Development of a Mathematical Model and Simulation Environment for the Postural Robot (PostuRob II)

Prepared by

Mohammad AlBakri, B. Eng.

Department of Mechanical Engineering

Palestine Polytechnic University

Hebron - Palestine

02. June 2008

Table of Contents

1. 1 Introduction

To understand and investigate human postural control, a specially designed robot, PostuRob II, is constructed at the Neurocenter of the clinics of Freiburg University. Biological control concepts can be embodied in this robot so that it serves in establishing a hardware-in-the-loop simulation environment. MATLAB and Simulink can then be used to interact with the robot through x-PC target modality. Furthermore, the robot can serve as a test bed for alternative technical control approached.

The robot resembles a simplified version of the human body such that it consists of three rigid bodies: a foot, a leg and a torso. The robot stands freely on a movable Stewart-like platform which enables for small rotations and translations in the three dimensional space. However, since the robot is constructed in such a way that its links can move (rotate) only in the sagittal plane, only one rotation and one translation of the platform become relevant.

PostuRob II is equipped with pneumatic actuators arranged in pairs to generate necessary torque to actuate the joints. The first set of these actuators is inserted between the foot and the leg to cause a relative rotation of the leg with respect to the foot while the second set is inserted between the leg and the torso to cause a relative motion of the torso with respect to the leg.

A set of sensors are used to capture information about the state of the robot and to be used for the control loop. First, the foot is supplied with two normal reaction force sensors for measuring the reaction forces between the foot and the platform. These sensors are located at the frontal and rear parts of the foot. The force measurements provide information about the Center of Pressure (COP) shift. Second, the relative motion of the three rigid bodies is measured by a set of three potentiometers (for measuring the relative angles) and three tachometers (for measuring the relative velocities). Third, an artificial vestibular system composed of an accelerometer and a gyrometer provides information about the translation acceleration (two components) and angular velocity of the head.

1.2 Lagrange's Approach

Lagrange's approach is used to derive the nonlinear equations that describe the motion of PostuRob II at each of its generalized coordinates. The basic formula of Lagrange's equation is:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_n}\right)-\frac{\partial L}{\partial q_n}=Q_n \tag{1}$$

Where:

- L: the lagrangian term, such that:

$$L = T - U$$

- T: the total kinetic energy of the system.
- U: the total potential energy of the system.
- Q_n: Forces and torques that act in each coordinate including the non-conservative forces due to Coulomb friction and viscous damping.
- $q_1 \ldots q_n$: the generalized coordinates that describe system's motion.
  Where a set of (n) generalized coordinates are needed to describe the motion of an n-degree-of-freedom system.

In the upcoming section this methodology will be followed to derive the mathematical model for the postural robot "PostuRob II".

1.3 Model Derivation

The mathematical model of the system consists of four second-order nonlinear differential equations, these equations are derived using Lagrange's approach explained earlier. The model is first derived based on the absolute definition of the angles, and then the relative equivalent of those absolute angles is substituted. Based on Fig. 1, the total kinetic and potential energies of the system can be found as follows:

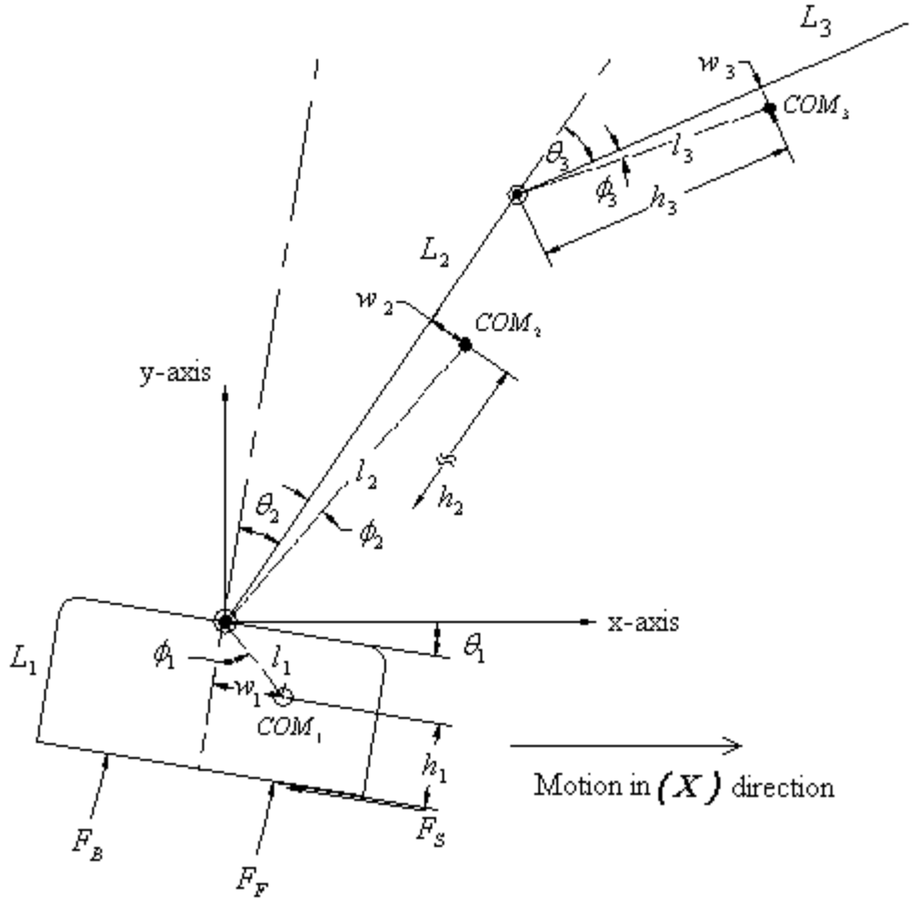

Figure 1: Free-Body Diagram of PostuRob II . The geometrical parameters are

$l_1 = \sqrt{(L_1 - h_1)^2 + w_1^2}$	$l_2 = \sqrt{h_2^2 + w_2^2}$	$l_3 = \sqrt{h_3^2 + w_3^2}$
$\phi_1 = \tan^{-1}\left(w_1 / (L_1 - h_1) \right)$	$\phi_2 = \tan^{-1}\left(w_2 / h_2 \right)$	$\phi_3 = \tan^{-1}\left(w_3 / h_3 \right)$

For simplicity, the Lagrnage's approach is first applied to the system using absolute angles, that is link angles measured with respect to the absolute vertical. Later on, the relative angles, that is link angles measured with respect to each other, are substituted in the obtained equations to obtain the desired version of the dynamic model. The relation between absolute (θ_{1a} , θ_{2a} and θ_{3a}) and relative angles θ_2 and θ_3) is as follows:

$$\theta_{1a} = \theta_1 \tag{2}$$

$$\theta_{2a} = \theta_1 + \theta_2 \tag{3}$$

$$\theta_{3a} = \theta_1 + \theta_2 + \theta_3 \tag{4}$$

Then, in the final results, each absolute angle is substituted by its relative equivalents according to the equations above.

The total kinetic energy of the system is the sum of those energies for each of the system components, i.e. kinetic energy for the foot, in addition to that for the leg and torso.

$$T = T_{foot} + T_{leg} + T_{torso} \tag{5}$$

In more details:

$$T = \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} m_1 v_1^2 + \frac{1}{2} I_2 \dot{\theta}_{2a}^2 + \frac{1}{2} m_2 v_2^2 + \frac{1}{2} I_3 \dot{\theta}_{3a}^2 + \frac{1}{2} m_3 v_3^2 \tag{6}$$

As Eq. (6) indicates, the kinetic energy of each link (the foot, leg and torso) is due to its center of gravity translational and angular velocities. That translational velocity consists of two components, horizontal and vertical. These components are derived referring to Fig. 2 as follows:

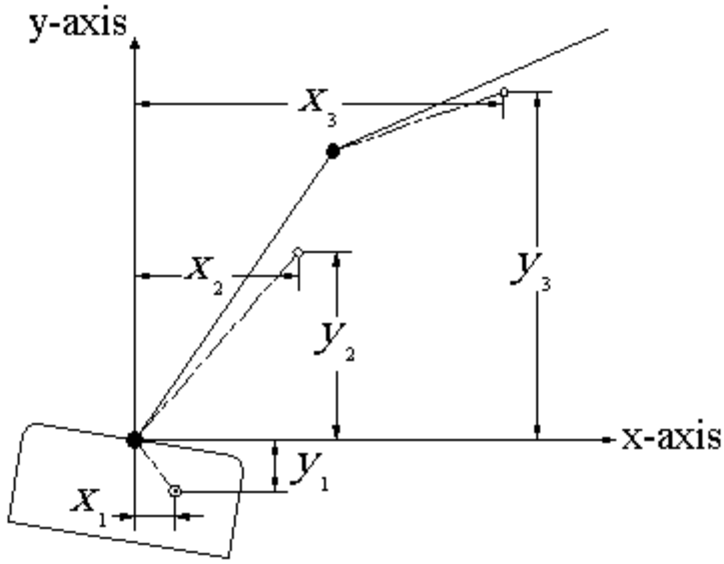

Figure 2 Center of mass displacement for each link

For the foot:

$$x_1 = x + l_1 \sin(\phi_1 - \theta_1) \tag{7}$$

$$y_1 = -l_1 \cos(\phi_1 - \theta_1) \tag{8}$$

Thus:

$$v_1^2 = \dot{x}_1^2 + \dot{y}_1^2 = \dot{x}^2 - 2\dot{x}\,\dot{\theta}_1 l_1 \cos(\phi_1 - \theta_1) + l_1^2\dot{\theta}_1^2 \tag{9}$$

For the leg:

$$x_2 = x + l_2 \sin(\theta_{2a} + \phi_2) \tag{10}$$

$$y_2 = l_2 \cos(\theta_{2a} + \phi_2) \tag{11}$$

Thus:

$$v_2^2 = \dot{x}_2^2 + \dot{y}_2^2 = \dot{x}^2 + 2\dot{x}\,\dot{\theta}_{2a} l_2 \cos(\theta_{2a} + \phi_2) + l_2^2\dot{\theta}_{2a}^2 \tag{12}$$

For the torso:

$$x_3 = x + L_2 \sin(\theta_{2a}) + l_3 \sin(\theta_{3a} + \phi_3) \tag{13}$$

$$y_3 = L_2 \cos(\theta_{2a}) + l_3 \cos(\theta_{3a} + \phi_3) \tag{14}$$

Thus:

$$\begin{aligned} v_3^2 = \dot{x}_3^2 + \dot{y}_3^2 = \dot{x}^2 + L_2^2\dot{\theta}_{2a}^2 + l_3^2\dot{\theta}_{3a}^2 + 2\dot{x}\,\dot{\theta}_{2a} L_2 \cos(\theta_{2a}) \\ + 2\dot{x}\,\dot{\theta}_{3a} l_3 \cos(\theta_{3a} + \phi_3) + 2\dot{\theta}_{2a}\dot{\theta}_{3a} L_2 l_3 \cos(\theta_{2a} - \theta_{3a} - \phi_3) \end{aligned} \tag{15}$$

