Non-inertial modeling and efficient simulation of a miniature R/C helicopter using the Body Decomposition Approach

Víctor Dimitri Palacios Uyaguari, Ernesto Olguín Díaz
CINVESTAV, Unidad-Saltillo
Saltillo, México
dimitrizero@gmail.com, ernesto.olguin@cinvestav.edu.mx

***Abstract*— In this work a dynamic model of the rotor of a helicopter is obtained with one single forward recursion using the novel *Body Decomposition Approach*, which allows for modular and simple programming of simple body blocks. The rotor is treated as a non-inertial tree-like multibody mechanical system. The generalized coordinates are represented by the pose of the helicopter and the articular mechanism movements of the blades. The generalized forces are compounded by the control forces at each coordinates plus the external aerodynamic effects which can be included as lumped forces by the power transmission principle.**

I. INTRODUCTION

Recent technology advances had made possible the study and development of an unmanned aerial vehicles (UAV) which has been used to perform a wide variety of functions such as scientific research, remote sensing, transport and interact in hostile environment areas which may be too dangerous for piloted aircrafts. While early UAVs were not fully autonomous (Shim, 2000), the field of air-vehicle autonomy has been a recently emerging field, whose economics is largely driven by the military to develop battle-ready technology. An UAV is considered difficult to use or even dangerous (Pir, 2009) because human intervention is required to command the UAV altogether with normal control algorithms (Shim, 2000). Compared to the manufacturing of UAV flight hardware, the market for autonomy technology is fairly immature and undeveloped. Because of this, autonomy is the bottleneck for future UAV developments, and the overall value and rate of expansion of the future UAV market could be largely driven by advances to be made in the field of autonomy (Dickerson, 2007).

The helicopter is a particular type of UAV that shows a versatile operational regime that surpasses other UAVs. In contrast to fixed-wing airplanes, the rotary-wing (rotor) of the helicopter has the ability to perform different flight regimes like hover, backward, lateral and pure vertical flight. These characteristics have attracted civil and military interests in applications such as traffic, surveillance, air pollution monitoring, area mapping, agricultural applications, exploration and many others. However to achieve such flying operational versatility helicopters requires complex controllers because helicopters are under-actuated mechanisms whose dynamic model exhibits high non-linearities and its physical parameters are difficult to measure accurately.

We present a high fidelity model of a helicopter, based on the classical architecture of one vertical main rotor and a secondary horizontal compensation tail rotor -as the one shown in Fig.1 - generating a 13 DOF (Degrees Of Freedom) model which traps all the multi-body coupled inertial effects and most of the aerodynamics nonlinearities, based on the Body Decomposition Approach developed in (Olguin, 2011). This approach allows to obtain the nonlinear dynamic model in a modular and simple algorithm which allows for the addition of any external forces contribution at each body level. The resulting simulator provides a computationally efficient model-based estimation.

Figura 1: Xcell-90 helicopter

The implementation is made using MATLAB-SIMULINK, creating a specific Body Decomposition modeling toolbox, and it is applied in a radio-controlled mini helicopter, model X-cell 90 of Miniature-Aircraft©.

II. MODELING BACKGROUND

II-A. Spatial Kinematics

To define the kinematics of the model the *Screw* theory is used to express 6D (6 Dimension) point vectors, also called *spatial vectors* whose algebra has been more recently used and widely explained by (Featherstone, 2000). A *Screw* is a mathematical term defined by a pair $(\mathbf{a}, \mathbf{b}) \in I\!R^3$ of 3D vectors that express a product of the form $\mathbf{a} + \mathbf{b} \times \mathbf{x} = \mathbf{y}$ for all 3D vectors $(\mathbf{a}, \mathbf{b}, \mathbf{x}, \mathbf{y}) \in I\!R^3$ where $\mathbf{x}$ and $\mathbf{y}$ are inputs variables, $\mathbf{a}$ is a *line vector*, and $\mathbf{b}$ is a *free vector* (Featherstone, 2000).

