

Symbolic Computation and Simulation of a Spatial Pendulum

Vinay Gupta and S.K. Saha

Abstract

This paper describes the formulation of equations of motion of a spatial pendulum using Euler-Lagrange formulation. Note that the pendulum has one spherical joint with three rotational degrees of freedom. Out of 12 possible Euler-Angles combinations available to represent the spherical joint, the ZYZ Euler angles were used to describe the rotations of the link. The equations of motions involving partial differentiations, etc. were obtained using symbolic computations. Such symbolic expressions help in interpreting the effect of parameters on the overall dynamic behavior of the system at hand. For the spatial pendulum, simulations were performed by varying the different rotational angles. The Coriolis and centrifugal forces were evaluated symbolically. After four-stage simplifications about 1000 terms could be brought down to less than 100 terms using the “simplify” command of MATLAB in a systematic way which will be shown in the paper.

Keywords: Spatial pendulum, symbolic computations, forward dynamics

1 Introduction

In computer-aided control system design, both symbolic and numeric manipulation tools for equations play important roles. The modelling and simulation of mechanical systems are typical examples where such mixed manipulation is effective [1]. The demonstration of the various applications of symbolic computation in robot manipulator design and analysis is presented by [2]. The general purpose computer algebra tools packages such as Mathematica, REDUCE and MAPLE for the manipulation of mechanical system dynamics equations are used extensively. Some work based on this packages, has reported- for example Mathematica [1], REDUCE [3], SYMRO+ [4], MAPLE [5, 6]. This paper deals with the dynamic modelling and simulation of a spatial pendulum using symbolic and numeric computations. The spatial pendulum consists of a rigid body, supported at a fixed pivot, with three rotational degrees of freedom. The mathematical model of the system is systematically given through the Euler-Lagrange formulation with some symbolic operations. The simulation is treated as initial value problem to the equations of motion using the Runge-kutta algorithm of MATLAB [7]. Pendulum models are useful for both pedagogical and research reasons. A Pendulum represents a physical system that can be viewed as one type of mechanical system that arise in, for example, robotics and spacecraft applications [8]. The spherical joint can also be represented as a set of three intersecting revolute joints meeting at a joint called Euler-angle-joints [9]. In this paper, the equations of motion of the pendulum are

Vinay Gupta

Department of Mechanical Engineering, IEC Group of Institutions, Greater Noida-201306,

E-mail: vinayguptaiec@gmail.com

S.K. Saha(Corresponding author)

Department of Mechanical Engineering, Indian Institute of Technology Delhi, New Delhi-110016,

E-mail: saha@mech.iitd.ac.in

derived in symbolic form using the symbolic computation functions of MATLAB. The expressions were used to perform the simulation of pendulum when it was allowed to fall freely under gravity.

2 Euler-Lagrange Formulation

The dynamic model of a mechanical system can be derived by using the concept of generalized coordinates and a scalar function called Lagrangian. The Lagrangian is defined as the difference between the kinetic and potential energy of the mechanical system under study [10]. i.e.

$$L=T-U \quad (1)$$

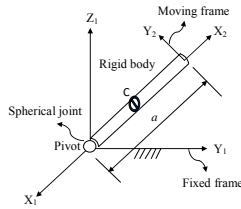


Figure 1: Spatial pendulum

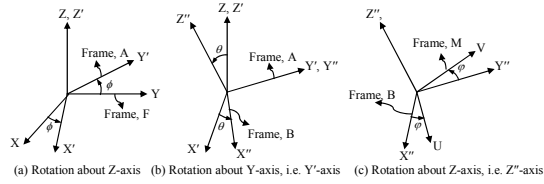


Figure 2: ZYZ Euler angles

where L denotes the Lagrangian, and T and U are respectively the total kinetic and potential energy of the system at hand. The kinetic energy depends on both configuration, i.e., the position and orientation, and velocity of the system, whereas the potential energy depends only on the configuration of the links. The Euler-Lagrange equations of motion are then given by