The total potential energy (U) of the system is due to each link's center of mass elevation, thus (U) is given by:

$$\begin{aligned} U &= M_1 y_1 + M_2 y_2 + M_3 y_3 \\ &= -M_1 l_1 \cos(\phi_1 - \theta_1) + M_2 l_2 \cos(\theta_{2a} + \phi_2) + M_3\left(L_2 \cos(\theta_{2a}) + l_3 \cos(\theta_{3a} + \phi_3)\right) \end{aligned} \tag{16}$$

The Lagrangian (L) is defined as follows:

$$\begin{aligned} L &= T - U \\ &= \frac{1}{2} I_1\dot{\theta}_1^2 + \frac{1}{2} m_1\left(\dot{x}^2 - 2\dot{x}\,\dot{\theta}_1 l_1 \cos(\phi_1 - \theta_1) + l_1^2\dot{\theta}_1^2\right) + \frac{1}{2} I_2\dot{\theta}_{2a}^2 + \\ &\quad \frac{1}{2} m_2\left(\dot{x}^2 + 2\dot{x}\,\dot{\theta}_{2a} l_2 \cos(\theta_{2a} + \phi_2) + l_2^2\dot{\theta}_{2a}^2\right) + \frac{1}{2} I_3\dot{\theta}_{3a}^2 + \\ &\quad \frac{1}{2} m_3\begin{pmatrix} \dot{x}^2 + L_2^2\dot{\theta}_{2a}^2 + l_3^2\dot{\theta}_{3a}^2 + 2\dot{x}\,\dot{\theta}_{2a} L_2 \cos(\theta_{2a}) \\ +2\dot{x}\,\dot{\theta}_{3a} l_3 \cos(\theta_{3a} + \phi_3) + 2\dot{\theta}_{2a}\dot{\theta}_{3a} L_2 l_3 \cos(\theta_{2a} - \theta_{3a} - \phi_3) \end{pmatrix} + \\ &\quad M_1 l_1 \cos(\phi_1 - \theta_1) - M_2 l_2 \cos(\theta_{2a} + \phi_2) - M_3\left[L_2 \cos(\theta_{2a}) + l_3 \cos(\theta_{3a} + \phi_3)\right] \end{aligned} \tag{17}$$

Applying Lagrange's equation for each generalized coordinate, x , θ_1, θ_{2a} and θ_{3a} , yields:

1. In (x) direction:

The Lagrange's equation in x-direction is given as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right)-\frac{\partial L}{\partial x}=Q_x \tag{18}$$

Where:

$$\frac{\partial L}{\partial \dot{x}}=(m_1+m_2+m_3)\dot{x}-m_1 l_1\cos(\phi_1-\theta_1)\dot{\theta}_1+\left[m_2 l_2\cos(\theta_{2a}+\phi_2)+m_3 L_2\cos(\theta_{2a})\right]\dot{\theta}_{2a} \\ +m_3 l_3\cos(\theta_{3a}+\phi_3)\dot{\theta}_{3a} \tag{19}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right)=(m_1+m_2+m_3)\ddot{x}-m_1 l_1\cos(\phi_1-\theta_1)\ddot{\theta}_1-m_1 l_1\sin(\phi_1-\theta_1)\dot{\theta}_1^2 \\ +\left[m_2 l_2\cos(\theta_{2a}+\phi_2)+m_3 L_2\cos(\theta_{2a})\right]\ddot{\theta}_{2a}-\left[m_2 l_2\sin(\theta_{2a}+\phi_2)+m_3 L_2\sin(\theta_{2a})\right]\dot{\theta}_{2a}^2 \\ +m_3 l_3\cos(\theta_{3a}+\phi_3)\ddot{\theta}_{3a}-m_3 l_3\sin(\theta_{3a}+\phi_3)\dot{\theta}_{3a}^2 \tag{20}$$

$$\frac{\partial L}{\partial x}=0 \tag{21}$$

$$Q_x=F_e-F_S\cos(\theta_1)+F_F\sin(\theta_1)+F_B\sin(\theta_1) \tag{22}$$

Applying Eq. (18) yields:

$$(m_1+m_2+m_3)\ddot{x}-m_1 l_1\cos(\phi_1-\theta_1)\ddot{\theta}_1+\left[m_2 l_2\cos(\theta_{2a}+\phi_2)+m_3 L_2\cos(\theta_{2a})\right]\ddot{\theta}_{2a} \\ +m_3 l_3\cos(\theta_{3a}+\phi_3)\ddot{\theta}_{3a}-m_1 l_1\sin(\phi_1-\theta_1)\dot{\theta}_1^2-\left[m_2 l_2\sin(\theta_{2a}+\phi_2)+m_3 L_2\sin(\theta_{2a})\right]\dot{\theta}_{2a}^2 \\ -m_3 l_3\sin(\theta_{3a}+\phi_3)\dot{\theta}_{3a}^2=F_e-F_S\cos(\theta_1)+\left(F_F+F_B\right)\sin(\theta_1) \tag{23}$$

2. In θ_1 direction:

The Lagrange's equation in θ_1 direction is given as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_1}\right)-\frac{\partial L}{\partial \theta_1}=Q_{\theta_1} \tag{24}$$

Where:

$$\frac{\partial L}{\partial \dot{\theta}_1}=-m_1 l_1\cos(\phi_1-\theta_1)\dot{x}+\left(I_1+m_1 l_1^2\right)\dot{\theta}_1 \tag{25}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_1}\right)=-m_1 l_1\cos(\phi_1-\theta_1)\ddot{x}+\left(I_1+m_1 l_1^2\right)\ddot{\theta}_1-m_1 l_1\sin(\phi_1-\theta_1)\dot{x}\,\dot{\theta}_1 \tag{26}$$

$$\frac{\partial L}{\partial \theta_1}=-m_1 l_1\sin(\phi_1-\theta_1)\dot{x}\,\dot{\theta}_1+M_1 l_1\sin(\phi_1-\theta_1) \tag{27}$$

$$Q_{\theta_1}=F_S L_1+F_B D_B-F_F D_F+\left(F_{lF}-F_{lB}\right)D_{la} \tag{28}$$

Applying Eq. (24) yields:

$$-m_1 l_1\cos(\phi_1-\theta_1)\ddot{x}+\left(I_1+m_1 l_1^2\right)\ddot{\theta}_1-M_1 l_1\sin(\phi_1-\theta_1)=F_S L_1+F_B D_B-F_F D_F+\left(F_{lF}-F_{lB}\right)D_{la} \tag{29}$$

3. In θ_2 direction:

The Lagrange's equation in θ_2 direction is given as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{2a}}\right)-\frac{\partial L}{\partial \theta_{2a}}=Q_{\theta_{2a}} \tag{30}$$

Where:

$$\frac{\partial L}{\partial \dot{\theta}_{2a}}=[m_2 l_2\cos(\theta_{2a}+\phi_2)+m_3 L_2\cos(\theta_{2a})]\dot{x}+\left(I_2+m_2 l_2^2+m_3 L_2^2\right)\dot{\theta}_{2a}+m_3 L_2 l_3\cos(\theta_{2a}-\theta_{3a}-\phi_3)\dot{\theta}_{3a}$$

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{2a}}\right)=&[m_2 l_2\cos(\theta_{2a}+\phi_2)+m_3 L_2\cos(\theta_{2a})]\ddot{x}+\left(I_2+m_2 l_2^2+m_3 L_2^2\right)\ddot{\theta}_{2a}\\&+m_3 L_2 l_3\cos(\theta_{2a}-\theta_{3a}-\phi_3)\ddot{\theta}_{3a}+m_3 L_2 l_3\sin(\theta_{2a}-\theta_{3a}-\phi_3)\dot{\theta}_{3a}^2\\&-[m_2 l_2\sin(\theta_{2a}+\phi_2)+m_3 L_2\sin(\theta_{2a})]\dot{x}\,\dot{\theta}_{2a}-m_3 L_2 l_3\sin(\theta_{2a}-\theta_{3a}-\phi_3)\dot{\theta}_{2a}\dot{\theta}_{3a}\end{aligned}$$

$$\begin{aligned}\frac{\partial L}{\partial \theta_{2a}}=&-[m_2 l_2\sin(\theta_{2a}+\phi_2)+m_3 L_2\sin(\theta_{2a})]\dot{x}\,\dot{\theta}_{2a}-m_3 L_2 l_3\sin(\theta_{2a}-\theta_{3a}-\phi_3)\dot{\theta}_{2a}\dot{\theta}_{3a}\\&+M_2 l_2\sin(\theta_{2a}+\phi_2)+M_3 L_2\sin(\theta_{2a})\end{aligned}$$

$$Q_{\theta_{2a}}=F_e L_2\cos(\theta_{2a})+\left(F_{lB}-F_{lF}\right)D_{la}+\left(F_{tF}-F_{tB}\right)D_{ta}$$

Now if we define for simplicity

$$T_l=D_{la}\left(F_{lB}-F_{lF}\right) \tag{31}$$

$$T_t=D_{ta}\left(F_{tB}-F_{tF}\right) \tag{32}$$

Then:

$$Q_{\theta_{2a}}=F_e L_2\cos(\theta_{2a})+T_l-T_t \tag{33}$$

Applying Eq. (30) yields:

$$\begin{aligned}&[m_2 l_2\cos(\theta_{2a}+\phi_2)+m_3 L_2\cos(\theta_{2a})]\ddot{x}+\left(I_2+m_2 l_2^2+m_3 L_2^2\right)\ddot{\theta}_{2a}\\&+m_3 L_2 l_3\cos(\theta_{2a}-\theta_{3a}-\phi_3)\ddot{\theta}_{3a}+m_3 L_2 l_3\sin(\theta_{2a}-\theta_{3a}-\phi_3)\dot{\theta}_{3a}^2\\&-M_2 l_2\sin(\theta_{2a}+\phi_2)-M_3 L_2\sin(\theta_{2a})=F_e L_2\cos(\theta_{2a})+T_l-T_t\end{aligned} \tag{34}$$

4. In θ_3 direction:

The Lagrange's equation in θ_3 direction is given as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{3a}}\right)-\frac{\partial L}{\partial \theta_{3a}}=Q_{\theta_{3a}} \tag{35}$$

Where:

$$\frac{\partial L}{\partial \dot{\theta}_{3a}} = m_3 l_3 \cos(\theta_{3a} + \phi_3)\dot{x} + m_3 L_2 l_3 \cos(\theta_{2a} - \theta_{3a} - \phi_3)\dot{\theta}_{2a} + \left(I_3 + m_3 l_3^2\right)\dot{\theta}_{3a} \tag{36}$$

