$. The revolute joints $\mathcal{J}_1$ and $\mathcal{J}_2$, which allow the rotation around $\mathbf{x}$, are actuated with torques T_1 and T_2. $\mathcal{B}_3$ is a point mass representing the end-effector carrying a load, and it is rigidly connected to $\mathcal{B}_2$ (no degree of freedom). The characteristics of the rigid bodies are indicated in Table I. The position of the center of gravity (CoG) is the distance of the CoG from the body's left tip (in Fig. 3), normalized by the length of the body.

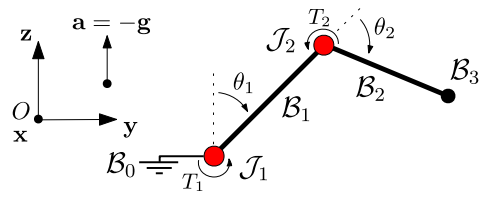

Fig. 3. Two-link robotic robotic arm.

TABLE I

PHYSICAL PARAMETERS OF THE ROBOTIC ARM

	$\mathcal{B}_1$	$\mathcal{B}_2$	$\mathcal{B}_3$
Mass m_i (kg)	3 (± 20%)	2	5 (± 20%)
Moment of inertia J_i (kg.m^2)	0.2 (± 20%)	0.1	0
Length L_i (m)	1	1 (± 20%)	0
Position of the CoG ρ_i (-)	0.3 (± 20%)	0.5	0

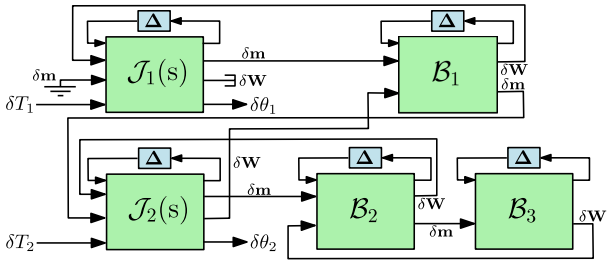

Fig. 4. Multibody LFT model of the robotic arm.

Uncertainties of ±20% have been set on some parameters. The scheduling parameters $t_1 = \tan(\bar{\theta}_1/2)$ and $t_2 = \tan(\bar{\theta}_2/2)$ are defined as uncertain parameters in the revolute joints' blocks.

B. Multibody LFT Modeling

The proposed approach is implemented on MATLAB with the robust control toolbox. The uncertain and scheduling parameters are declared with the routine ureal. The trim conditions (DCMs, position vectors, wrenches) are evaluated with the routine ulinearize as static input–output transfers in separated Simulink files, where the individual static LFT models of each body at equilibrium are assembled (see steps 1 and 2). Once the trim conditions are obtained as LFT models, they are re-injected in the linearized models which are assembled as in Fig. 4 (see step 3). For readability, it is indicated whether the connections represent a motion vector $\delta\mathbf{m}$ or a wrench $\delta\mathbf{W}$, but the full nomenclature adopted in previous sections is omitted. The LFT dependencies of the trim conditions are carried by the blocks $\mathbf{\Delta}$ of the revolute joints. A damping $K_d = 0.1$ Ns rad^{-1} and a stiffness $K_p = 0.1$ N rad^{-1} are added to the linear models of the revolute joints. The procedure took 20 s on an Intel Core i7 processor.

C. Comparison With Simscape Multibody

To validate the proposed approach, the model of the same robotic arm is built with Simscape Multibody and linearized around the equilibrium. The proposed LFT model matches Simscape's model in the nominal configuration of the uncertain parameters and across all the angular configurations, as shown in Fig. 5, where the relative error between the two models stays small even in the worst case configurations around $\bar{\theta}_1 = \pm 90°$. Non-nominal configurations were also tested and matched the corresponding Simscape's model.