We can use some physical screws in the motion of rigid bodies like the *twist*, defined by the pair $(\mathbf{v}, \omega)$, and the *wrench*, defined by the pair $(\mathbf{f}, \mathbf{n})$:

$$\nu \triangleq \begin{pmatrix} \mathbf{v} \\ \omega \end{pmatrix} \in I\!M \subset I\!R^6, \qquad \mathbf{F} \triangleq \begin{pmatrix} \mathbf{f} \\ \mathbf{n} \end{pmatrix} \in I\!F \subset I\!R^6$$

where $\mathbf{v}$ and ω are the linear and angular velocities of a non-inertial frame such as those attached to a body and $\mathbf{f}$ and $\mathbf{n}$ are the force and torque applied to the body. Notice that each of these spatial vectors defines a vector space, called the motion space $\mathbb{M}$ and the force space $\mathbb{F}$, where both are subset of the 6D real numbers. Also notice that screws are *point vectors*, that is they belong to an specified Euclidian point in space an no other. Although this is sufficient for definition of these spacial vectors, the following sections assumes that both twist and wrench are point vectors at the same point in a rigid body, unless otherwise specified.

On the other hand, the *pose*

$$\mathbf{x} \triangleq \begin{pmatrix} \mathbf{d} \\ \theta \end{pmatrix} \in \mathbb{R}^{3+m}$$

of a mobile (non-inertial) frame Σ_m (for example the one attached to a rigid body) has 6 DOF, consisting in the distance $\mathbf{d} \in \mathbb{R}^3$ that represents Cartesian position of the frame origin measured from an inertial frame Σ_0 and the set of attitude parameters $\theta \in \mathbb{R}^m$ (such as Euler angles, roll-pitch-yaw or any other representation) that defines the rotation matrix $R(\theta) \in SO(3)$ that transforms the noninertial coordinates $\bar{\mathbf{v}}^{(m)}$ of any 3D vector $\bar{\mathbf{v}}$ to the inertial coordinates $\bar{\mathbf{v}}^{(0)}$:

$$\bar{\mathbf{v}}^{(0)} = R(\theta)\bar{\mathbf{v}}^{(m)}$$

and whose time derivative defines uniquely the angular velocity of the corresponding frame by the relationship: $\dot{R} = \left[\omega^{(0)}\times\right]R = R\left[\omega^{(m)}\times\right]$.

Coordinates transforation of screws (both motion and force screws) can be carried out by the use of a linear operator $\mathscr{R}(\theta) : \mathbb{R}^6 \mapsto \mathbb{R}^6$ called **extended rotation**:

$$\mathscr{R}(\theta) \triangleq \begin{bmatrix} R(\theta) & 0 \\ 0 & R(\theta) \end{bmatrix} \in SO_6 \tag{1}$$

such that

$$\nu^{(0)} = \mathscr{R}(\theta)\nu^{(m)}; \qquad \mathbf{F}^{(0)} = \mathscr{R}(\theta)\mathbf{F}^{(m)}$$

Being screws point vectors, equivalent translated screws can be obtained from appropriate linear transformations. In the case of motion screws, take the example of the center of mass, whose twist ν_c can be computed from the twist ν of the non-inertial frame attached to it and the distance $\mathbf{r}_c$ from the origin of the mentioned frame to the center of mass of the body:

$$\begin{aligned} \mathbf{v}_c &= \mathbf{v} + \omega \times \mathbf{r}_c \\ \omega_c &= \omega \end{aligned}$$

Then the **extended translation** operator $\mathscr{T} : \mathbb{M} \mapsto \mathbb{M}$ defined as follows

$$\mathscr{T}(\mathbf{a}) \triangleq \begin{bmatrix} I & -[\mathbf{a}\times] \\ 0 & I \end{bmatrix} \quad \forall\ \mathbf{a} \in \mathbb{R}^3 \tag{2}$$

is a linear transformation for translating motion screws a distance $\mathbf{a} \in \mathbb{R}^3$, where the $[\mathbf{a}\times] \in SS(3)$ is the cross product operator defined as a skew symmetric matrix of order 3 that fulfils the 3D vector cross product $[\mathbf{a}\times]\mathbf{b} = \mathbf{a} \times \mathbf{b}$. Notice that $\mathscr{T}^T : \mathbb{F} \mapsto \mathbb{F}$. This is that the transpose of the extended translation operator, translates wrenches. Then the translation of both motion and force screws is given as

$$\nu_c = \mathscr{T}(\mathbf{r}_c)\nu; \qquad \mathbf{F} = \mathscr{T}^T(\mathbf{r}_c)\mathbf{F}_c$$

The *extended cross product* is a vector product of 6D screws which is an extension of the 3D vector cross product defined as

$$\mathbf{a} \wedge \mathbf{b} \triangleq \begin{pmatrix} \mathbf{a}_2 \times \mathbf{b}_1 + \mathbf{a}_1 \times \mathbf{b}_2 \\ \mathbf{a}_2 \times \mathbf{b}_2 \end{pmatrix}, \quad \forall\ (\mathbf{a}, \mathbf{b}) \in \mathbb{R}^6$$