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{3a}}\right) &= m_3 l_3 \cos(\theta_{3a} + \phi_3)\ddot{x} + m_3 L_2 l_3 \cos(\theta_{2a} - \theta_{3a} - \phi_3)\ddot{\theta}_{2a} + \left(I_3 + m_3 l_3^2\right)\ddot{\theta}_{3a} \\ &- m_3 L_2 l_3 \sin(\theta_{2a} - \theta_{3a} - \phi_3)\dot{\theta}_{2a}^2 - m_3 l_3 \sin(\theta_{3a} + \phi_3)\dot{x}\,\dot{\theta}_{3a} + m_3 L_2 l_3 \sin(\theta_{2a} - \theta_{3a} - \phi_3)\dot{\theta}_{2a}\dot{\theta}_{3a}\end{aligned} \tag{37}$$

$$\frac{\partial L}{\partial \theta_{3a}} = -m_3 l_3 \sin(\theta_{3a} + \phi_3)\dot{x}\,\dot{\theta}_{3a} + m_3 L_2 l_3 \sin(\theta_{2a} - \theta_{3a} - \phi_3)\dot{\theta}_{2a}\dot{\theta}_{3a} + M_3 l_3 \sin(\theta_{3a} + \phi_3) \tag{38}$$

$$Q_{\theta_{3a}} = F_e L_e \cos(\theta_{3a}) + T_t \tag{39}$$

Applying Eq. (35) yields:

$$\begin{aligned}&m_3 l_3 \cos(\theta_{3a} + \phi_3)\ddot{x} + m_3 L_2 l_3 \cos(\theta_{2a} - \theta_{3a} - \phi_3)\ddot{\theta}_{2a} + \left(I_3 + m_3 l_3^2\right)\ddot{\theta}_{3a} \\ &- m_3 L_2 l_3 \sin(\theta_{2a} - \theta_{3a} - \phi_3)\dot{\theta}_{2a}^2 - M_3 l_3 \sin(\theta_{3a} + \phi_3) = F_e L_e \cos(\theta_{3a}) + T_t\end{aligned} \tag{40}$$

Equations (34) and (40) can be written in a matrix form leading to the following dynamics model:

$$\begin{aligned}&\begin{bmatrix} \left(I_2 + m_2 l_2^2 + m_3 L_2^2\right) & m_3 L_2 l_3 \cos(\theta_{2a} - \theta_{3a} - \phi_3) \\ m_3 L_2 l_3 \cos(\theta_{2a} - \theta_{3a} - \phi_3) & \left(I_3 + m_3 l_3^2\right) \end{bmatrix} \begin{bmatrix} \ddot{\theta}_{2a} \\ \ddot{\theta}_{3a} \end{bmatrix} + \\ &\begin{bmatrix} 0 & m_3 L_2 l_3 \sin(\theta_{2a} - \theta_{3a} - \phi_3) \\ -m_3 L_2 l_3 \sin(\theta_{2a} - \theta_{3a} - \phi_3) & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_{2a}^2 \\ \dot{\theta}_{3a}^2 \end{bmatrix} + \begin{bmatrix} -M_2 l_2 \sin(\theta_{2a} + \phi_2) - M_3 L_2 \sin(\theta_{2a}) \\ -M_3 l_3 \sin(\theta_{3a} + \phi_3) \end{bmatrix} \\ &= \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} T_l \\ T_t \end{bmatrix} - \begin{bmatrix} \left[m_2 l_2 \cos(\theta_{2a} + \phi_2) + m_3 L_2 \cos(\theta_{2a})\right] & 0 & -L_2 \cos(\theta_{2a}) \\ m_3 l_3 \cos(\theta_{3a} + \phi_3) & 0 & -L_e \cos(\theta_{3a}) \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ F_e \end{bmatrix}\end{aligned} \tag{41}$$

However, if one is interested in the dynamics model given the relative angles instead of the absolute ones, then substituting Eqs. (2-4) into Eq. (41) yields:

$$\begin{aligned}&\begin{bmatrix} I_2 + m_2 l_2^2 + m_3 L_2^2 + m_3 L_2 l_3 \cos(\theta_3 + \phi_3) & m_3 L_2 l_3 \cos(\theta_3 + \phi_3) \\ m_3 L_2 l_3 \cos(\theta_3 + \phi_3) + I_3 + m_3 l_3^2 & \left(I_3 + m_3 l_3^2\right) \end{bmatrix} \begin{bmatrix} \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \\ &\begin{bmatrix} -m_3 L_2 l_3 \sin(\theta_3 + \phi_3) & -m_3 L_2 l_3 \sin(\theta_3 + \phi_3) \\ m_3 L_2 l_3 \sin(\theta_3 + \phi_3) & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix} + \begin{bmatrix} -M_2 l_2 \sin(\theta_1 + \theta_2 + \phi_2) - M_3 L_2 \sin(\theta_1 + \theta_2) \\ -M_3 l_3 \sin(\theta_1 + \theta_2 + \theta_3 + \phi_3) \end{bmatrix} \\ &= \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} T_l \\ T_t \end{bmatrix} + \begin{bmatrix} m_3 L_2 l_3 \sin(\theta_3 + \phi_3)\left(\dot{\theta}_1^2 + 2\dot{\theta}_1\dot{\theta}_2 + 2\dot{\theta}_1\dot{\theta}_3 + 2\dot{\theta}_2\dot{\theta}_3\right) \\ -m_3 L_2 l_3 \sin(\theta_3 + \phi_3)\left(\dot{\theta}_1^2 + 2\dot{\theta}_1\dot{\theta}_2\right) \end{bmatrix} \\ &- \begin{bmatrix} \left[m_2 l_2 \cos(\theta_1 + \theta_2 + \phi_2) + m_3 L_2 \cos(\theta_1 + \theta_2)\right] & I_2 + m_2 l_2^2 + m_3 L_2^2 + m_3 L_2 l_3 \cos(\theta_3 + \phi_3) & -L_2 \cos(\theta_1 + \theta_2) \\ m_3 l_3 \cos(\theta_1 + \theta_2 + \theta_3 + \phi_3) & m_3 L_2 l_3 \cos(\theta_3 + \phi_3) + I_3 + m_3 l_3^2 & -L_e \cos(\theta_1 + \theta_2 + \theta_3) \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ F_e \end{bmatrix}\end{aligned} \tag{42}$$

To avoid the coupling between the two input torques and to achieve symmetry in the inertia matrix, the above equation can be multiplied by the inverse of the torque matrix, i.e.:

$$\begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

which yields:

$$\begin{aligned}
&\begin{bmatrix} I_2 + I_3 + m_2 l_2^2 + m_3 \left(L_2^2 + l_3^2 + 2L_2 l_3 \cos(\theta_3 + \phi_3) \right) & I_3 + m_3 l_3^2 + m_3 L_2 l_3 \cos(\theta_3 + \phi_3) \\ I_3 + m_3 l_3^2 + m_3 L_2 l_3 \cos(\theta_3 + \phi_3) & I_3 + m_3 l_3^2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \\
&\begin{bmatrix} 0 & -m_3 L_2 l_3 \sin(\theta_3 + \phi_3) \\ m_3 L_2 l_3 \sin(\theta_3 + \phi_3) & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix} + \begin{bmatrix} -M_2 l_2 \sin(\theta_1 + \theta_2 + \phi_2) - M_3 \left(L_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3 + \phi_3) \right) \\ -M_3 l_3 \sin(\theta_1 + \theta_2 + \theta_3 + \phi_3) \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} T_l \\ T_t \end{bmatrix} + \begin{bmatrix} 2m_3 L_2 l_3 \sin(\theta_3 + \phi_3) \left(\dot{\theta}_1 \dot{\theta}_3 + \dot{\theta}_2 \dot{\theta}_3 \right) \\ -m_3 L_2 l_3 \sin(\theta_3 + \phi_3) \left(\dot{\theta}_1^2 + 2\dot{\theta}_1 \dot{\theta}_2 \right) \end{bmatrix} \\
&- \begin{bmatrix} \begin{pmatrix} m_2 l_2 \cos(\theta_1 + \theta_2 + \phi_2) + m_3 L_2 \cos(\theta_1 + \theta_2) \\ + m_3 l_3 \cos(\theta_1 + \theta_2 + \theta_3 + \phi_3) \end{pmatrix} & \begin{pmatrix} I_2 + I_3 + m_2 l_2^2 + \\ m_3 \left(L_2^2 + l_3^2 + 2L_2 l_3 \cos(\theta_3 + \phi_3) \right) \end{pmatrix} & \begin{pmatrix} -L_2 \cos(\theta_1 + \theta_2) \\ -L_e \cos(\theta_1 + \theta_2 + \theta_3) \end{pmatrix} \\ m_3 l_3 \cos(\theta_1 + \theta_2 + \theta_3 + \phi_3) & m_3 L_2 l_3 \cos(\theta_3 + \phi_3) + I_3 + m_3 l_3^2 & -L_e \cos(\theta_1 + \theta_2 + \theta_3) \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ F_e \end{bmatrix}
\end{aligned} \tag{43}$$

1.4 Measurements:

1.4.1 The accelerometer measurements

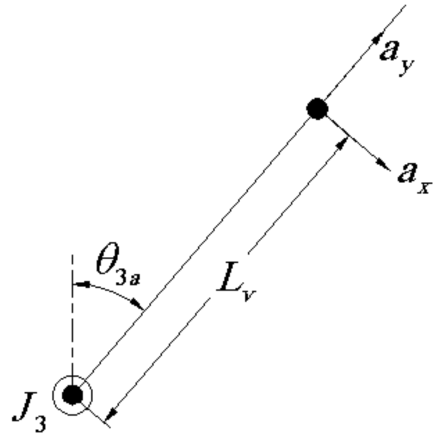

Figure 3 The Torso with the accelerometer measurements

$$a_x = \ddot{x}\cos(\theta_{3a}) + L_2\ddot{\theta}_{2a}\cos(\theta_{3a}-\theta_{2a}) + L_v\ddot{\theta}_{3a} + L_2\dot{\theta}_{2a}^2\sin(\theta_{3a}-\theta_{2a}) + g\sin(\theta_{3a}) \tag{44}$$

$$a_y = \ddot{x}\sin(\theta_{3a}) + L_2\ddot{\theta}_{2a}\sin(\theta_{3a}-\theta_{2a}) - L_2\dot{\theta}_{2a}^2\cos(\theta_{3a}-\theta_{2a}) - L_v\dot{\theta}_{3a}^2 - g\cos(\theta_{3a}) \tag{45}$$

$$a = \sqrt{a_x^2 + a_y^2} \tag{46}$$

$$\gamma_a = \tan^{-1}\left({a_x}/{a_y} \right) \tag{47}$$

And in terms of relative angles, Equations (43) and (44) become:

$$\begin{aligned} a_x &= \ddot{x}\cos(\theta_1+\theta_2+\theta_3) + \left(L_2\cos(\theta_3)+L_v\right)\left(\ddot{\theta}_1+\ddot{\theta}_2\right) + L_v\ddot{\theta}_3 \\ &+ L_2\sin(\theta_3)\left(\dot{\theta}_1^2+\dot{\theta}_2^2+2\dot{\theta}_1\dot{\theta}_2\right) + g\sin(\theta_1+\theta_2+\theta_3) \end{aligned} \tag{48}$$

$$\begin{aligned} a_y &= \ddot{x}\sin(\theta_1+\theta_2+\theta_3) + L_2\sin(\theta_3)\left(\ddot{\theta}_1+\ddot{\theta}_2\right) - g\cos(\theta_1+\theta_2+\theta_3) - L_v\dot{\theta}_3^2 \\ &-\left(L_2\cos(\theta_3)+L_v\right)\left(\dot{\theta}_1^2+\dot{\theta}_2^2\right) - 2\left(L_2\cos(\theta_3)+L_v\right)\dot{\theta}_1\dot{\theta}_2 - 2L_v\left(\dot{\theta}_1\dot{\theta}_3+\dot{\theta}_2\dot{\theta}_3\right) \end{aligned} \tag{49}$$

