Cross interactions

Why?

One of our hypotheses is that the two active arms are completely decoupled. Our main reason behind it is that the revolute joints don't transfer load between them and that the high gear ratio "blocks" every (small) disturbance on the load axle from reaching the motor axle.

This is a good first approximation because it allows us to ignore every cross-interactions between the two servo motors and design the controllers independently.

We can apply a more rigorous approach in the study of these effects before moving forward with other tasks. (Just in case the professor wants to be pedantic).

2-DOF Kinematics

Seen from the 2 load axles our system has only masses and moments of inertia, no potential energies or dissipations.

$$T = \frac{1}{2}\underline{\dot{y}}^T \left[m_{\text{ph}}\right] \underline{\dot{y}} \tag{1.}$$

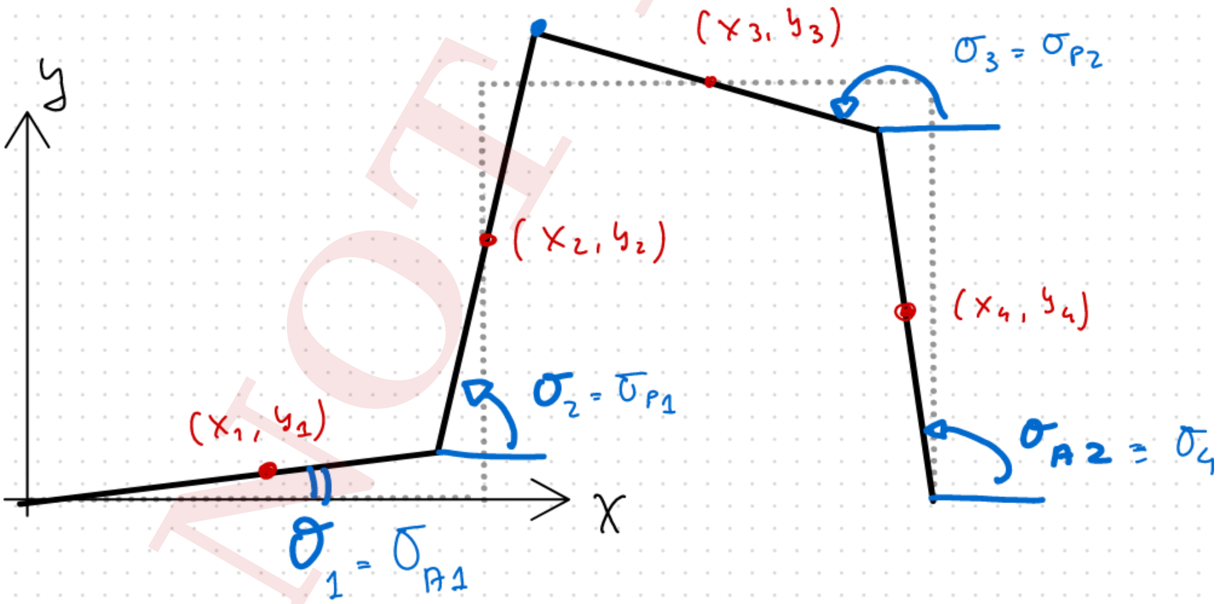
Where $\left[m_{\text{ph}}\right] = \left[m, m, J_l + r^2 J_r, m, m, J_l, m, m, J_l, m, m, J_l + r^2 J_r\right]^T$

(n.b: J_l is the moment of inertia of the links at the center of gravity, do not use the Huygens–Steiner theorem or that other moment of inertia around the pivot)

(n.n.b: J_r is the moment of inertia of rotor moved on the load axle)

and $\underline{\dot{y}} = \left[x_{l1}, y_{l1}, \theta_{l1}, x_{l2}, y_{l2}, \theta_{l2}, x_{l3}, y_{l3}, \theta_{l3}, x_{l4}, y_{l4}, \theta_{l4}\right]^T$

is the vector of independent coordinates of the bodies according to this diagram:



We can describe $\underline{\dot{y}}$ in terms of our 2 dof $\left[q_1, q_2\right]^T = \left[\theta_{\text{A1}}, \theta_{\text{A2}}\right]^T$ via a Jacobian matrix $[\Lambda_m]$:

$$\underline{\dot{y}} = [\Lambda_m]\underline{\dot{q}} \tag{2.}$$

The new system will be:

$$T = \frac{1}{2}\underline{\dot{y}}^T [m_{\text{ph}}] \underline{\dot{y}} = \frac{1}{2}\underline{\dot{q}}^T [m_0] \underline{\dot{q}} \tag{3.}$$

With $[m_0] = [\Lambda_m]^T [m_{\text{ph}}][\Lambda_m]$

The kinetic term of the Lagrangian equation becomes:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \underline{\dot{q}}}\right)^T - \left(\frac{\partial T}{\partial \underline{q}}\right)^T = [m_0]\underline{\ddot{q}} \tag{4.}$$

We can then calculate the work of the time dependent forces (our servo actuators):

$$\delta W = \delta \underline{y}_f^T \underline{f}_{\text{ph}} = [\delta\theta_{\text{A1}}, \delta\theta_{\text{A2}}] \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = [\delta q_1, \delta q_2] \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \delta \underline{q}^T \underline{f}_{\text{ph}} \tag{5.}$$

(In this case $[\Lambda_f]$ is simply the identity matrix)

$$\underline{f}(t) = [\Lambda_f]^T \underline{f}_{\text{ph}} = \underline{f}_{\text{ph}} \tag{6.}$$

Our Lagrange equation becomes:

$$[m_0]\underline{\ddot{q}} = \underline{f}(t) \tag{7.}$$

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \tag{8.}$$

With these equations we can calculate the torque applied on the load axles when a specific angular acceleration occurs (or if we invert it, the necessary torque to have a desired acceleration on the load axles).

This is a good starting point to analyze eventual cross interactions between the two axles.

The humble Jacobian of Inertia

The system has 4 links (4x3 DOF = 12 DOF) and 5 revolute joints (each cosntraing 2 DOF => −10 DOF) : in total we have 2 DOF

Our Jacobian will be a 12x2 matrix

$$[\Lambda_m] = \begin{bmatrix} \lambda_{1,1} & \lambda_{1,2} \\ \lambda_{2,1} & \lambda_{2,2} \\ \lambda_{3,1} & \lambda_{3,2} \\ \lambda_{4,1} & \lambda_{4,2} \\ \lambda_{5,1} & \lambda_{5,2} \\ \lambda_{6,1} & \lambda_{6,2} \\ \lambda_{7,1} & \lambda_{7,2} \\ \lambda_{8,1} & \lambda_{8,2} \\ \lambda_{9,1} & \lambda_{9,2} \\ \lambda_{10,1} & \lambda_{10,2} \\ \lambda_{11,1} & \lambda_{11,2} \\ \lambda_{12,1} & \lambda_{12,2} \end{bmatrix} = \begin{bmatrix} -L\sin(\theta_{\text{A1}}) & 0 \\ +L\cos(\theta_{\text{A1}}) & 0 \\ 1 & 0 \\ \lambda_{4,1} & \lambda_{4,2} \\ \lambda_{5,1} & \lambda_{5,2} \\ \lambda_{6,1} & \lambda_{6,2} \\ \lambda_{7,1} & \lambda_{7,2} \\ \lambda_{8,1} & \lambda_{8,2} \\ \lambda_{9,1} & \lambda_{9,2} \\ 0 & -L\sin(\theta_{\text{A2}}) \\ 0 & +L\cos(\theta_{\text{A2}}) \\ 0 & 1 \end{bmatrix}$$

The remaining terms needs to be calculated keeping in mind the inverse kinematics, which is cumbersome by hand. I used the MATLAB Symbolic Math Toolbox. For this reason they are ***slightly longer*** and are left in the appendix for clarity.