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = \phi_i \quad \text{for } i = 1, \dots, n \quad (2)$$

where n is the number of independent generalized coordinates used to define system's configuration, and q_i 's and ϕ_i 's are the generalized coordinates and generalized forces. For the spatial pendulum at hand shown in Fig. (1), the degree of freedom (DOF) is three. Hence, $n=3$. As shown in [10], Eqs. (1) and (2) can be obtained as

$$\mathbf{I}\ddot{\mathbf{h}} + \mathbf{h} + \boldsymbol{\gamma} = \boldsymbol{\tau} \quad (3)$$

where \mathbf{I} is $n \times n$ generalized inertia matrix (GIM) and the n -dimensional vectors \mathbf{h} , $\boldsymbol{\gamma}$ and $\boldsymbol{\tau}$ are defined below:

$\mathbf{h} \equiv [h_1, \dots, h_n]^T$: the n -dimensional vector of centrifugal and Coriolis accelerations;

$\boldsymbol{\gamma} \equiv [\gamma_1, \dots, \gamma_n]^T$: the n -dimensional vector of gravitational accelerations;

$\boldsymbol{\tau} \equiv [\tau_1, \dots, \tau_n]^T$: the n -dimensional vector of generalized forces.

2.1 Kinematics of spatial pendulum

The Spatial pendulum of Fig. (1) has a rigid body supported by a fixed frictionless pivot. The supporting pivot allows three degrees of rotational freedom. Out of the two frames shown in Fig. (1) the fixed frame is located at the origin of the pendulum and the moving frame is attached to the end of the body. The inertial parameters, i.e., the mass, the inertia tensor and the position of the center of gravity of the pendulum were provided with respect to the inertial frames. Note that the Euler angle rotations (ZYZ) were used to represent DOF of the system and all the vectors expressed in the body fixed-frame were transformed to the fixed-frame using the ZYZ rotational angles. The Euler angles constitute a minimal representation for the orientation of a rigid body which is essentially composed of three elementary rotations with respect to the axes of the current frames. The three rotation angles are denoted here as ϕ , θ and φ , as shown in Fig. (2).

2.2 Dynamics of spatial pendulum

The motion of the pendulum was described by ZYZ Euler angle rotations. The associated rotation matrix denoting the orientation of rigid body with respect to the fixed-frame is given by [10].

$$\mathbf{Q} \equiv \begin{bmatrix} C\phi C\theta C\varphi - S\phi S\varphi & -C\phi C\theta S\varphi - S\phi C\varphi & C\phi S\theta \\ S\phi C\theta C\varphi + C\phi S\varphi & -S\phi C\theta S\varphi + C\phi C\varphi & S\phi S\theta \\ -S\theta C\varphi & S\theta S\varphi & C\theta \end{bmatrix} \quad (4)$$

where ‘C’ and ‘S’ stand for Cosine and Sine functions, respectively. The associated angular velocity of the pendulum is denoted by $\boldsymbol{\omega}$, which is expressed as

$$\boldsymbol{\omega} = \mathbf{L}\dot{\boldsymbol{\theta}} \quad (5)$$

where $\boldsymbol{\theta} \equiv [\phi \ \theta \ \varphi]^T$ the angles ϕ, θ and φ are shown in Fig. (2) and \mathbf{L} is the 3x3 transformation matrix represented as [10]

$$\mathbf{L} \equiv \begin{bmatrix} 0 & -S\phi & S\theta C\phi \\ 0 & C\phi & S\theta S\phi \\ 1 & 0 & C\theta \end{bmatrix} \quad (6)$$