1.4.2 The contact forces

The contact forces acting between the plate and the foot $\left(F_S,\ \ F_F\ \ \ and\ F_B\right)$ can be obtained Using Newton's Euler method. Figure 4 shows the forces acting on the foot. Applying Newton's 2nd law in $\bar{x}$ and $\bar{y}$ directions; which are the parallel and the perpendicular to the platform respectively, and rearranging the results yields:

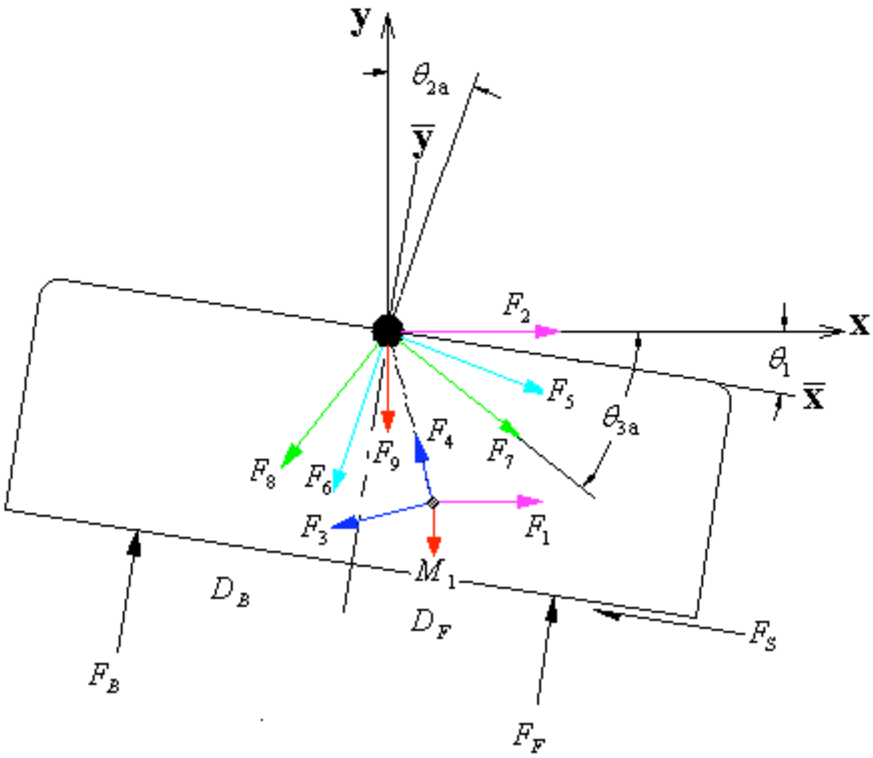

Figure 4 The free body diagram of the Foot, where:

$F_1 = m_1\ddot{x}$	$F_2 = (m_1 + m_2)\ddot{x} + F_e$	$F_3 = m_1 l_1 \ddot{\theta}_1$
$F_4 = m_1 l_1 \dot{\theta}_1^2$	$F_5 = (m_2 l_2 + m_3 L_2)\ddot{\theta}_2$	$F_6 = (m_2 l_2 + m_3 L_2)\dot{\theta}_2^2$
$F_7 = m_3 l_3 \ddot{\theta}_3$	$F_8 = m_3 l_3 \dot{\theta}_3^2$	$F_9 = M_2 + M_3$

$\sum F_{\bar{x}} =$ Inertial forces in $\bar{x}$ direction.

$$\begin{aligned}
F_S &= F_e\cos(\theta_1) + (M_1 + M_2 + M_3)\sin(\theta_1) - (m_1 + m_2 + m_3)\cos(\theta_1)\ddot{x} + m_1 l_1 \cos(\phi_1)\ddot{\theta}_1 \\
&- \left[m_2 l_2 \cos(\theta_{2a} + \phi_2 - \theta_1) + m_3 L_2 \cos(\theta_{2a} - \theta_1)\right]\ddot{\theta}_{2a} - m_3 l_3 \cos(\theta_{3a} + \phi_3 - \theta_1)\ddot{\theta}_{3a} + m_1 l_1 \sin(\phi_1)\dot{\theta}_1^2 \\
&+ \left[m_2 l_2 \sin(\theta_{2a} + \phi_2 - \theta_1) + m_3 L_2 \sin(\theta_{2a} - \theta_1)\right]\dot{\theta}_{2a}^2 + m_3 l_3 \sin(\theta_{3a} + \phi_3 - \theta_1)\dot{\theta}_{3a}^2
\end{aligned} \tag{50}$$

$\sum F_{\bar{y}} =$ Inertial forces in $\bar{y}$ direction.

$$\begin{aligned}
F_F + F_B &= -F_e\sin(\theta_1) + (M_1 + M_2 + M_3)\cos(\theta_1) + (m_1 + m_2 + m_3)\sin(\theta_1)\ddot{x} - m_1 l_1 \sin(\phi_1)\ddot{\theta}_1 \\
&- \left[m_2 l_2 \sin(\theta_{2a} + \phi_2 - \theta_1) + m_3 L_2 \sin(\theta_{2a} - \theta_1)\right]\ddot{\theta}_{2a} - m_3 l_3 \sin(\theta_{3a} + \phi_3 - \theta_1)\ddot{\theta}_{3a} + m_1 l_1 \cos(\phi_1)\dot{\theta}_1^2 \\
&+ \left[m_2 l_2 \cos(\theta_{2a} + \phi_2 - \theta_1) + m_3 L_2 \cos(\theta_{2a} - \theta_1)\right]\dot{\theta}_{2a}^2 + m_3 l_3 \cos(\theta_{3a} + \phi_3 - \theta_1)\dot{\theta}_{3a}^2
\end{aligned} \tag{51}$$

Rearranging Eq. (29), using the definition of the leg-actuators torque given in Eq. (31), yields:

$$F_B D_B - F_F D_F = -m_1 l_1 \cos(\phi_1 - \theta_1)\ddot{x} + \left(I_1 + m_1 l_1^2\right)\ddot{\theta}_1 - M_1 l_1 \sin(\phi_1 - \theta_1) + T_l - F_S L_1 \tag{52}$$

With respect to relative angles, Eq. (50), (51) and (52) become:

$$\begin{aligned}
F_S &= F_e\cos(\theta_1) + (M_1 + M_2 + M_3)\sin(\theta_1) - (m_1 + m_2 + m_3)\cos(\theta_1)\ddot{x} + m_1 l_1 \cos(\phi_1)\ddot{\theta}_1 \\
&- \left[m_2 l_2 \cos(\theta_2 + \phi_2) + m_3 L_2 \cos(\theta_2)\right]\left(\ddot{\theta}_1 + \ddot{\theta}_2\right) - m_3 l_3 \cos(\theta_2 + \theta_3 + \phi_3)\left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) \\
&+ m_1 l_1 \sin(\phi_1)\dot{\theta}_1^2 + \left[m_2 l_2 \sin(\theta_2 + \phi_2) + m_3 L_2 \sin(\theta_2)\right]\left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 + m_3 l_3 \sin(\theta_2 + \theta_3 + \phi_3)\left(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3\right)^2
\end{aligned} \tag{53}$$

$$F_F + F_B = -F_e \sin(\theta_1) + (M_1 + M_2 + M_3)\cos(\theta_1) + (m_1 + m_2 + m_3)\sin(\theta_1)\ddot{x} - m_1 l_1 \sin(\phi_1)\ddot{\theta}_1$$
$$-\left[m_2 l_2 \sin(\theta_2 + \phi_2) + m_3 L_2 \sin(\theta_2)\right]\left(\ddot{\theta}_1 + \ddot{\theta}_2\right) - m_3 l_3 \sin(\theta_2 + \theta_3 + \phi_3)\left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) + m_1 l_1 \cos(\phi_1)\dot{\theta}_1^2 \quad (54)$$
$$+\left[m_2 l_2 \cos(\theta_2 + \phi_2) + m_3 L_2 \cos(\theta_2)\right]\left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 + m_3 l_3 \cos(\theta_2 + \theta_3 + \phi_3)\left(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3\right)^2$$

$$F_B D_B - F_F D_F = -m_1 l_1 \cos(\phi_1 - \theta_1)\ddot{x} + \left(I_1 + m_1 l_1^2\right)\ddot{\theta}_1 - M_1 l_1 \sin(\phi_1 - \theta_1) + T_l - F_S L_1 \quad (55)$$

Equation (53) describes the shear (friction) measurement force F_S between the foot and the platform while Eqs. (54) and (55) can be solved for F_F and F_B after substituting Eq. (53) in Eq. (55) to eliminate F_S .

1.5 Linearization

In order to obtain the linear model for PostuRob II, an operating point is to be selected first. This point is chosen to represent the system in its vertical position. About this operating point and for small rotations, one can assume:

$$\cos\theta_2 = 1 \quad (56)$$

$$\cos\theta_3 = 1 \quad (57)$$

$$\sin\theta_2 = \theta_2 \quad (58)$$

$$\sin\theta_3 = \theta_3 \quad (59)$$

Further, it is necessary to neglect Coriolis and normal (centrifugal) acceleration terms arising from multiplying angular velocities as $\dot{\theta}_1\dot{\theta}_2$, $\dot{\theta}_1\dot{\theta}_3$, $\dot{\theta}_2\dot{\theta}_3$, $\dot{\theta}_2^2$, and $\dot{\theta}_3^2$.