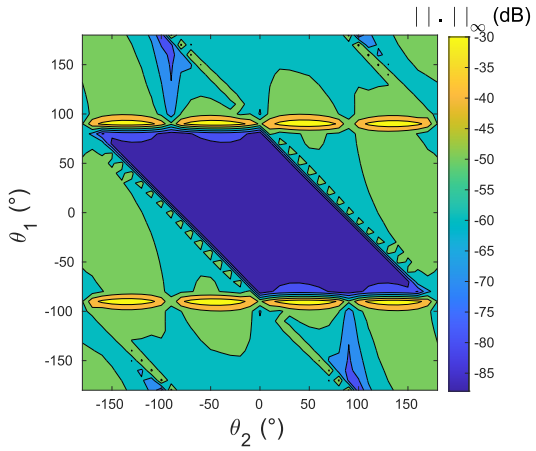

Fig. 5. $||\ (\mathbf{G}_1 - \mathbf{G}_2)\mathbf{G}_2^{-1}\ ||_\infty$ across angular configurations, where $\mathbf{G}_1$ is the nominal LFT model (no parametric uncertainty) and $\mathbf{G}_2$ is the Simscape Multibody's model.

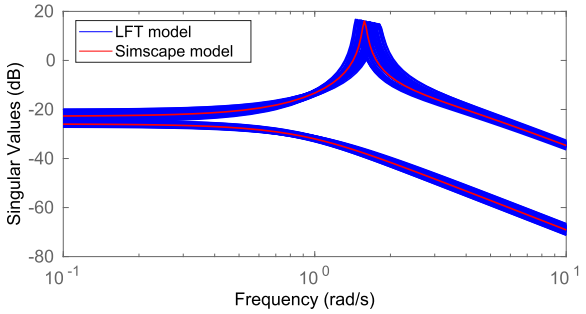

Fig. 6. Singular values of $[\delta T_1\ ,\ \delta T_2]^T \rightarrow [\delta\theta_1\ ,\ \delta\theta_2]^T$ for $\bar{\theta}_1 = 70°$ and $\bar{\theta}_2 = 30°$ (300 samples of the LFT model).

Moreover, Fig. 6 presents the singular values of the transfer $[\delta T_1\ ,\ \delta T_2]^T \rightarrow [\delta\theta_1\ ,\ \delta\theta_2]^T$ for both the models in one angular configuration. Let us emphasize that the proposed LFT model contains all the configurations of the scheduling parameters t_1 and t_2 as well as the parametric uncertainties in one single model, while the Simscape model needs to be reevaluated, trimmed, and linearized for every geometric or parametric configuration.

D. Robust LPV Control

To conclude, a robust LPV controller is proposed to illustrate the compatibility of the proposed approach with classical robust control tools and to show the advantages of the LFT model. The angles are limited to the following operating ranges: $\theta_1 \in [45°, 90°]$ and $\theta_2 \in [45°, 135°]$, and the set of scheduling parameters is noted $\boldsymbol{\tau} = \{t_1, t_2\}$.

Noting $\delta\boldsymbol{\theta}_{\text{ref}} = [\delta\theta_1^{\text{ref}}\ ,\ \delta\theta_2^{\text{ref}}]^T$ the vector of reference angles, $\delta\mathbf{e} = \delta\boldsymbol{\theta}_{\text{ref}} - [\delta\theta_1\ ,\ \delta\theta_2]^T$, and $\delta\mathbf{T} = [\delta T_1\ ,\ \delta T_2]^T$, the LPV controller $\mathbf{K}(s, \boldsymbol{\tau})$ is such that

$$\delta\mathbf{T} = \mathbf{K}(s, \boldsymbol{\tau})\delta\mathbf{e}. \tag{24}$$

Let the real matrices of appropriate dimensions $\mathbf{A}(\boldsymbol{\tau})$, $\mathbf{B}(\boldsymbol{\tau})$, $\mathbf{C}(\boldsymbol{\tau})$, and $\mathbf{D}(\boldsymbol{\tau})$ define the state-space representation of $\mathbf{K}(s, \boldsymbol{\tau})$. The scheduling surface $\mathbf{S}(\boldsymbol{\tau})$ is defined as

$$\mathbf{S}(\boldsymbol{\tau}) = \left[\begin{array}{c|c} \mathbf{A}(\boldsymbol{\tau}) & \mathbf{B}(\boldsymbol{\tau}) \\ \hline \mathbf{C}(\boldsymbol{\tau}) & \mathbf{D}(\boldsymbol{\tau}) \end{array}\right] = \mathbf{M}_0 + \mathbf{M}_1 t_1 + \mathbf{M}_2 t_2 \tag{25}$$