2.3 Generalized Inertia matrix (GIM)

The generalized inertia matrix (GIM) of the spatial pendulum shown in Fig. (1) can be derived from its kinetic energy expression as

$$T = \frac{1}{2} m \dot{\mathbf{c}}^T \dot{\mathbf{c}} + \frac{1}{2} \boldsymbol{\omega}^T \mathbf{I}^c \boldsymbol{\omega} \quad (7)$$

where $\dot{\mathbf{c}}$ represents the 3-dimensional vector of linear velocity of the mass center, \mathbf{c} in Fig. (1), of the pendulum, $\boldsymbol{\omega}$ is the 3-dimensional vector of angular velocity, and

\mathbf{I}^c is the 3x3 inertia tensor about the C . Using the 3-dimensional vector $\boldsymbol{\theta}$ as after Eq. (5), as the vector of independent generalized coordinates and the expression of $\boldsymbol{\omega}$ in Eq.(5), one can write

$$\dot{\mathbf{c}} = \mathbf{L}\dot{\boldsymbol{\theta}} \times \mathbf{c} \quad (8)$$

In Eq. (7), the 3x3 cross-product tensor [5] with respect to vector \mathbf{c} is a function of θ , and \mathbf{L} is given by Eq. (6). Substituting Eq. (8) in Eq. (7), the kinetic energy of the pendulum is given by

$$T = \frac{1}{2} \dot{\boldsymbol{\theta}}^T \mathbf{I} \dot{\boldsymbol{\theta}} \quad (9)$$

Where \mathbf{I} is the 3x3 GIM of the pendulum at hand.

If the vector \mathbf{c} denoting the mass center position can be represented as $\mathbf{c} = \frac{a}{2} \mathbf{i}_2$, where \mathbf{i}_2 is the unit vector along the axis X_2 , Fig. (1) then the expression of the GIM is obtained as

$$\mathbf{I}_{3 \times 3} = \frac{ma^2}{4} \mathbf{L}^T \mathbf{L} - \frac{ma^2}{4} \mathbf{L}^T \mathbf{i}_2 \mathbf{i}_2^T \mathbf{L} + \mathbf{L}^T \mathbf{I}^c \mathbf{L} \quad (10)$$

The elements of GIM were then computed using the symbolic computation functions of MATLAB [7] by inputting the following values:

Mass of the pendulum (rigid body) = m ; Length = a ; $\mathbf{i}_2 = [1 \ 0 \ 0]$; Diameter = d and

$$[\mathbf{I}^c]_2 = \begin{bmatrix} \frac{md^2}{8} & 0 & 0 \\ 0 & \frac{md^2}{16} + \frac{ma^2}{3} & 0 \\ 0 & 0 & \frac{md^2}{16} + \frac{ma^2}{3} \end{bmatrix}$$

Note that $[\mathbf{i}_2]_2$ and $[\mathbf{I}^c]_2$ are the vector \mathbf{i}_2 and tensor \mathbf{I}^c represented in the body-fixed frame 2 shown in Fig. (1). Defining all the variables in MATLAB as “syms,” i.e., “symbolic” the elements of the GIM were evaluated, as shown in Fig. (3). The few elements resolve after symbolic operation were shown in Appendix A.2.

2.4 Centrifugal and Coriolis accelerations

The components of the 3-dimensional vector \mathbf{h} in Eq. (3) representing centrifugal and Coriolis acceleration of the spatial pendulum at hand are evaluated as [10]

$$h_i \equiv \sum_{j=1}^3 \sum_{k=1}^3 \left(\frac{\partial i_{ij}}{\partial \theta_k} - \frac{1}{2} \frac{\partial i_{jk}}{\partial \theta_i} \right) \dot{\theta}_j \dot{\theta}_k \quad \text{for } i=1,2,3 \quad (11)$$