Based on Eq. (43) the resulting linearized model that describes the system in the operating region is:

$$\underbrace{\begin{bmatrix} I_2+I_3+m_2l_2^2+m_3\left(L_2^2+l_3^2+2L_2l_3\cos\phi_3\right) & I_3+m_3l_3^2+m_3L_2l_3\cos\phi_3 \\ I_3+m_3l_3^2+m_3L_2l_3\cos\phi_3 & I_3+m_3l_3^2 \end{bmatrix}}_{M}\begin{bmatrix}\ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix}=$$

$$\underbrace{\begin{bmatrix} M_2l_2\cos\phi_2+M_3L_2+M_3l_3\cos\phi_3 & M_3l_3\cos\phi_3 \\ M_3l_3\cos\phi_3 & M_3l_3\cos\phi_3 \end{bmatrix}}_{K}\begin{bmatrix}\theta_2\\ \theta_3\end{bmatrix}+\begin{bmatrix}T_l\\ T_t\end{bmatrix}+\underbrace{\begin{bmatrix} M_2l_2\sin\phi_2+M_3l_3\sin\phi_3 \\ M_3l_3\sin\phi_3 \end{bmatrix}}_{W}$$

$$-\underbrace{\begin{bmatrix} \begin{pmatrix} m_2l_2\cos\phi_2+m_3L_2 \\ +m_3l_3\cos\phi_3 \end{pmatrix} & -\begin{pmatrix} M_2l_2\cos\phi_2+M_3L_2 \\ +M_3l_3\cos\phi_3 \end{pmatrix} & 0 & \begin{pmatrix} I_2+I_3+m_2l_2^2 \\ +m_3\left(L_2^2+l_3^2+2L_2l_3\cos\phi_3\right) \end{pmatrix} & -L_2-L_e \\ m_3l_3\cos\phi_3 & -M_3l_3\cos\phi_3 & 0 & I_3+m_3l_3^2+m_3L_2l_3\cos\phi_3 & -L_e \end{bmatrix}}_{B_d}\begin{bmatrix}\ddot{x}\\ \theta_1\\ \dot{\theta}_1\\ \ddot{\theta}_1\\ F_e\end{bmatrix}$$

(60)

Where

M is the inertia matrix

K is the quasi stiffness matrix

B_d is the disturbance matrix

W is the gravity force (weight) matrix

The state-space model for the PostuRob II is expressed using the four state variables $\begin{bmatrix}\theta_2 & \theta_3 & \dot{\theta}_2 & \dot{\theta}_3\end{bmatrix}$. Based on Eq. (42), and using the previously defined variables, the resulting state-space representation for the system is:

$$\begin{bmatrix}\dot{z}\\ \ddot{z}\end{bmatrix}=\underbrace{\begin{bmatrix}0 & I_2\\ M^{-1}K & 0\end{bmatrix}}_{A}\begin{bmatrix}z\\ \dot{z}\end{bmatrix}+\underbrace{\begin{bmatrix}0\\ M^{-1}\end{bmatrix}}_{B}T+\underbrace{\begin{bmatrix}0\\ M^{-1}B_d\end{bmatrix}}_{N}F_d+\underbrace{\begin{bmatrix}0\\ M^{-1}W\end{bmatrix}}_{H} \quad (61)$$

Where:

$z=[\theta_2 \quad \theta_3]^T$

$T=[T_l \quad T_t]^T$

$F_d=\begin{bmatrix}\ddot{x} & \theta_1 & \dot{\theta}_1 & \ddot{\theta}_1 & F_e\end{bmatrix}^T$

I_2 is the identity matrix of order 2.

The output equations describing the contact forces and accelerometer measurements can be also linearized at the same operating point. Thus, Eqs. (48), (49) and (53-55) become:

$$a_x = \ddot{x} + (L_2 + L_v)\ddot{\theta}_1 + (L_2 + L_v)\ddot{\theta}_2 + L_v\ddot{\theta}_3 + g(\theta_1 + \theta_2 + \theta_3) \tag{62}$$

$$a_y = -g \tag{63}$$

$$\begin{aligned} F_S = &-(m_1 + m_2 + m_3)\ddot{x} + \left[m_1 l_1 \cos\phi_1 - m_2 l_2 \cos\phi_2 - m_3 L_2 - m_3 l_3 \cos\phi_3 \right]\ddot{\theta}_1 \\ &-\left[m_2 l_2 \cos\phi_2 + m_3 L_2 - m_3 l_3 \cos\phi_3 \right]\ddot{\theta}_2 - \left[m_3 l_3 \cos\phi_3 \right]\ddot{\theta}_3 + F_e + (M_1 + M_2 + M_3)\theta_1 \end{aligned} \tag{64}$$

$$\begin{aligned} F_F + F_B = &(M_1 + M_2 + M_3) - [m_1 l_1 \sin\phi_1 + m_2 l_2 \sin\phi_2 + m_3 l_3 \sin\phi_3]\ddot{\theta}_1 \\ &-[m_2 l_2 \sin\phi_2 + m_3 l_3 \sin\phi_3]\ddot{\theta}_2 - [m_3 l_3 \sin\phi_3]\ddot{\theta}_3 \end{aligned} \tag{65}$$

$$F_B D_B - F_F D_F = -m_1 l_1 \cos(\phi_1)\ddot{x} + \left(I_1 + m_1 l_1^2\right)\ddot{\theta}_1 - M_1 l_1 \sin\phi_1 + (M_1 l_1 \cos\phi_1)\theta_1 + T_l - F_S L_1 \tag{66}$$

2.1 Parameters Definition

2.1.1 Description

The main parameters of PostuRob II are defined explicitly in an m-file, which are then used by the S-function during simulation. Such a definition permits the user to flexibly change any of these parameters, with no need to make any further changes in other files. The parameters included in the m-file define the mass, moment of inertia, length, and COM eccentricity angle of each of the three links of PostuRob II related to the variables shown in Fig. 1.

2.1.2 Parameters M-file

```matlab
%==================================================================
%                   PostuRob_II_Parameters.m
%==================================================================
% Definig the basic parametres for the system
% units are K.M.s
clear all
global m1 m2 m3 I1 I2 I3 L1 L2 Lv Le l1 l2 l3 phi1 phi2 phi3 M1 M2
M3 Df Db g
%System parameters
m1=8.5;                 %Foot mass
m2=20;                  %Leg mass
m3=40;                  %Torso mass
I1=0.35/4;              %Foot Mass Moment of Inertia about its COM
I2=7/4;                 %Leg Mass Moment of Inertia about its COM
I3=4.8/4;               %Torso Mass Moment of Inertia about its
COM
L1=0.15;                %Foot Length
L2=0.95;                %Leg length
Lv=0.55;                %distance from J3 to the accelerometer
Le=0.20;                %distance from J3 to the disturbance Fe
w1=0.02;                %COM1 horizental offset
h1=0.04;                %COM1 vertical offset
l1=sqrt(w1^2+h1^2);     %Distance from J1 to COM1
phi1=atan(w1/(L1-h1));  %Angle between the line J1-COM1 the foot
                         centerline
w2=-0.02;               %COM2 horizental offset
h2=0.6;                 %COM2 vertical offset
l2=sqrt(w2^2+h2^2);     %Distance from J2 to COM2
phi2=atan(w2/h2);       %Angle between the line J2-COM2 the leg
                         centerline
w3=0.05;                %COM3 horizental offset
h3=0.3;                 %COM3 vertical offset
l3=sqrt(w3^2+h3^2);     %Distance from J3 to COM3
phi3=atan(w3/h3);       %Angle between the line J3-COM3 the torso
                         centerline
g=9.81;                 %Gravity acceleration constant
M1=m1*g;                %Foot wieght
M2=m2*g;                %Leg wieght
M3=m3*g;                %Torso wieght
Dla=0.1;                %Distance from J1 to the leg actuators
                         action point
Dta=0.1;                %Distance from J2 to the torso actuators
                         action point
Df=0.15;                %Distance from J1 to the front contact
                         force
Db=0.1;                 %Distance from J1 to the back contact
                         force
```

2.2 Linearized Model

2.2.1 Description

The state-space model for the PostuRob II is expressed using the four state variables $\begin{bmatrix}\theta_2 & \theta_3 & \dot{\theta}_2 & \dot{\theta}_3\end{bmatrix}$. Based on Eq. (60), and using the previously defined variables, the resulting state-space representation for the system is:

$$\begin{bmatrix}\dot{z}\\ \ddot{z}\end{bmatrix}=\underbrace{\begin{bmatrix}0 & I_2\\ M^{-1}K & 0\end{bmatrix}}_{A}\begin{bmatrix}z\\ \dot{z}\end{bmatrix}+\underbrace{\begin{bmatrix}0\\ M^{-1}\end{bmatrix}}_{B}T+\underbrace{\begin{bmatrix}0\\ M^{-1}B_d\end{bmatrix}}_{N}F_d+\underbrace{\begin{bmatrix}0\\ M^{-1}W\end{bmatrix}}_{H} \tag{67}$$

Where:

$z=[\theta_2 \quad \theta_3]^T$

$T=[T_l \quad T_t]^T$

$F_d=\begin{bmatrix}\ddot{x} & \theta_1 & \dot{\theta}_1 & \ddot{\theta}_1 & F_e\end{bmatrix}^T$

I_2 is the identity matrix of order 2.

Or in a compact form as:

$$\dot{Z}=AZ+BT+NF_d+H \tag{68}$$

and the output (measurement) vector:

$$Y=CZ+DT+N_{out}F_d+H_{out} \tag{69}$$

with

$$Y=\begin{bmatrix}\theta_2 & \theta_3 & \dot{\theta}_2 & \dot{\theta}_3 & \omega & a_x & a_y & F_s & F_F & F_B\end{bmatrix}^T \tag{70}$$

The constant matrices $(A, B, N, H, C, D, N_{out}, H_{out})$ based on the linearized model are calculated in the following m-file.