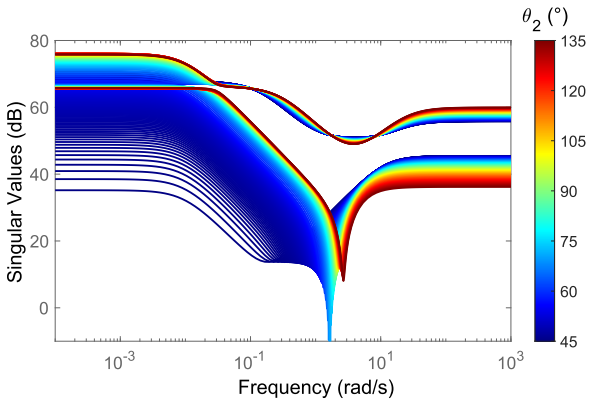

Fig. 7. Singular values of $\mathbf{K}(s, \boldsymbol{\tau})$, $\bar{\theta}_1 = 45°$, $\bar{\theta}_2 \in [45°, 135°]$.

where the matrices $\mathbf{M}_0$, $\mathbf{M}_1$, and $\mathbf{M}_2$ are to be tuned, and the LPV controller $\mathbf{K}(s, \boldsymbol{\tau})$ reads

$$\mathbf{K}(s, \boldsymbol{\tau}) = \mathcal{F}_u\left(\mathbf{S}(\boldsymbol{\tau}), \frac{\mathbf{I}_{n_s}}{s}\right) = \mathcal{F}_u\left(\mathbf{K}(s), \boldsymbol{\Delta}_{\boldsymbol{\tau}}^K\right) \tag{26}$$

where $\mathcal{F}_u$ refers to the upper LFT, n_s is the number of states of the controller, and the block $\Delta_{\boldsymbol{\tau}}^K$ isolates the occurrences of t_1 and t_2.

The value $n_s = 3$ was chosen, and after defining the weighting functions $\mathbf{W}_T = 1/1500\,\mathbf{I}_2$ (to limit the actuator's efforts) and $\mathbf{W}_e(s) = (s + 1/2s + 0.02)\,\mathbf{I}_2$ (to penalize low-frequency tracking error), the robust, structured $\mathcal{H}_\infty$ problem

$$\begin{aligned} &\underset{\mathbf{M}_0,\mathbf{M}_1,\mathbf{M}_2}{\text{minimize}} \quad \gamma_2 \quad \text{s.t.} \quad \max_{\boldsymbol{\tau},\boldsymbol{\Delta}}\{||\ \delta\boldsymbol{\theta}_{\text{ref}} \rightarrow \mathbf{W}_T\delta\mathbf{T}\ ||_\infty\} < \gamma_2 \\ &\text{subject to: } \max_{\boldsymbol{\tau},\boldsymbol{\Delta}}\{||\ \delta\boldsymbol{\theta}_{\text{ref}} \rightarrow \mathbf{W}_e\delta\mathbf{e}\ ||_\infty\} < \gamma_1 < 1 \end{aligned} \tag{27}$$

was solved with MATLAB routine `systune`, based on the algorithm presented in [17]. A performance ($\gamma_1 = 0.97$, $\gamma_2 = 0.69$) was obtained (corresponding to the worst case $\mathcal{H}_\infty$ norms of the transfers), and Fig. 7 represents the LPV controller. Since the proposed modeling approach provided all the parametric configurations of both the uncertain and scheduling parameters in one single LFT model, the robustness and the LPV controller synthesis were addressed together in one single control design iteration, and the resulting performance is guaranteed across all the parametric configurations.

VII. Conclusion

After introducing a multibody modeling framework based on the Newton–Euler equations, it was shown why a numerical trim computation is not adequate to derive an LFT model, and a specific assembly procedure, based on the linearization of the equations of motion at the substructure level, was proposed to solve this issue. An application to a robotic arm was outlined to show how the proposed approach can be implemented on MATLAB and used for control design.

Appendix

The transformation $\Theta(.)$ from Definition 2.5 cannot be expressed as an LFT of uncertain or varying Euler angles, because it includes trigonometric functions. Therefore, propagating Euler angles from one body to another (with the function $\Theta_{b/a}^{\mathcal{J}}$ from Property 3.2) cannot be done while preserving the LFT form. As a consequence, if Euler angles are defined as output measurements, the corresponding output gains cannot always be obtained as exact LFTs, and rational approximations of $\Theta_{a/b}^{\mathcal{J}}$ and its derivatives may be necessary (it can be noted that for problems in a single plane, the transformation $\Theta_{a/b}^{\mathcal{J}}$ becomes trivial and this issue disappears). Nonetheless, the dynamical model can always be obtained because the inclusion of the acceleration vector $\mathbf{a}$ in the motion vector allows to dispense with Euler angles in the equations of the dynamics (see (14) in Section IV-B).