Then an **extended cross product operator** $\Omega(\mathbf{a}) = [\mathbf{a}\wedge]$ arises as follows

$$\Omega(\mathbf{a}) = [\mathbf{a}\wedge] \triangleq \begin{bmatrix} [\mathbf{a}_2\times] & [\mathbf{a}_1\times] \\ 0 & [\mathbf{a}_2\times] \end{bmatrix} \quad \forall\ \mathbf{a} \in \mathbb{R}^6 \tag{3}$$

with the following properties for all $(\mathbf{a}, \mathbf{b}) \in \mathbb{R}^6$, and $\mathbf{r} \in \mathbb{R}^3$:

$$\begin{aligned} \Omega(\mathbf{a})\mathbf{a} &= 0 \\ \Omega(\mathbf{a})\mathbf{b} &= -\Omega(\mathbf{b})\mathbf{a} \\ \Omega^T(\mathbf{a})\mathbf{b} &= S(\mathbf{b})\mathbf{a} \end{aligned}$$

where $S(\mathbf{b}) \in SS(6)$ is a skew symmetric matrix of order 6 defined as follows

$$S(\mathbf{b}) \triangleq \begin{bmatrix} 0 & [\mathbf{b}_1\times] \\ [\mathbf{b}_1\times] & [\mathbf{b}_2\times] \end{bmatrix} \tag{4}$$

Finally, the rigid motion of a non-inertial frame Σ_m, i.e. the 6 DOF that defines the pose of the frame with respect to an inertial one can be expressed by the use of single **Kinematic Operator** $\mathscr{X}(\mathbf{d}, R)$ is defined by using the extended operators as follows:

$$\mathscr{X}(\mathbf{d}, R) \triangleq \mathscr{R}^T(R)\mathscr{T}\left(\mathbf{d}^{(0)}\right) = \mathscr{T}\left(\mathbf{d}^{(m)}\right)\mathscr{R}^T(R) \tag{5}$$

whose time derivative happens to be (Olguin, 2011):

$$\begin{aligned} \dot{\mathscr{X}}(\mathbf{d}, R) &= -\mathscr{X}(\mathbf{d}, R)\ \Omega\left(\nu^{(0)}\right) \\ &= -\Omega\left(\nu^{(m)}\right)\ \mathscr{X}(\mathbf{d}, R) \end{aligned} \tag{6}$$

II-B. Spatial Dynamics

Kirchhoff equations of motion for a rigid body are the quasi-Lagrangian version of Newton equations of motion (Olguin, 2011), in which the coordinates of all variables are expressed with respect to the non-inertial frame attached to the body. These equations can be written using the same notation used to define the screws as follows

$$\begin{aligned} \frac{d}{dt}\frac{\partial K}{\partial \mathbf{v}} + \omega \times \frac{\partial K}{\partial \mathbf{v}} &= \mathbf{f} \\ \frac{d}{dt}\frac{\partial K}{\partial \omega} + \mathbf{v} \times \frac{\partial K}{\partial \mathbf{v}} + \omega \times \frac{\partial K}{\partial \omega} &= \mathbf{n} \end{aligned}$$

where the kinetic energy, $K = \frac{1}{2}\nu^T M \nu$, is a quadratic matrix expression of the twist ν at the origin of frame Σ_m and M is the constant Inertia Matrix of order 6:

$$M = \begin{bmatrix} mI & -m[r_c\times] \\ m[r_c\times] & I_c - m[r_c\times] \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{7}$$

where m is the total mass of the body, $I_c \in I\!R^{3\times3}$ is the inertia tensor calculated at the center of mass, and $r_c \in I\!R^3$ stands for the distance from the origin of the reference frame to its center of mass. Notice that the product $M\nu = \mathbf{P} = \frac{\partial K}{\partial \nu}$, where $\mathbf{P} \in I\!F$ is the extended momentum, whose time derivative would yield the extended Newton-Euler equation of motion. Then $M : I\!M \mapsto I\!F$ is also a linear operator that maps the motion extended subset to the force extended subset.