2.1.2 Linearized Model M-file

```
%=================================================================
%          PostuRob_II_Linearized_State_Space_Model.m
%=================================================================
% Based on Eqs. 67-70, the system matrices corresponding to the
% state and output equations for the linearized model are
calculated.

% Zdot=A*Z+B*T+N*Fd+H
% Y=C*Z+D*T+Nout*Fd+Hout
% Where:
%   Z = [Th2 Th3 Th2d Th3d]'
%   T = [Tl Tt]'
%   Fd= [Xdd Th1 Th1d Th1dd Fe]

M(1,1)=I2+I3+m2*l2^2+m3*(L2^2+l3^2+2*L2*l3*cos(phi3));
M(1,2)=I3+m3*l3^2+m3*L2*l3*cos(phi3);
M(2,1)=I3+m3*l3^2+m3*L2*l3*cos(phi3);
M(2,2)=I3+m3*l3^2;


K(1,1)=M2*l2*cos(phi2)+M3*L2+M3*l3*cos(phi3);
K(1,2)=M3*l3*cos(phi3);
K(2,1)=M3*l3*cos(phi3);
K(2,2)=M3*l3*cos(phi3);



W=[M2*l2*sin(phi2)+M3*l3*sin(phi3)
   M3*l3*sin(phi3)                 ];

Bd(1,1)=-m2*l2*cos(phi2)-m3*L2-m3*l3*cos(phi3);
Bd(1,2)=M2*l2*cos(phi2)+M3*L2+M3*l3*cos(phi3);
Bd(1,3)=0;
Bd(1,4)=-I2-I3-m2*l2^2-m3*(L2^2+l3^2+2*L2*l3*cos(phi3));
Bd(1,5)=L2+Le;
Bd(2,1)=-m3*l3*cos(phi3);
Bd(2,2)=M3*l3*cos(phi3);
Bd(2,3)=0;
Bd(2,4)=-I3-m3*l3^2-m3*L2*l3*cos(phi3);
Bd(2,5)=Le;

% The linearized system equations according to equation 60

A=[zeros(2)    eye(2)
   inv(M)*K    zeros(2)]

B=[zeros(2)
   inv(M)]

N=[zeros(2,5)
   inv(M)*Bd]
```

```
H=[zeros(2,1)
   inv(M)*W]

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                         Output Equations
% The outputs of the system are:
% [Th2 Th3 Th2d Th3d Omega_g  ax  ay  Fs  Ff  Fb]';
% Refering to Eqs. (61 - 65) the following vectors and matrices
are defined
% so as to simplify the outputs representation

Vax=[L2+Lv L2];

VFs1(1,1)=-m2*l2*cos(phi2)-m3*L2-m3*l3*cos(phi3);
VFs1(1,2)=-m3*l3*cos(phi3);

VFs2(1,1)=-(m1+m2+m3);
VFs2(1,2)=m1*l1*cos(phi1)-m2*l2*cos(phi2)-m3*L2-m3*l3*cos(phi3);
VFs2(1,3)=0;
VFs2(1,4)=M1+M2+M3;
VFs2(1,5)=1;

MF1=[1   1
    -Df Db];

MF2=[-m2*l2*sin(phi2)-m3*l3*sin(phi3) -m3*l3*sin(phi3)
     0                                 0              ];

MF3(1,1)=0;
MF3(1,2)=-m1*l1*sin(phi1)-m2*l2*sin(phi2)-m3*l3*sin(phi3);
MF3(1,3)=0;
MF3(1,4)=0;
MF3(1,5)=0;
MF3(2,1)=-m1*l1*cos(phi1);
MF3(2,2)=I1+m1*l1^2;
Mf3(2,3)=0;
MF3(2,4)=M1*l1*cos(phi1);
MF3(2,5)=0;

 % Using the previously defined matrices; the output equations can
be
% rearranged as follows:
%
% Th2     = [1  0  0  0]*Z +[0  0]*T+[0  0  0  0  0]*Fd+[0]*W
%
% Th3     = [0  1  0  0]*Z +[0  0]*T+[0  0  0  0  0]*Fd+[0]*W
%
% Th2d    = [0  0  1  0]*Z +[0  0]*T+[0  0  0  0  0]*Fd+[0]*W
%
% Th3d    = [0  0  0  1]*Z +[0  0]*T+[0  0  0  0  0]*Fd+[0]*W
%
% Omega_g = [0  0  1  1]*Z +[0  0]*T+[0  0  1  0  0]*Fd+[0]*W
%
% ax      = [Vax*inv(M)*K+[g g] 0  0]*Z+ [Vax*inv(M)*Bt]*T+
```

```
%            [Vax*inv(M)*Bd+[1 L2+Lv 0 g 0]]*Fd+Vax*inv(M)*W
%
% ay       = [0 0 0 0]*Z +[0 0]*T+[0 0 0 0 0]*Fd+[-g]*W
%
% Fs       = [VFs1*inv(M)*K 0 0]*Z+ [VFs1*inv(M)*Bt]*T+
%            [VFs1*inv(M)*Bd+VFs2]*Fd+VFs1*inv(M)*W
%
%[Ff; Fb]= [inv(MF1)*(MF2+[0;-L1]*VFs1)*inv(M)*K zeros(2)]*Z +
%           [inv(MF1)*((MF2+[0;-L1]*VFs1)*inv(M)*Bt+[0 0;1 0]]*T+
%           [inv(MF1)*((MF2+[0;-L1]*VFs1)*inv(M)*Bd+[0;-
L1]*VFs2+MF3)]*Fd+
%           [inv(MF1)*((MF2+[0;-L1]*VFs1)*inv(M)*W+[M1+M2+M3;-
M1*l1*sin(phi1)]]
%
% Based on the above equations,the matrices C, D, Nout, Hout are
defined as
% follows:

C=[eye(4);
    0 0 1 1;
    Vax*inv(M)*K+[g g] 0 0;
    0 0 0 0;
    VFs1*inv(M)*K 0 0;
    inv(MF1)*(MF2+[0;-L1]*VFs1)*inv(M)*K zeros(2);
    ]

D=[zeros(4,2)
    0 0
    Vax*inv(M)*Bt
    0 0
    VFs1*inv(M)*Bt
    inv(MF1)*((MF2+[0;-L1]*VFs1)*inv(M)*Bt+[0 0;1 0])
    ]

Nout=[zeros(4,5)
       0 0 1 0 0
       Vax*inv(M)*Bd+[1 L2+Lv 0 g 0]
       0 0 0 0 0
       VFs1*inv(M)*Bd+VFs2
       inv(MF1)*((MF2+[0;-L1]*VFs1)*inv(M)*Bd+[0;-L1]*VFs2+MF3)
      ]

Hout=[zeros(4,1)
       Vax*inv(M)*W
       -g
       VFs1*inv(M)*W
       inv(MF1)*((MF2+[0;-L1]*VFs1)*inv(M)*W+[M1+M2+M3;-
M1*l1*sin(phi1)])
     ]
```

2.3 S-Function

2.3.1 Description

An S-function is a computer language description of a Simulink block. S-functions can be written in MATLAB®, C, C++, or Fortran. They are compiled as MEX-files using the mex utility. As with other MEX-files, they are dynamically linked into MATLAB when needed.

S-functions use a special calling syntax that enables the user to interact with Simulink equation solvers. This is very similar to the interaction that takes place between the solvers and built-in Simulink blocks. The form of an S-function is very general and can accommodate continuous, discrete, and hybrid systems.

The most common use of S-functions is to create custom Simulink blocks. One can use S-functions for a variety of applications, including; adding new general purpose blocks to Simulink, adding blocks that represent hardware device drivers, incorporating existing C code into a simulation, describing a system as a set of mathematical equations and using graphical animations.

As other Simulink blocks, S-functions consist of a set of inputs, a set of states, and a set of outputs, where the outputs are a function of the sample time, the inputs, and the block's states, as expressed by the following mathematical relations:

$$y = f_O(t, x, u) \qquad (Output)$$

$$\dot{x}_c = f_d(t, x, u) \qquad (Derivative)$$

Executing an S-function model proceeds in a set of stages. First comes the initialization phase. In this phase, S-function propagates data types, and sample times, evaluates block parameters, determines block execution order, and allocates memory. Then the simulation loop starts, where each pass through the loop is referred to as a simulation step. During each simulation step, S-function invokes the functions that compute the block's states, derivatives, and outputs for the current sample time. This continues until the simulation is complete.

The following figure illustrates the stages of an S-function execution:

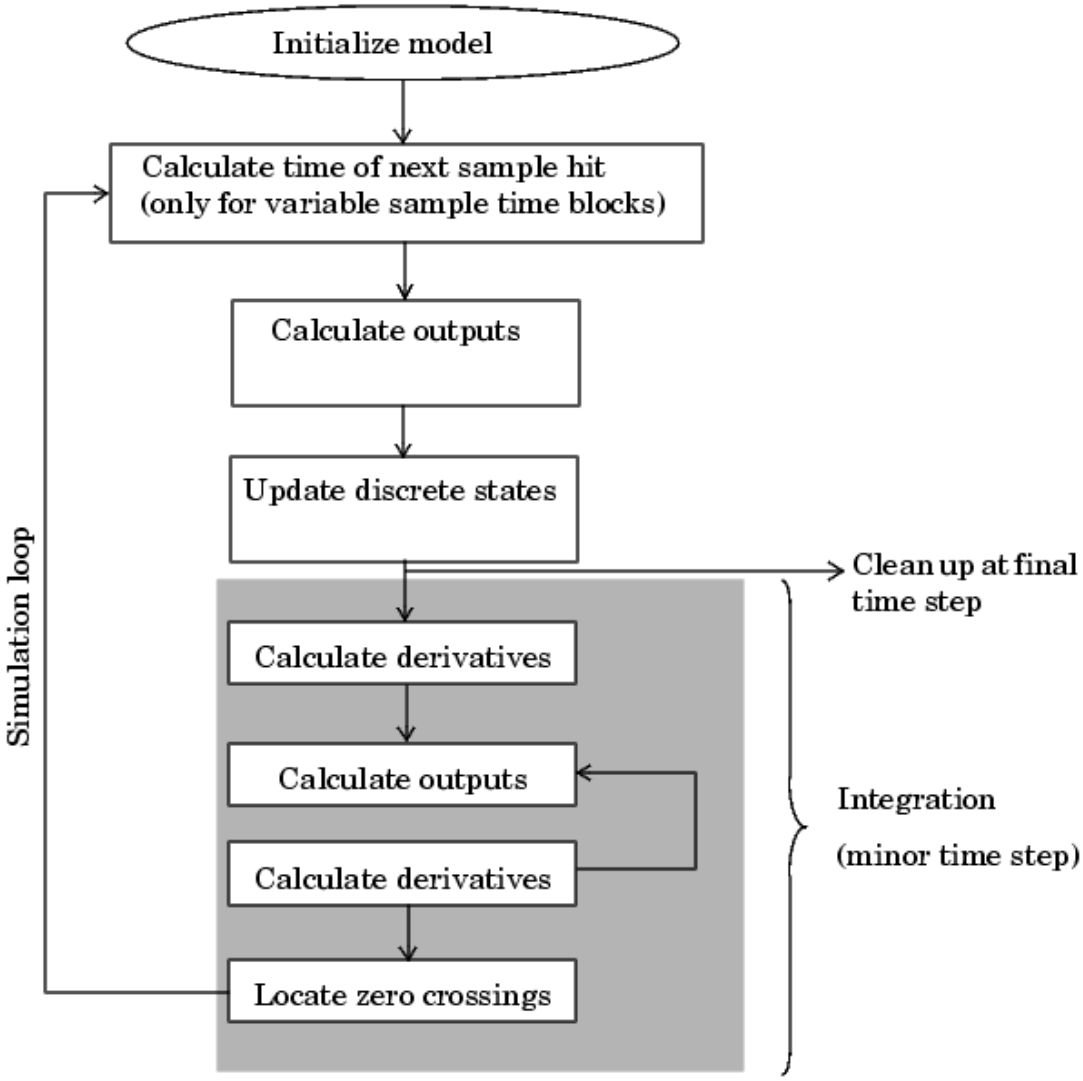

Figure 2-1 S-function execution stages.