With our Jacobian we can then calculate the generalized mass $[m_0]$ (again with the Symbolic Toolbox)

How decoupled are we?

$$\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22}\end{pmatrix}\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\end{bmatrix} \qquad 9.$$

Let's bring back the Lagrange equation and see what happens at the "resting position":

```
rgaf(0,pi/2,true)
```

$$[m_0](\text{rest}) = \begin{pmatrix} 10.2287 & 0.0002 \\ 0.0002 & 10.2271\end{pmatrix} \qquad 10.$$

So, to obtain $1\frac{\text{rad}}{s^2}$ of acceleration on the 1st active link we need 10.2287 Nm on the 1st load axle and 0.0002 Nm on the 2nd load axle (which when accounting for the r = 70 gear ratio results in 0.1461 Nm and 2.8571×10^{-6} on the motor axles respectively)

Dually, by inverting we have:

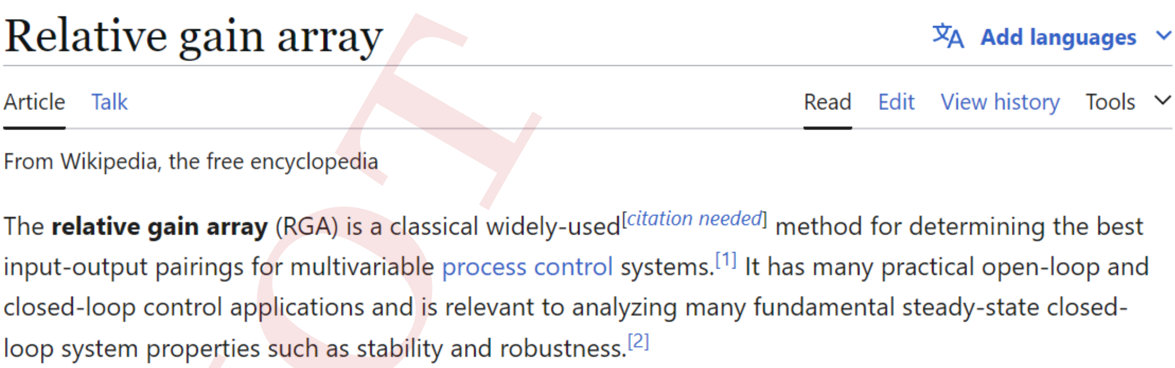
$$\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\end{bmatrix} = [m_0]^{-1}\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} \qquad 11.$$

$$[m_0]^{-1}(\text{rest}) = \begin{pmatrix} 0.0978 & -0.0000 \\ -0.0000 & 0.0978\end{pmatrix} \qquad 12.$$

Which means that when applying $1Nm$ on the 1st active load axle we'll end up with $0.0978\frac{\text{rad}}{s^2}$ on the first active link and a negligible acceleration on the second load axle. **Great!**

Be aware that symbolically inverting $[m_0]$ is useless because we can just invert the matrix after the evaluation.

Relative gain array analysis



Relative gain array

Add languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

The **relative gain array** (RGA) is a classical widely-used[*citation needed*] method for determining the best input-output pairings for multivariable process control systems.[1] It has many practical open-loop and closed-loop control applications and is relevant to analyzing many fundamental steady-state closed-loop system properties such as stability and robustness.[2]

Standard control theory method to evaluate the decoupling (or not) of a MIMO system. I only have these slides in Italian, I'm sorry guys :(

- MATRICE DEI GUADAGNI RELATIVI (RGA, RELATIVE GAIN ARRAY)

- UTILE PER: - VALUTARE IL GRADO DI INTERAZIONE
  - SCEGLIERE I MIGLIORI ACCOPPIAMENTI I/O

- IPOTESI
  1. $G(s)$ AS. STABILE
  2. $\det G(0) \neq 0$

- DEFINIZIONE

PRODOTTO DI SCHUR (O HADAMARD) (ELEMENTO PER ELEMENTO)

RGA $\boxed{\Lambda = G(0) \odot \left(G(0)^{-1}\right)'}$ MATRICE $m \times m$

- SE $\Lambda \simeq I_m \implies$ INTERAZIONE DEBOLE CON GLI ACCOPPIAMENTI $\{u_i, y_i\}$