Using the spatial representation screws, Kirchhoff equations can be written in a compact (spatial) form (Featherstone, 2000) as:

$$M\dot{\nu} - \Omega^T(\nu)M\nu = \mathbf{F}$$

where $\mathbf{F} = \mathbf{F}_g + \mathbf{F}_D + \mathbf{F}_u$ represent all the exogenous force/torque influences on the body like gravity $\mathbf{F}_g = m\mathbf{G}$ -where $\mathbf{G} = (R^T\mathbf{g}_0, 0)^T \in I\!M$ is the gravity acceleration screw in body's frame coordinates, and $\mathbf{g}_0$ is the inertial coordinates gravity vector-, dissipative (like aerodynamic forces) $\mathbf{F}_D = -D(\cdot)\nu$ and active control wrenches. Using the above formulation and properties of the extended cross product, the Coriolis term can be written as $C(\nu)\nu = -\Omega^T(\nu)M\nu = -S(M\nu)\nu$. Then the Coriolis matrix of the motion equation of a rigid body when expressed on its own frame can be written as a skew-symmetric matrix. Then the motion equation can be written as

$$M\dot{\nu} - S(M\nu)\nu - m\mathbf{G} - \mathbf{F}_D = \mathbf{F}_u \tag{8}$$

In can be proved (Olguin, 2011) that in the absence of dissipative forces the product

$$\langle \nu, \mathbf{F}_u \rangle = \dot{K} + \dot{U} = P \tag{9}$$

is the power flow of conservative energies in the rigid body.

II-C. Lagrangian Dynamics

Lagrangian formulation (Goldstein, 1980) :

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} = \tau \tag{10}$$

is a methodology for modeling multi-bodies mechanical systems where the bodies have inner holonomic constraints, and the system can be expressed uniquely by a set of n generalized coordinates $\mathbf{q} \in I\!R^n$ that represents the directions of admissible motion of the system, and their time derivatives.

In (10) the Lagrangian function $L = K(\mathbf{q}, \dot{\mathbf{q}}) - U(\mathbf{q})$ is the difference of kinetic and potential energy and τ is the generalized force vector whose coordinates act in the admissible motion directions corresponding to the generalized coordinates $\mathbf{q}$ of the system. In this formulation the *twist* of any body is constrained by the holonomic constraints of the system and shall be given by a kinematic equation of the form:

$$\nu_i^{(0)} \;=\; {}^0J_i(\mathbf{q})\dot{\mathbf{q}} \tag{11}$$

where ${}^0J_i$ is a Jacobian matrix that transforms the generalized velocity to the inertial coordinates expression of each body's *twist*, and the kinetic energy $K(\mathbf{q}, \dot{\mathbf{q}})$ can be expressed using (11) as a function of both the generalized coordinates $\mathbf{q}$ and the generalized velocity $\dot{\mathbf{q}}$:

$$K = \frac{1}{2}\dot{\mathbf{q}}^T H(\mathbf{q})\dot{\mathbf{q}} \tag{12}$$

where $H(\mathbf{q})$ is the system inertia matrix, defined as

$$H(\mathbf{q}) = \sum_{i=1}^{N} \left[{}^0J_i^T \mathscr{R}_i \; M_i \; \mathscr{R}_i^T \; {}^0J_i \right] \in I\!R^{n\times n} \tag{13}$$

Notice that the dimensions of this inertia matrix depends on the dimension n of the generalized coordinates of the system, while the summation limit is the number N of different body elements, which happens to be the same only in the case of open kinematic chains.

Using (12) in (10), the dynamic equation of motion of a multi-body mechanical system in the absence of gravity is given as follows

$$\tau = H(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) \tag{14}$$

where the gravity vector $\mathbf{g}(\mathbf{q}) = \frac{\partial U}{\partial \mathbf{q}} = -\sum_{i=1}^{N} \left[m_i {}^0J_{\nu_{cm_i}}^T \right] \mathbf{g}_0$ is the gradient of the potential energy and can be calculated by the summation of the mass-weighted transposes of the linear velocity jacobian of the center of mass of each element, by the inertial coordinates gravity vector $\mathbf{g}_0$; and the Coriolis term, explicitly given as $C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \dot{H}(\mathbf{q})\dot{\mathbf{q}} - \frac{1}{2}\frac{\dot{\mathbf{q}}^T H(\mathbf{q})\dot{\mathbf{q}}}{\partial \mathbf{q}}$ is very complex to be calculated. Analytical solutions, using the so called Christoffel symbols lead to accurate solution but very inefficient for real time computing.