2.3.2 S-Function File

The s-function file used to define the nonlinear dynamics of PostuRob II is given below. The kinematic and dynamic parameters are passed to this s-function as global parameters. So, these variables should not be used by other Matlab commands. The control inputs to the system are the two commanded torques. The external disturbances acting on the system (platform tilt and translation as well as the external forces) are passed to the s-function as inputs. So the control input and the external disturbances are collected in one input vector. The number of states used is 4 corresponding to the relative position of the second link to the first and of the third link to the second one and their velocities. Finally, the output vector of this function comprises all 10 measurements (4 states, 3 contact forces, and 3 vestibular outputs).

This function has two main parts: the first relies on the nonlinear equation 42 to calculate the derivative of the state vector and on the nonlinear equations 47-51 to calculate the outputs. It is important to note that the linearized model is not used in the s-function and thus it serves as a good simulation tool reflecting the physics considered in the modeling.

```
function [sys,x0,str,ts] =
PostuRob_II_Nonlinear_Dynamics(t,x,u,flag,th2_0,th3_0,th2d0,th3d0)
global m1 m2 m3 I1 I2 I3 L1 L2 Lv l1 l2 l3 phi1 phi2 phi3 M1 M2 M3
Le Df Db Fs Ff Fb ax ay g

%PostuRob_II_Nonlinear_Dynamics
%Defines the nonlinear dynamics of PostuRob II according to
Lagrangian derivations


% Nonlinear model for PostuRob II, comprised of:
%       2 inputs:
%               1- Leg torque   (Tl)
%               2- Torso torque (Tt)
%
%       5 disturbances
%               1- Xdd acting on the platform
%               2- Theta1   acting on the platform joint
%               3- Theta1d  acting on the platform joint
%               4- Theta1dd acting on the platform joint
%               5- Fe Acting on the Torso
%
%       4 states:
%               1- Theta2 (th2)
%               2- Theta2 (th3)
%               3- Theta2_dot (th2d)
%               4- Theta3_dot (th3d)
%
%       10 outputs:
%               1-  Theta2 (th2)
%               2-  Theta3 (th3)
%               3-  Theta2_dot (th2d)
%               4-  Theta3_dot (th3d)
%               5-  Gyrometer angular velocity (Omega_g)
%               6-  Accelerometer x-direction measurement(ax)
%               7-  Accelerometer y-direction measurement(ay)
%               8-  Contact Force Fs
%               9-  Contact Force Ff
%               10-  Contact Force Fb
%
%       4 initial conditions:
%               1- Initial link2 position =th2_0
%               2- Initial link3 position =th3_0
%               3- Initial link2 velocityt=th2d0
%               4- Initial link3 velocityt=th3d0

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

% The following outlines the general structure of an S-function.
%
switch flag,
```

```
  %%%%%%%%%%%%%%%%%%
  % Initialization %
  %%%%%%%%%%%%%%%%%%
  case 0,
      Fs=0;
      Ff=Db*(M1+M2+M3)/(Df+Db);
      Fb=Df*(M1+M2+M3)/(Df+Db);
      Omega_g=0;
      ax=0;
      ay=-g;

[sys,x0,str,ts]=mdlInitializeSizes(t,x,u,flag,th2_0,th3_0,th2d0,th
3d0);

  %%%%%%%%%%%%%%%
  % Derivatives %
  %%%%%%%%%%%%%%%
  case 1,
    sys=mdlDerivatives(t,x,u);

  %%%%%%%%%%
  % Update %
  %%%%%%%%%%
  case 2,
    sys=mdlUpdate(t,x,u);

  %%%%%%%%%%%
  % Outputs %
  %%%%%%%%%%%
  case 3,
    sys=mdlOutputs(t,x,u);

  %%%%%%%%%%%%%%%%%%%%%%%
  % GetTimeOfNextVarHit %
  %%%%%%%%%%%%%%%%%%%%%%%
  case 4,
    sys=mdlGetTimeOfNextVarHit(t,x,u);

  %%%%%%%%%%%%%
  % Terminate %
  %%%%%%%%%%%%%
  case 9,
    sys=mdlTerminate(t,x,u);

  %%%%%%%%%%%%%%%%%%%%
  % Unexpected flags %
  %%%%%%%%%%%%%%%%%%%%
  otherwise
    error(['Unhandled flag = ',num2str(flag)]);

end

% end sfuntmpl
```

```
%
%==============================================================================
=============
% mdlInitializeSizes
% Return the sizes, initial conditions, and sample times for the
S-function.
%==============================================================================
=============
%
function
[sys,x0,str,ts]=mdlInitializeSizes(t,x,u,flag,th2_0,th3_0,th2d0,th
3d0)

%
% call simsizes for a sizes structure, fill it in and convert it
to a
% sizes array.
%
% Note that in this example, the values are hard coded.  This is
not a
% recommended practice as the characteristics of the block are
typically
% defined by the S-function parameters.
%
sizes = simsizes;

sizes.NumContStates  = 4;
sizes.NumDiscStates  = 0;
sizes.NumOutputs     = 10;
sizes.NumInputs      = 7;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;   % at least one sample time is needed

sys = simsizes(sizes);

%
% initialize the initial conditions
%
x0  = [th2_0,th3_0,th2d0,th3d0];

% str is always an empty matrix
%
str = [];

%
% initialize the array of sample times
%
ts  = [0 0];

% end mdlInitializeSizes

%
%==============================================================================
```

```
============
% mdlDerivatives
% Return the derivatives for the continuous states.
%=====================================================================
============
%
function sys=mdlDerivatives(t,x,u,th2_0,th3_0,th2d0,th3d0)
global m1 m2 m3 I1 I2 I3 L1 L2 Lv l1 l2 l3 phi1 phi2 phi3 M1 M2 M3
Le Df Db g
Tl=u(1);
Tt=u(2);
Xdd=u(3);
Th1=u(4);
Th1d=u(5);
Th1dd=u(6);
Fe=u(7);
Th2=x(1);
Th2d=x(3);
Th3=x(2);
Th3d=x(4);

a=[I2+I3+m2*l2^2+m3*(L2^2+l3^2+2*L2*l3*cos(Th3+phi3)),
I3+m3*l3^2+m3*L2*l3*cos(Th3+phi3);
    I3+m3*l3^2+m3*L2*l3*cos(Th3+phi3),                I3+m3*l3^2
];

b=[
m3*L2*l3*sin(Th3+phi3)*(Th3d^2+2*Th1d*Th3d+2*Th2d*Th3d)+M2*l2*sin(
Th1+Th2+phi2)+M3*(L2*sin(Th1+Th2)+l3*sin(Th1+Th2+Th3+phi3))+Tl-
(m2*l2*cos(Th1+Th2+phi2)+m3*L2*cos(Th1+Th2)+m3*l3*cos(Th1+Th2+Th3+
phi3))*Xdd-
(I2+I3+m2*l2^2+m3*(L2^2+l3^2+2*L2*l3*cos(Th3+phi3)))*Th1dd+(L2*cos
(Th1+Th2)+Le*cos(Th1+Th2+Th3))*Fe;
   -
m3*L2*l3*sin(Th3+phi3)*(Th1d+Th2d)^2+M3*l3*sin(Th1+Th2+Th3+phi3)+T
t-m3*l3*cos(Th1+Th2+Th3+phi3)*Xdd-
(m3*L2*l3*cos(Th3+phi3)+I3+m3*l3^2)*Th1dd+Le*cos(Th1+Th2+Th3)*Fe];

c= inv(a)*b;
Th2dd=c(1);
Th3dd=c(2);
sys = [ Th2d ;Th3d; c(1);c(2) ];

% end mdlDerivatives

%=====================================================================
============
% mdlUpdate
% Handle discrete state updates, sample time hits, and major time
step
% requirements.
%=====================================================================
============
%
```

```
function sys=mdlUpdate(t,x,u)

sys = [];

% end mdlUpdate

%
%=============================================================
============
% mdlOutputs
% Return the block outputs.
%=============================================================
============
%
function sys=mdlOutputs(t,x,u)
global m1 m2 m3 I1 I2 I3 L1 L2 Lv l1 l2 l3 phi1 phi2 phi3 M1 M2 M3
Le Df Db g
Tl=u(1);
Tt=u(2);
Xdd=u(3);
Th1=u(4);
Th1d=u(5);
Th1dd=u(6);
Fe=u(7);
Th2=x(1);
Th2d=x(3);
Th3=x(2);
Th3d=x(4);

a=[I2+I3+m2*l2^2+m3*(L2^2+l3^2+2*L2*l3*cos(Th3+phi3))
I3+m3*l3^2+m3*L2*l3*cos(Th3+phi3)
    I3+m3*l3^2+m3*L2*l3*cos(Th3+phi3)                I3+m3*l3^2
];

b=[
m3*L2*l3*sin(Th3+phi3)*(Th3d^2+2*Th1d*Th3d+2*Th2d*Th3d)+M2*l2*sin(
Th1+Th2+phi2)+M3*(L2*sin(Th1+Th2)+l3*sin(Th1+Th2+Th3+phi3))+Tl-
(m2*l2*cos(Th1+Th2+phi2)+m3*L2*cos(Th1+Th2)+m3*l3*cos(Th1+Th2+Th3+
phi3))*Xdd-
(I2+I3+m2*l2^2+m3*(L2^2+l3^2+2*L2*l3*cos(Th3+phi3)))*Th1dd+(L2*cos
(Th1+Th2)+Le*cos(Th1+Th2+Th3))*Fe;
    -
m3*L2*l3*sin(Th3+phi3)*(Th1d+Th2d)^2+M3*l3*sin(Th1+Th2+Th3+phi3)+T
t-m3*l3*cos(Th1+Th2+Th3+phi3)*Xdd-
(m3*L2*l3*cos(Th3+phi3)+I3+m3*l3^2)*Th1dd+Le*cos(Th1+Th2+Th3)*Fe];

c=inv(a)*b;
Th2dd=c(1);
Th3dd=c(2);
Omega_g=Th1d+Th2d+Th3d;
ax=Xdd*cos(Th1+Th2+Th3)+(L2*cos(Th3)+Lv)*(Th1dd+Th2dd)+Lv*Th3dd+L2
*sin(Th3)*(Th1d+Th2d)^2+g*sin(Th1+Th2+Th3);

ay=Xdd*sin(Th1+Th2+Th3)+L2*sin(Th3)*(Th1dd+Th2dd)-
```

```
(L2*sin(Th3)+Lv)*(Th1d+Th2d)^2-Lv*Th3d^2-
2*Lv*(Th1d*Th3d+Th2d*Th3d)-g*cos(Th1+Th2+Th3);

Fs=Fe*cos(Th1)+(M1+M2+M3)*sin(Th1)-
(m1+m2+m3)*cos(Th1)*Xdd+m1*l1*cos(phi1)*Th1dd-
(m2*l2*cos(Th2+phi2)+m3*L2*cos(Th2))*(Th1dd+Th2dd)-
m3*l3*cos(Th2+Th3+phi3)*(Th1dd+Th2dd+Th3dd)+m1*l1*sin(phi1)*Th1d^2
+(m2*l2*sin(Th2+phi2)+m3*L2*sin(Th2))*(Th1d+Th2d)^2+m3*l3*cos(Th2+
Th3+phi3)*(Th1d+Th2d+Th3d)^2;

Mf1=[1 1; -Df Db];
Mf2=[-Fe*sin(Th1)+(M1+M2+M3)*cos(Th1)+(m1+m2+m3)*sin(Th1)*Xdd-
m1*l1*sin(phi1)*Th1dd-
(m2*l2*sin(Th2+phi2)+m3*L2*sin(Th2))*(Th1dd+Th2dd)-
m3*l3*sin(Th2+Th3+phi3)*(Th1dd+Th2dd+Th3dd)+m1*l1*cos(phi1)*Th1d^2
+(m2*l2*cos(Th2+phi2)+m3*L2*cos(Th2))*(Th1d+Th2d)^2+m3*l3*cos(Th2+
Th3+phi3)*(Th1d+Th2d+Th3d)^2;
   -m1*l1*cos(phi1-Th1)*Xdd+(I1+m1*l1^2)*Th1dd-M1*l1*sin(phi1-
Th1)+Tl-Fs*L1 ];
Mf3=inv(Mf1)*Mf2;

Ff=Mf3(1);
Fb=Mf3(2);
sys = [x; Omega_g; ax; ay; Fs; Ff; Fb ];

% end mdlOutputs

%
%==================================================================
============
% mdlGetTimeOfNextVarHit
% Return the time of the next hit for this block.  Note that the
result is
% absolute time.  Note that this function is only used when you
specify a
% variable discrete-time sample time [-2 0] in the sample time
array in
% mdlInitializeSizes.
%==================================================================
============
%
function sys=mdlGetTimeOfNextVarHit(t,x,u)

sampleTime = 1;    %  Example, set the next hit to be one second
later.
sys = t + sampleTime;

% end mdlGetTimeOfNextVarHit

%
%==================================================================
============
% mdlTerminate
% Perform any end of simulation tasks.
```

```
%==================================================================
============
%
function sys=mdlTerminate(t,x,u)

sys = [];

% end mdlTerminate
```