where, θ_j or θ_k in the above equations corresponds to three DOF's of the pendulum or the 3 rotations, namely ϕ , θ and φ . Since the expressions of \mathbf{h} -vector include partial differentiation of the inertia matrix elements which is not easy to compute by hand before any numerical simulation, they were computed symbolically by using the symbolic operations of MATLAB[7]. For the symbolic computation, instead of defining the parameters numerically they were represented as symbolic using the command 'syms.' Further, the elements of vector \mathbf{h} computed symbolically were lengthy containing more than 1000 terms. Hence the four-stage simplification was carried out using the command 'simplify.' That way, the number of terms in the element of vector \mathbf{h} were reduced under hundred. Terms before and after simplification were given in Appendix A.1. The expressions generated by the symbolic operations were then exported to the numerical computation environment. The Fig. (3) shows the systematic use of symbolic computation of vector \mathbf{h} and sequence of simplifications.

```

% To calculate h vector
syms th1 th2 th3 th1d th2d th3d m a d
Qz=[cos(th1)  -sin(th1)  0
     sin(th1)  cos(th1)  0
     0         0         1];
Qy= [cos(th2)  0  sin(th2)
     0         1  0
     -sin(th2) 0  cos(th2)];
Qzz= [cos(th3)  -sin(th3)  0
     sin(th3)   cos(th3)  0
     0         0         1];
%Q=rotation matrix
Q = Qz*Qy*Qzz;
Q1=simplify(Q);  → Stage 1 simplification
Icff=Q1*Ic*transpose (Q1);
Icff1=simplify(Icff);  → Stage 2 simplification
L=[0  -sin(th1)  sin(th2)*cos(th1)
   0  cos(th1)  sin(th2)*sin(th1)
   1  0         cos(th2)          ];
% I=(GIM)
I=(0.25*m*a*a*transpose(L)*L)-(0.25*m*a*a*transpose(L)*[1;0;0]*[0 0
1]*L)+(transpose(L)*Icff1*L);
I1=simplify(I);  → Stage 3 simplification
h1=(diff(I1(1,1),th1)-.5*diff(I1(1,1),th1))*th1d*th1d+....
h2=(diff(I1(2,1),th1)-.5*diff(I1(1,1),th2))*th1d*th1d+....
h3=(diff(I1(3,1),th1)-.5*diff(I1(1,1),th3))*th1d*th1d+....
h11=simplify(h1);
h22=simplify(h2);
h33=simplify(h3); } → Stage 4 simplification

```

Figure 3: Symbolic computation of vector \mathbf{h} in (MATLAB)

2.5 Gravitational accelerations

The terms for the gravitational accelerations can be obtained from [5] as

$$\gamma_i \equiv -\sum_{j=1}^3 m_j \mathbf{g}^T \frac{\partial \mathbf{c}_j}{\partial \theta_i}, \text{ for } i = 1, 2, 3 \quad (12)$$

Which for the spatial pendulum leads to

$$\gamma_1 = 0; \gamma_2 = \frac{mag}{2} C\theta C\varphi; \gamma_3 = -\frac{mag}{2} S\theta S\varphi \quad (13)$$

3 Simulation

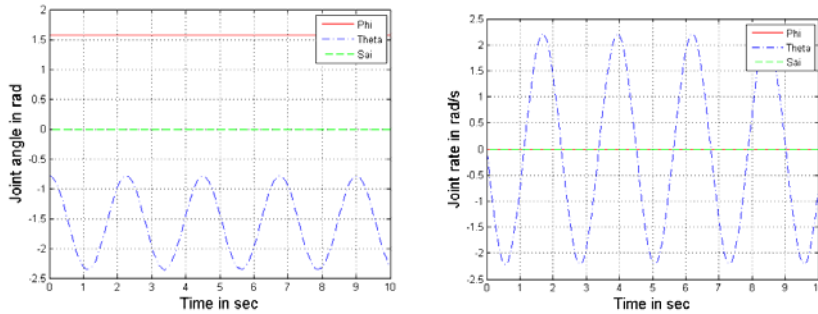
Simulation was performed using the equations of motion derived in section 2 to predict the behavior of the spatial pendulum. The following numerical values were used for the numerical simulation.