The Power (Spong, 2005) of an overall constrained mechanical system, expressed with Lagrangian approach is

$$P = \langle \dot{\mathbf{q}}, \tau \rangle = \dot{\mathbf{q}}^T \tau = \dot{K} + \dot{U} \tag{15}$$

III. Body Decomposition Modeling

The Body Decomposition Approach given in (Olguin, 2011) is based in the principle that summation of the Power of every single rigid body (9) in a multi-body system (15) must be the same, or equivalently that the mechanical *Work* performed by a mechanical structure of N rigid elements and shall be expressed as the total summation of the individual *Work* contribution. Then the dynamic modeling of a multi-body mechanical system is given by the following expression:

$$\tau \;=\; \sum_{i=1}^{N} J_i^T \mathbf{F}_i(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \tag{16}$$

where the Jacobians $J_i(\mathbf{q})$ are *local Jacobians* that transform the generalized velocities coordinates $\dot{\mathbf{q}}$ to the local frame coordinates twist:

$$\nu_i^{(i)} \;=\; J_i(\mathbf{q})\dot{\mathbf{q}} \tag{17}$$

$$\dot{\nu}_i^{(i)} \;=\; J_i(\mathbf{q})\ddot{\mathbf{q}} + \dot{J}_i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \tag{18}$$

and the wrenches $\mathbf{F_i}$ are the active wrenches on each single body applied at the same reference point of the given body being reciprocal to the corresponding twist $\nu_\mathbf{i}$.

Using the equation of motion (8) for rigid bodies and the *local* cinematic expressions (17)-(18) in the general BDA expression (16), it yields:

$$\begin{aligned} \tau &= \underbrace{\sum_{i=1}^{N}\left[J_i^T M_i J_i\right]\ddot{q}}_{H(q)} + \underbrace{\sum_{i=1}^{N}\left[J_i^T M_i \dot{J}_i - J_i^T S(M_i J_i \dot{q}) J_i\right]\dot{q}}_{C(q,\dot{q})} \\ &\quad \underbrace{-\sum_{i=1}^{N}\left[J_i^T M_i \mathscr{R}_i^T\right]\mathbf{G}_0}_{\mathbf{g}(q)} \underbrace{-\sum_{i=1}^{N}\left(J_i^T \mathbf{F}_{D_i}\right)}_{D(\cdot)\dot{\mathbf{q}}} \end{aligned} \tag{19}$$

which is equivalent to explicit Lagrangian expression (14) plus the dissipative terms -which can also be included using the Virtual Power principle. Notice that the Coriolis matrix becomes extremely simple and fulfils the basic Lagrange/passsive property: $2C - \dot{H} \in SS(n)$ (Olguin, 2011). Then the Christoffel which are calculated with the partial derivative of each element of H is not longer need, but instead the time derivative of each local jacobian $\dot{J}_i$ is required.

III-A. *Recursive BDA*

Open kinematic chains allow a recursive calculation of all the terms in the BDA expression, which simplifies the numerical evaluation of each term. This recursion is

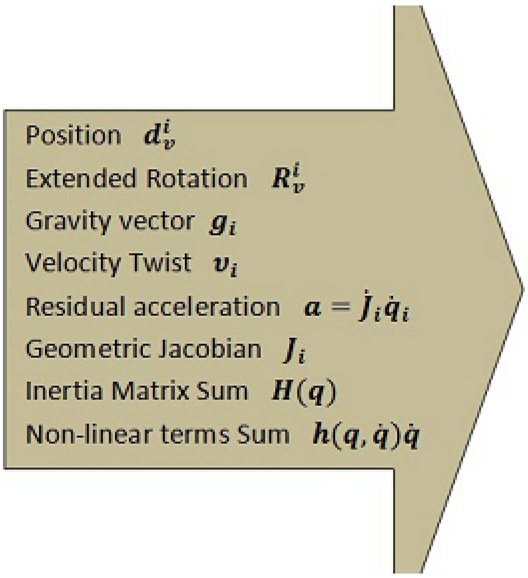

Figura 2: Terms obtained from the BDA Block

based on the same recursion used in the efficient inverse dynamic methods proposed in (Featherstone, 2000). While the direct kinematics expression for these methods are based in the modified Denavit-Hartenberg convention, in this work however, we have adapted this recursion to the original D-H convention, resulting in the following twist recursive basic expression:

$$v_i^{(i)} = \mathscr{X}_i(\mathbf{d}_i, R_i)\left(v_{i-1}^{(i-1)} + \lambda_i \dot{q}_i\right) \tag{20}$$