References

[1] B. Rong, X. Rui, L. Tao, and G. Wang, "Theoretical modeling and numerical solution methods for flexible multibody system dynamics," *Nonlinear Dyn.*, vol. 98, no. 2, pp. 1519–1553, Oct. 2019.

[2] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[3] A. Marcos, D. G. Bates, and I. Postlethwaite, "Exact nonlinear modelling using symbolic linear fractional transformations," *IFAC Proc. Volumes*, vol. 38, no. 1, pp. 190–195, 2005.

[4] Z. Szabó *et al.*, "Development of an integrated LPV/LFT framework: Modeling and data-based validation tool," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 1, pp. 104–117, Jan. 2011.

[5] H. Pfifer and S. Hecker, "Generation of optimal linear parametric models for LFT-based robust stability analysis and control design," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 1, pp. 118–131, Jan. 2011.

[6] C. Roos, G. Hardier, and J.-M. Biannic, "Polynomial and rational approximation with the APRICOT library of the SMAC toolbox," in *Proc. IEEE Conf. Control Appl. (CCA)*, Oct. 2014, pp. 1473–1478.

[7] D. Alazard, C. Cumer, and K. Tantawi, "Linear dynamic modeling of spacecraft with various flexible appendages and on-board angular momentums," in *Proc. 7th Int. ESA Conf. Guid., Navigat. Control Syst.*, 2008, vol. 41, no. 2, pp. 11148–11153.

[8] D. Alazard and F. Sanfedino, "Satellite dynamics toolbox for preliminary design phase," in *Proc. 43rd Annu. AAS Guid. Control Conf.*, vol. 172, 2020, pp. 1461–1472.

[9] D. Alazard, J. A. Perez, C. Cumer, and T. Loquen, "Two-input two-output port model for mechanical systems," in *Proc. AIAA Guid., Navigat., Control Conf.* Reston, VA, USA: American Institute of Aeronautics and Astronautics, Jan. 2015, p. 1778.

[10] F. Sanfedino, D. Alazard, V. Pommier-Budinger, A. Falcoz, and F. Boquet, "Finite element based N-port model for preliminary design of multibody systems," *J. Sound Vib.*, vol. 415, pp. 128–146, Feb. 2018.

[11] F. Sanfedino, V. Preda, V. Pommier-Budinger, D. Alazard, F. Boquet, and S. Bennani, "Robust active mirror control based on hybrid sensing for spacecraft line-of-sight stabilization," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 1, pp. 220–235, Jan. 2021.

[12] J. A. Perez, C. Pittet, D. Alazard, and T. Loquen, "Integrated control/structure design of a large space structure using structured H_∞ control," *IFAC-PapersOnLine*, vol. 49, no. 17, pp. 302–307, 2016.

[13] E. Kassarian, F. Sanfedino, D. Alazard, H. Evain, and J. Montel, "Modeling and stability of balloon-borne gondolas with coupled pendulum-torsion dynamics," *Aerosp. Sci. Technol.*, vol. 112, May 2021, Art. no. 106607.

[14] P. Zipfel, *Modeling and Simulation of Aerospace Vehicle Dynamics*. Reston, VA, USA: American Institute of Aeronautics and Astronautics, 2014.

[15] V. Dubanchet, "Modeling and control of a flexible space robot to capture a tumbling debris," Ph.D. dissertation, Ecole Polytechnique de Montréal, Montreal, QC, Canada, 2016.

[16] A. Varga and G. Looye, "Symbolic and numerical software tools for LFT-based low order uncertainty modeling," in *Proc. IEEE Int. Symp. Comput. Aided Control Syst. Design*, no. 1, Aug. 1999, pp. 1–6.

[17] P. Apkarian and D. Noll, "Nonsmooth H_∞ synthesis," *IEEE Trans. Autom. Control*, vol. 51, no. 1, pp. 71–86, Jan. 2006.