Length, $a=1\text{m}$; mass, $m= 2.5 \text{ kg}$; Diameter, $d=.02$; Acceleration due to gravity, $=9.8\text{m/s}^2$; Initial values, $\phi = \frac{\pi}{2}$; $\theta = -\frac{\pi}{4}$; $\varphi = 0$; Input torques, τ of Eq. (3)=0. No

input torques were considered, as the interest was to simulate the free-fall behavior of the system. Fig. 4 (a) shows the behavior of the spatial pendulum. Note that the behavior of Fig. (4) corresponds to the behavior of a planar pendulum where a rigid-body is attached to the fixed frictionless pivot using a revolute joint, as the initial condition were chosen that way. In order to visualize a spatial behavior, the initial

conditions were changed to $\phi = -\frac{\pi}{2}$; $\theta = -\frac{\pi}{4}$; $\varphi = -\frac{\pi}{4}$ for which the results are

shown in Fig. (5). Results of Fig. (5), clearly demonstrates the motion in 3-dimensional space. The results of Figs. (4) and (5) were validated with the algorithm developed by [9] using a full numerical computer algorithm compared to symbolic-numeric algorithm proposed here. Fig 5(a) shows the displacement in all three directions, showing the spatial motion of the pendulum. Fig. 4(b), and Fig 5(b), shows joint rates for respective Euler angle rotation. For $\theta = 0$ (the second Euler angle rotation) the system shows singular behavior.



(a) Joint angles

(b) Joint rates

Figure 4: Simulation for $\phi = \frac{\pi}{2}$; $\theta = -\frac{\pi}{4}$; $\varphi = 0$ Euler angle rotations

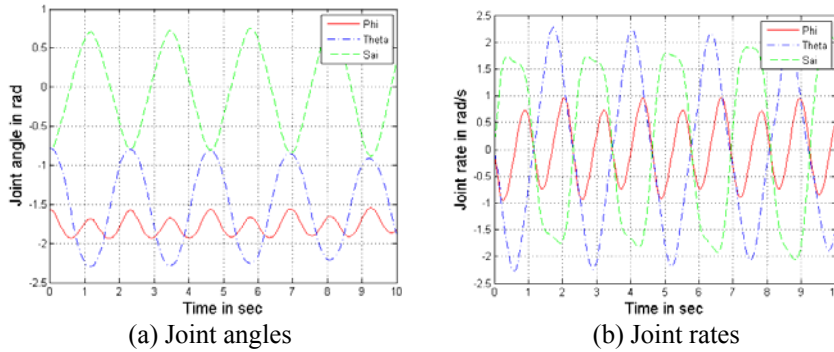


Figure 5: Simulation for $\phi = -\frac{\pi}{2}; \theta = -\frac{\pi}{4}; \varphi = -\frac{\pi}{4}$ Euler angle rotation

4 Conclusions

A systematic development of the dynamic equations of motion of a spatial pendulum has been presented in this paper using Euler-Lagrange formulation. The symbolic operations using MATLAB's 'syms' and 'simplify' commands were performed to compute GIM, centrifugal and Coriolis acceleration terms. Simulations were performed using the above computations. Even though the symbolic-cum-numerical algorithm proposed for a pendulum was used for a simple simulation. The concept can be used for stability study of the same pendulum or can be used for more complex mechanical system with spatial motions.