where $\mathscr{X}_i(\mathbf{d}_i, R_i) = \mathscr{X}_i(q_i)$ is the relative kinematic operator from a father frame Σ_{i-1} to his son frame Σ_i, whose arguments $\mathbf{d}_i(q_i) \in I\!R^3$ and $R_i(q_i) \in SO(3)$ are the relative distance and relative rotation matrix resulting from the homogeneous transformation from frame Σ_{i-1} to Σ_i; and $\lambda_i \in I\!R^6$ is a joint director screw, constant in the father frame, defined as

$$\lambda_i = \lambda_i^{i-1} \triangleq \begin{pmatrix} \lambda_{T_i} \\ \lambda_{R_i} \end{pmatrix} \tag{21}$$

where vectors $(\lambda_{T_i}, \lambda_{R_i}) \in I\!R^3$ stands for the 3D unit direction vectors of either pure translation or pure rotation of link i. It is worth noticing that the term $\lambda_i \dot{q}_i = v_{i/i-1}^{(i-1)}$ stands for the relative velocity between a frame Σ_i and his father Σ_{i-1} via the relative motion of the associated generalized coordinate. Then from (6) it follows that

$$\dot{\mathscr{X}}_i(\mathbf{d}_i, R_i) = -\mathscr{X}_i(\mathbf{d}_i, R_i)\Omega(\lambda_i \dot{q}_i)$$

Comparing (17) with (20) it follows that the local Jacobian used in the BDA can be expressed in a recursive formulation as

$$J_i = \mathscr{X}_i(q_i)\left[J_{i-1}(q_1 \cdots qi-1) + \Lambda_i\right] \quad \in I\!R^{6\times n} \tag{22}$$

where Λ_i is a constant matrix with null elements except for the i^{th}-column which is indeed the corresponding director vector λ_i:

$$\Lambda_i \triangleq \begin{array}{c} 1,\ldots,i,\ldots,n \\ \left[0,\ldots,\lambda_i,\ldots,0\right] \end{array} \in I\!R^{6\times n}$$

Then it follows that the time derivative of each local Jacobian becomes

$$\dot{J}_i = \mathscr{X}_i(q_i)\left[\dot{J}_{i-1} - \Omega(\lambda_i \dot{q}_i) J_{i-1}\right]$$

However, in the recursive methodology, it result more advantageous to recursively calculate the term $\dot{J}_i\dot{\mathbf{q}}$ instead. If this is defined as a *residual acceleration*, its recursive equation follows straightforward as:

$$\mathbf{a}_i \triangleq \dot{J}_i \dot{\mathbf{q}} = \mathscr{X}_i(q_i)\left(\mathbf{a}_{i-1} + \Omega(v_{i-1})\lambda_i \dot{q}_i\right)$$

Then the i^{th} generalized force contribution in (19) due to the body i becomes $\tau_i = H_i \ddot{q} + \mathbf{h}_i(\mathbf{q}, \dot{\mathbf{q}})$ where

$$\begin{aligned} H_i &= J_i^T M_i J_i \\ \mathbf{h}_i(\mathbf{q}, \dot{\mathbf{q}}) &= J_i^T\left(M_i \mathbf{a}_i - S(M_i v_i^{(i)}) v_i^{(i)} - m_i \mathbf{G}_i - \mathbf{F}_{D_i}\right) \end{aligned}$$

Which are the terms that can be evaluated recursively as shown in Fig.3.