2.4 Simulink Model

2.4.1 Description

Simulink® is a software package for modeling, simulating, and analyzing dynamic systems. It supports linear and nonlinear systems, modeled in continuous time, discrete time, or a hybrid of the two. With Simulink one can model physical systems and controllers as block diagrams, and with the Virtual Reality Toolbox, it is possible to visualize the simulation of dynamic systems over time, as shown in the upcoming section.

For modeling, Simulink provides a graphical user interface (GUI) for building models as blocks utilizing a comprehensive block library of sinks, sources, linear and nonlinear components, and connectors. One can also customize and create his own blocks using S-Functions, as shown in the previous section.

After creating the Simulink model, one can simulate it, using a choice of integration methods. Using scopes and other display blocks, it is possible to see the simulation results while the simulation is running. In addition, one can change many parameters and see what happens for "what if" exploration.

2.4.2 Simulink Model File

PostuRob II Simulink model is shown in Fig. 2-2. This model consists of the following main blocks:

- S-function block (Simulink icon of the previous s-function) that defines the dynamics of the system and its outputs based on the results obtained in part one of this documentation. The complete code that lies behind this block is shown in the previous section.
- Several graphical scopes for monitoring the states and outputs over time.
- A virtual world scope that uses the signals generated by the s-function execution to animate PostuRob II virtual model.

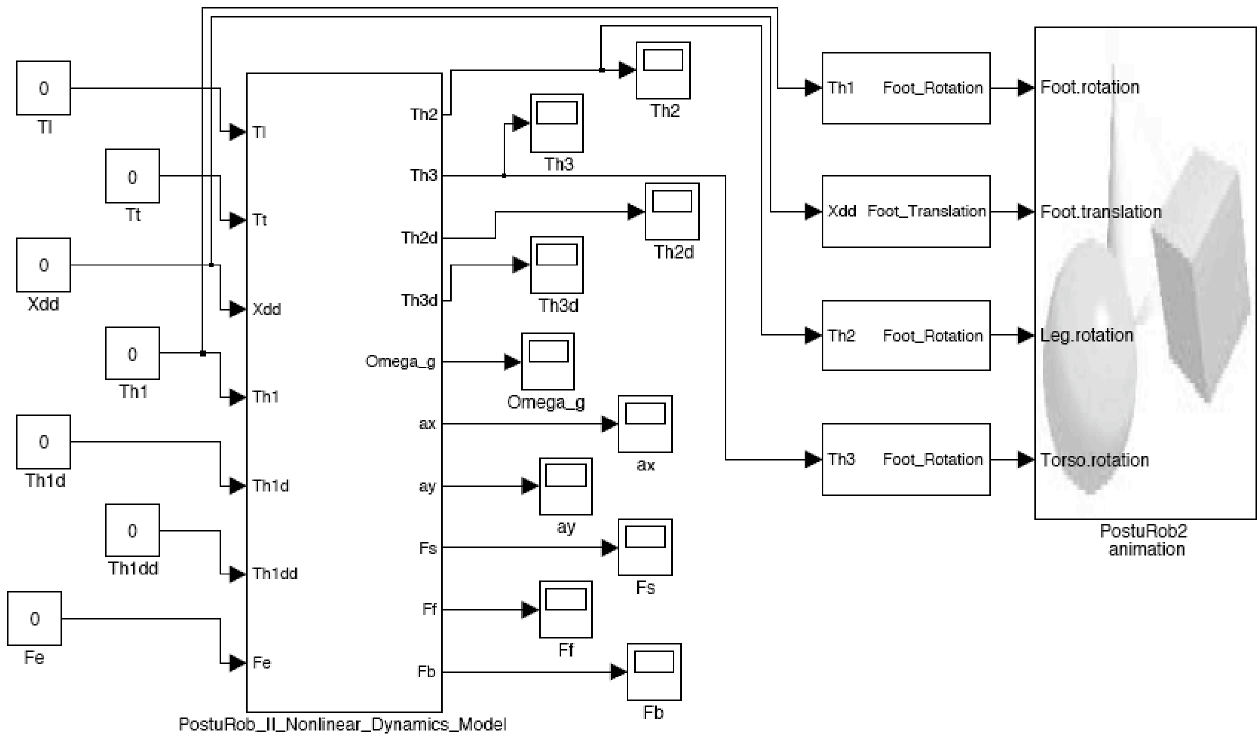

Figure 2-2 PostuRob II Simulink model.

2.5 Animation

2.5.1 Description

To obtain a better visualization of the simulation results, a virtual model of PostuRob II is created and animated using the *Virtual Reality Toolbox* and the *Virtual Reality Builder*. The virtual reality toolbox is a solution for interacting with virtual reality models of dynamic systems over time. It extends the capabilities of MATLAB and Simulink into the world of virtual reality graphics. With the virtual reality builder one can create virtual worlds or three-dimensional scenes using standard Virtual Reality Modeling Language (VRML) technology. The dynamics of the system are defined with MATLAB and Simulink using the S-function. By signals from the Simulink environment; animation of the three-dimensional scenes are obtained.

2.5.2 Animation File

Figure 2-3 shows PostuRob II virtual model. This model consists of a box that represents the foot, and two cylinders corresponding to the leg and the torso. The foot is allowed to move horizontally, and to rotate about the centerline of its upper surface. The leg, which is represented by the red cylinder, is defined as a child with respect to the foot, and it is allowed to rotate about its lower tip. The same relation exists between the torso, the yellow cylinder, and the leg, i.e. the torso is a defined as a child related to the leg, and it rotates about its lower tip too. The parent-child relation between these parts ensures that the motion of any part will affect all of its children. This is used to overcome the fact that there are no joints available in the virtual reality builder environment.

The only parameters used to create PostuRob II virtual model are the lengths of the three links, other parameters as masses and inertias are of no importance here, since this model depends in its execution on the results generated from the S-function. The motion of the foot, which represents the disturbances affecting the system are defined by the application; the foot inclination angle is used as passed by the application in radians, while the foot horizontal displacement is obtained by a double integration of the disturbing base acceleration.

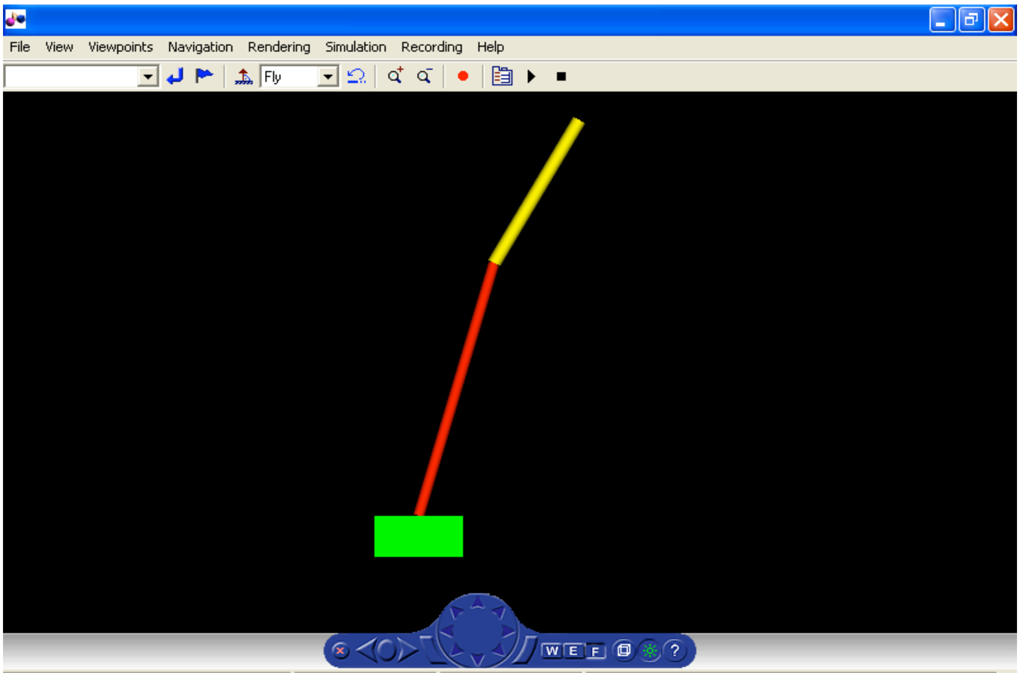

Figure 2-3 PostuRob II virtual model animation scope.