References

- [1] T. Hoshino, and K. Furuta, "Modeling and Simulation of Mechanical Systems- Combination of a symbolic Computation Tool and M_ATX," International Symposium on Computer aided Control System Design, USA, August 22-27, 1999.
- [2] N. Rentia, and N. Vira, "Why Symbolic computation in Robotics?" Elsevier, Computers in Industry, vol. 17, pp. 49-62, 1991.
- [3] J. Kopic and M. C. Leu, "Computer generation of robot dynamics and related issues," Journal of Robotic Systems, vol. 3, issue 3, pp. 301-319, 1986.
- [4] W. Khalil, and D. Greusot, "SYMORO+: A system for the symbolic modelling of robots," Robotica, vol. 15, pp. 153-161, 1997.
- [5] P. I. Corke, "A symbolic and numeric procedure for manipulator rigid-body dynamic significance analysis and simplification," Robotica, vol. 16, pp. 589-594, 1998.
- [6] H. Shah, S. Tripathi, L. Lee, and V. Krovi, "Role of automated symbolic generation of equations of motion in mechanism and robotics education," Proceedings of ASME- IDETC/CIE, Canada, pp. 1-8, August 15-18, 2010.
- [7] Matlab, MathWorks, Inc, Massachusetts, USA, 2008.
- [8] N. A. Chaturvedi et. al., "Nonlinear Dynamics of the 3D Pendulum," Journal of Non-linear Science, vol. 21, pp. 3-32, 2010.

- [9] S. Shah, S. K. Saha, and J. K. Dutt, "Spherical Joint representation In Multibody Systems," Multibody Dynamics, ECCOMAS Thematic Conference, Warsaw, Poland, pp. 1-16, 29 June-2 July 2009.
- [10] S. K. Saha, *Introduction to Robotics*. New Delhi: Tata McGraw Hills, 2008.

Appendix

A.1 Term of vector h

h1 terms before simplification: Are more than 1000

h1 terms after four stage simplification:

$$\begin{aligned} & 1/10000*m*(1250*th1d*th2d*d^2*cos(th2)*cos(th3)^2*sin(th2)+6666*th1d*th2d*cos(th2)*a^2*cos(th3) \\ & ^2*sin(th2)+1250*th1d*th3d*d^2*cos(th3)*sin(th3)-1250*th1d*th3d*d^2*cos(th2)^2*cos(th3)*sin(th3)- \\ & 6666*th1d*th3d*cos(th3)*a^2*sin(th3)+6666*th1d*th3d*cos(th2)^2*a^2*cos(th3)*sin(th3)+1250*a^2*c \\ & os(th1)*th1d*th2d+625*cos(th2)*cos(th3)*sin(th3)*th2d^2*d^2- \\ & 3333*cos(th2)*cos(th3)*sin(th3)*th2d^2*a^2+1250*th3d*th2d*sin(th2)*cos(th3)^2*d^2+9166*sin(th2)*t \\ & h3d*th2d*a^2- \\ & 6666*th3d*th2d*sin(th2)*cos(th3)^2*a^2+1250*th3d*th2d*a^2*cos(th1)*cos(th2)+1250*a^2*sin(th1)*si \\ & n(th2)*th1d*th3d+1250*a^2*sin(th1)*sin(th2)*cos(th2)*th3d^2). \end{aligned}$$

A.2 Element of GIM

i_{33} element of 3x3 GIM before simplification:

$$\begin{aligned} & 1/4*m*a^2*cos(th1)^2*sin(th2)^2+1/4*m*a^2*sin(th1)^2*sin(th2)^2+1/4*m*a^2*cos(th2)^2+1/4*m*a^2* \\ & cos(th1)*sin(th2)*cos(th2)+(cos(th1)*sin(th2)*(1/8*(cos(th1)*cos(th2)*cos(th3)-sin(th1)*sin(th3))^2 \\ & *m*d^2+(-cos(th1)*cos(th2)*sin(th3)-sin(th1)*cos(th3))^2*(1/16*m*d^2+3333/10000*m*a^2) \\ & +cos(th1)^2*sin(th2)^2*(1/16*m*d^2+3333/10000*m*a^2))+sin(th1)*sin(th2)*(1/8*(cos(th1)*cos(th2)*c \\ & os(th3)-sin(th1)*sin(th3))*m*d^2*(sin(th1)*cos(th2)*cos(th3)+cos(th1)*sin(th3))+(-cos(th1)*cos(th2) \\ & *sin(th3)-sin(th1)*cos(th3))*(1/16*m*d^2+3333/10000*m*a^2)*(-sin(th1)*cos(th2)*sin(th3) \\ & +cos(th1)*cos(th3))+cos(th1)*sin(th2)^2*(1/16*m*d^2+3333/10000*m*a^2)*sin(th1)+cos(th2)*(- \\ & 1/8*(cos(th1)*cos(th2)*cos(th3)-sin(th1)*sin(th3))*m*d^2*sin(th2)*cos(th3)+(-cos(th1)*cos(th2) \\ & *sin(th3)-sin(th1)*cos(th3))*(1/16*m*d^2+3333/10000*m*a^2)*sin(th2)*sin(th3)+cos(th1)*sin(th2) \\ & *(1/16*m*d^2+3333/10000*m*a^2)*cos(th2))*cos(th1)*sin(th2)+(cos(th1)*sin(th2)*(1/8*(cos(th1)*cos \\ & (th2)*cos(th3)sin(th1)*sin(th3))*m*d^2*(sin(th1)*cos(th2)*cos(th3)+cos(th1)*sin(th3))+cos(th1)*cos(th2) \\ &)*sin(th3)sin(th1)*cos(th3))*(1/16*m*d^2+3333/10000*m*a^2)*(sin(th1)*cos(th2)*sin(th3)+cos(th1)*co \\ & s(th3))+cos(th1)*sin(th2)^2*(1/16*m*d^2+3333/10000*m*a^2)*sin(th1))+sin(th1)*sin(th2)*(1/8*(sin(th \\ & 1)*cos(th2)*cos(th3)+cos(th1)*sin(th3))^2*m*d^2+(sin(th1)*cos(th2)*sin(th3)+cos(th1)*cos(th3))^2*(1/ \\ & 16*m*d^2+3333/10000*m*a^2)+sin(th1)^2*sin(th2)^2*(1/16*m*d^2+3333/10000*m*a^2))+cos(th2)*(1 \\ & /8*(sin(th1)*cos(th2)*cos(th3)+cos(th1)*sin(th3))*m*d^2*sin(th2)*cos(th3)+(sin(th1)*cos(th2)*sin(th3)+ \\ & cos(th1)*cos(th3))*(1/16*m*d^2+3333/10000*m*a^2)*sin(th2)*sin(th3)+sin(th1)*sin(th2)*(1/16*m*d^2 \\ & +3333/10000*m*a^2)*cos(th2))*sin(th1)*sin(th2)+(cos(th1)*sin(th2)*(- \\ & 1/8*(cos(th1)*cos(th2)*cos(th3)-sin(th1)*sin(th3))*m*d^2*sin(th2)*cos(th3)+(- \\ & cos(th1)*cos(th2)*sin(th3)sin(th1)*cos(th3))*(1/16*m*d^2+3333/10000*m*a^2)*sin(th2)*sin(th3)+cos(t \\ & h1)*sin(th2)*(1/16*m*d^2+3333/10000*m*a^2)*cos(th2))+sin(th1)*sin(th2)*(1/8*(sin(th1)*cos(th2)*cos \\ & (th3)+cos(th1)*sin(th3))*m*d^2*sin(th2)*cos(th3)+(sin(th1)*cos(th2)*sin(th3)+cos(th1)*cos(th3))*(1/16 \\ & *m*d^2+3333/10000*m*a^2)*sin(th2)*sin(th3)+sin(th1)*sin(th2)*(1/16*m*d^2+3333/10000*m*a^2)*c \\ & os(th2))+cos(th2)*(1/8*sin(th2)^2*cos(th3)^2*m*d^2+sin(th2)^2*sin(th3)^2*(1/16*m*d^2+3333/10000 \\ & *m*a^2)+cos(th2)^2*(1/16*m*d^2+3333/10000*m*a^2))*cos(th2)] \end{aligned}$$

i_{33} element of 3x3 GIM after simplification:

$$-1/10000*m*(2500*a^2*cos(th1)*sin(th2)*cos(th2)-5833*a^2-625*d^2)].$$