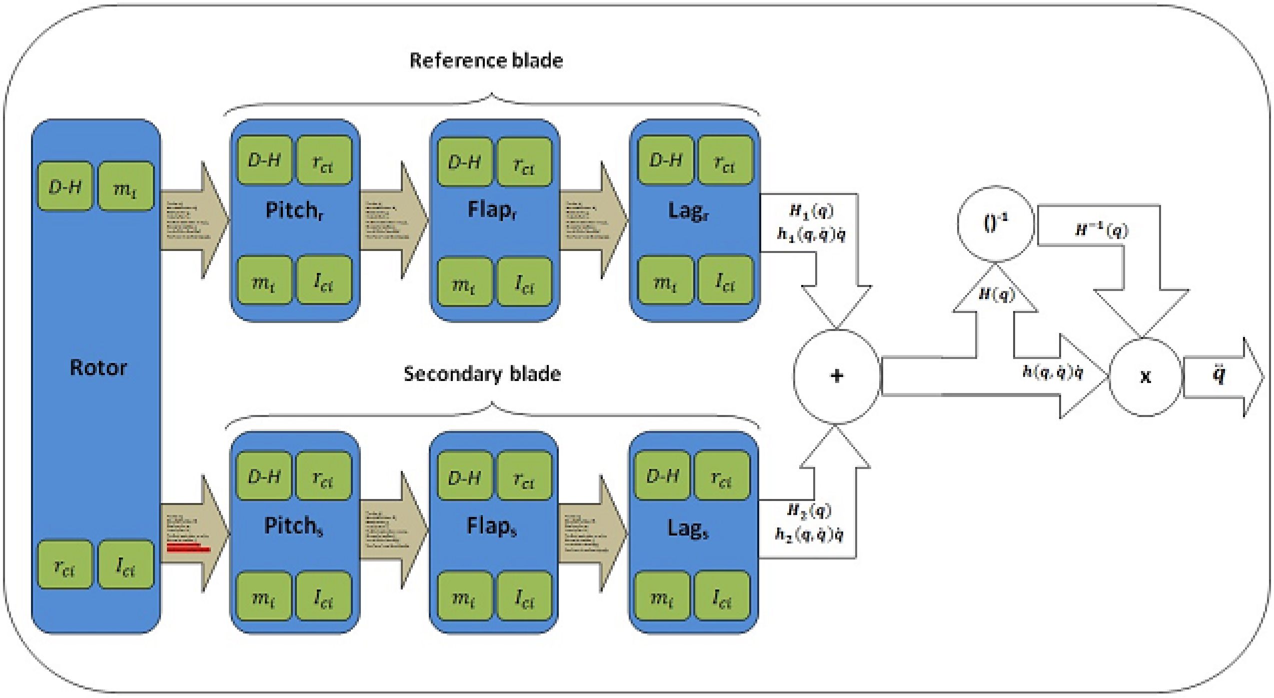

Figura 3: Main Rotor BDA Blockset

IV. Helicopter Modeling

The body of the helicopter is modeled as a rigid body whose input is a *wrench* provided by the aerodynamics of the main and tail rotors and its output is the helicopter *pose*. The aerodynamic forces produced in the main rotor depend on the dynamic behavior of the blades, which depends itself on the relative position of each blade with respect to the helicopter and its relative velocity with respect to the wind. In order to keep the following analysis as simple as possible, the tail rotor has been neglected. Then the vector of generalized coordinates is the following:

$$\mathbf{q} = \begin{pmatrix} \dot{\mathbf{x}}_v \\ \\ \dot{\mathbf{q}}_n \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \psi \\ \hline \Psi \\ \theta_r \\ \beta_r \\ \gamma_r \\ \theta_s \\ \beta_s \\ \gamma_s \end{pmatrix} = \begin{pmatrix} longitude \\ latitude \\ altitude \\ roll \\ pitch \\ yaw \\ \hline rotor\ azimuth \\ ref\ blade\ pitch \\ ref\ blade\ flap \\ ref\ blade\ lag \\ sec\ blade\ pitch \\ sec\ blade\ flap \\ sec\ blade\ lag \end{pmatrix} \in I\!R^{13}$$

The dynamic model of the helicopter is obtained using the *BDA* formulation in the same way as it is used to analyse articular robots, the analysis using *Euler-Lagrange* formulation can be seen more detailed in (Dorado y Olguin, 2010).

In the case of the X-cell 90, whose main rotor has 2 blades in a tree-like open kinematic structure the blocks describing the rotor model can be seen in Fig.3.

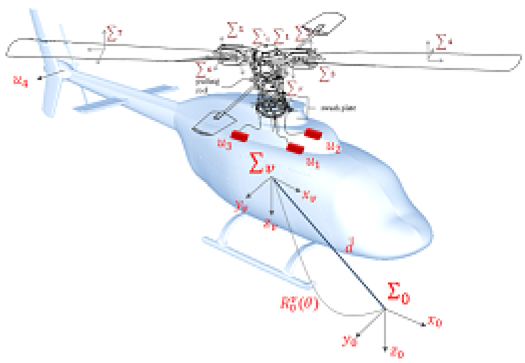

Figura 4: Helicopter Scheme Model

Every blue block represents a single free flying rigid body dynamic, having constant kinematic and dynamic parameters (small green blocks) that can be added externally to make the appropriate estimation if it is required. The brown arrow (see Fig.2) heritage both kinematic and dynamic variables in a parent to child relationship. An exception is found only in the inertia matrix and non-linear dynamic terms at the beginning of the secondary blade chain in order to avoid inertial duplication in the final summation.

To include the aerodynamic forces in (19), they are considered as lumped forces at the equivalent *lag pressure points* on each blade (Dorado y Olguin, 2010). On the other hand, the rotor wind induced velocity v_i is considered quasi-

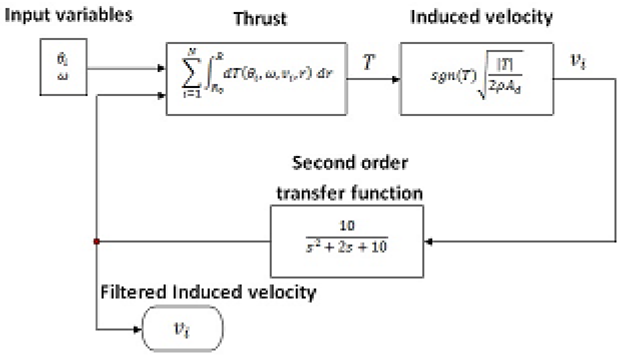

Figura 5: Induced velocity calculation using a second order transfer function

constant because its dynamics is slower than the mechanics of the rotor. Then an off-line evaluation would give sufficiently good performance (Morten, 2010). In consequence, we have proposed the following modification shown in Fig.5: 1) the aerodynamic thrust (and drag) produced by each blade is evaluated with a a-priori given induced velocity $v_i(t)$ using the blade element theory, 2) a new estimated induced velocity $\hat{v}_i(t)$ is calculated using the air momentum theory, and 3) a delayed 2^{nd}-order filtered induced velocity $v_i(t+D)$ is used to evaluate again the thrust and drag forces on each blade.

The operations can be done using most of the simulation software in the market. In this case we have used Simulink from MathWorks where we have created a generalized non-inertial dynamic block used to simulate each one of the bodies in the articulated chain shown in Fig.4. It seems important to remark that recursive geometric Jacobians, evaluated from (22) depend on the proper evaluation of the corresponding Λ_i matrices. Then proper use of this operators is essential for the procedure to work properly.

V. Conclusions

The mechanic complexity of a non-inertial multi-body system can be modeled using the **BDA** formulation and further it can be implemented on a common simulation environment such as Matlab Simulink, obtaining a very efficient simulator because of the computation time and the simple implementation of the operators. In addition the complexity of both the Coriolis term and the non-linear aerodynamics are solved by a proper matrix transformation of each of these terms at each individual body element in the system, and thanks to the recursive property in open kinematic chains. The modularity of the methodology allows the construction of simple yet efficient simulators which allows the simplification of the induced wind velocity calculation. Each body can be expressed in a general form using blocks to represent each operation, having as inputs the required constant kinematic parameters given by the Denavit-Hartenberg convention and dynamic parameters. Simulation time has been reduced substantially from classic Euler-Lagrange formulation, then advanced programming method with a C or C++ languages are promising.

Referencias

Dickerson, L., *UAVs on the Rise*, Aviation Week & Space Technology, January 15, 2007.

Featherstone, R. and Orin. D., *Robot dynamics: Equations and algorithms*. In IEEE International Conference on Robotics and Automation, pages 826-834, San Francisco, CA, April 2000.

Goldstein, H., *Classical Mechanics*, 2nd edition, Addison-Wesley, July 1980.

Méndez Dorado, Mario and Olguín-Díaz, Ernesto, *Lagrangian Aerodynamic Modeling of an Articulated Rotor for UAVs*, *in XII COMROB Congreso Mexicano de Robótica*, Mazatlan, Sinaloa, 2010.

Olguín Díaz, Ernesto, *A General Body Decomposition Solution for the Dynamic Modeling of Multi-body Systems: The Body Decomposition Approach*, Internal report. Cinvestav, Unidad Saltillo, México, 2011.

Pir, Zubair, S., *Pakistan Says U.S. Drone Kills 13*, New York Times, June 18, 2009.

Shim, D. H, Kim, H. J., Sastry, S., *Hierarchical Control System Synthesis for Rotorcraft-based Unmanned Aerial Vehicles*, AIAA Guidance, Navigation and Control Conference, Denver, CO, 2000.

Spong, Mark W., Hutchinson Seth and Vidyasagar M., *Robot Modeling and Control*, First Edition, John Wiley & Sons, Inc., 2005.

Morten Bisgaard and Anders la Cour-Harbo and Jan Dimon Bendtsen, *Adaptive Control System for Autonomous Helicopter Slung Load Operations* Control Engineering Practice, Vol 18, N. 7 July 2010, ISSN: 0967-0661, pp 800-811.