- CASO $m=2$

$$G(0) = \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix} \qquad G(0)^{-1} = \frac{1}{\det G(0)} \begin{bmatrix} \mu_{22} & -\mu_{12} \\ -\mu_{21} & \mu_{11} \end{bmatrix}$$

$$\left(G(0)^{-1}\right)' = \frac{1}{\det G(0)} \begin{bmatrix} \mu_{22} & -\mu_{21} \\ -\mu_{12} & \mu_{11} \end{bmatrix} \qquad \det G(0) = \mu_{11}\mu_{22} - \mu_{12}\mu_{21}$$

$$\Lambda = G(0) \odot \left(G(0)^{-1}\right)' = \frac{1}{\det G(0)} \begin{bmatrix} \mu_{11}\mu_{22} & -\mu_{12}\mu_{21} \\ -\mu_{12}\mu_{21} & \mu_{11}\mu_{22} \end{bmatrix}$$

- PONENDO $\boxed{\lambda = \frac{\mu_{11}\mu_{22}}{\det G(0)}} \implies \boxed{\Lambda = \begin{bmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{bmatrix}}$

- SE $\lambda = 1 \implies \Lambda = I_2$ ✓

  $\lambda = 0 \implies \Lambda = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ MA, SCAMBIANDO GLI INGRESSI u_i $\implies \bar{\Lambda} = I_2$

09 118

- CASO $m=2$ (SEGUE)

$$\Lambda = \begin{bmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{bmatrix} \qquad \bar{\Lambda} = \begin{bmatrix} 1-\lambda & \lambda \\ \lambda & 1-\lambda \end{bmatrix}$$

A COPPIE $\{u_1, y_1\}, \{u_2, y_2\}$ B COPPIE $\{u_1, y_2\}, \{u_2, y_1\}$

- REGOLA DI SCELTA

$\lambda = 1 \implies$ A

$\lambda = 0 \implies$ B

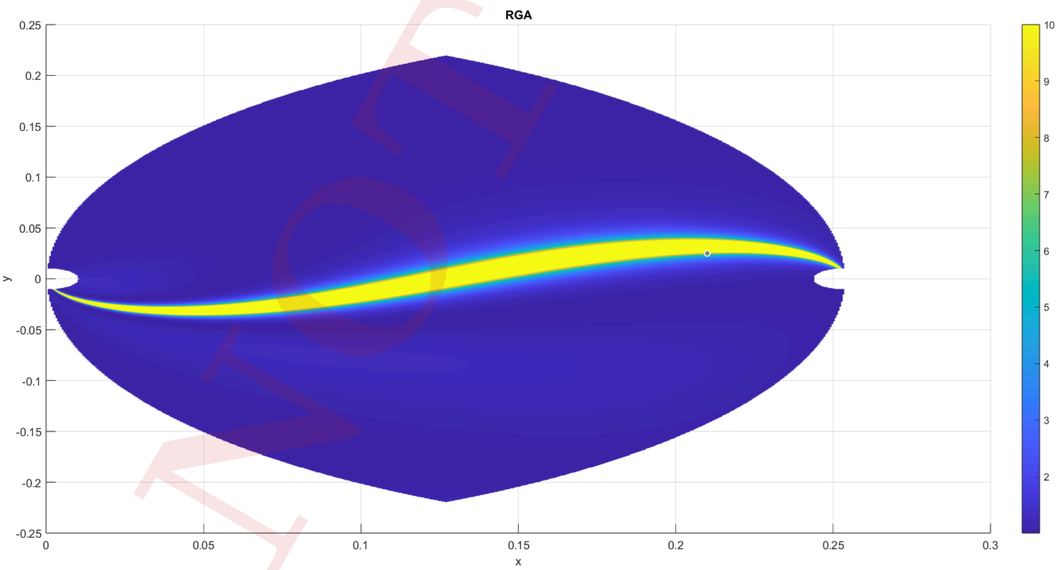
$\lambda > \frac{1}{2} \implies$ MEGLIO A CHE B

$\lambda < \frac{1}{2} \implies$ MEGLIO B CHE A

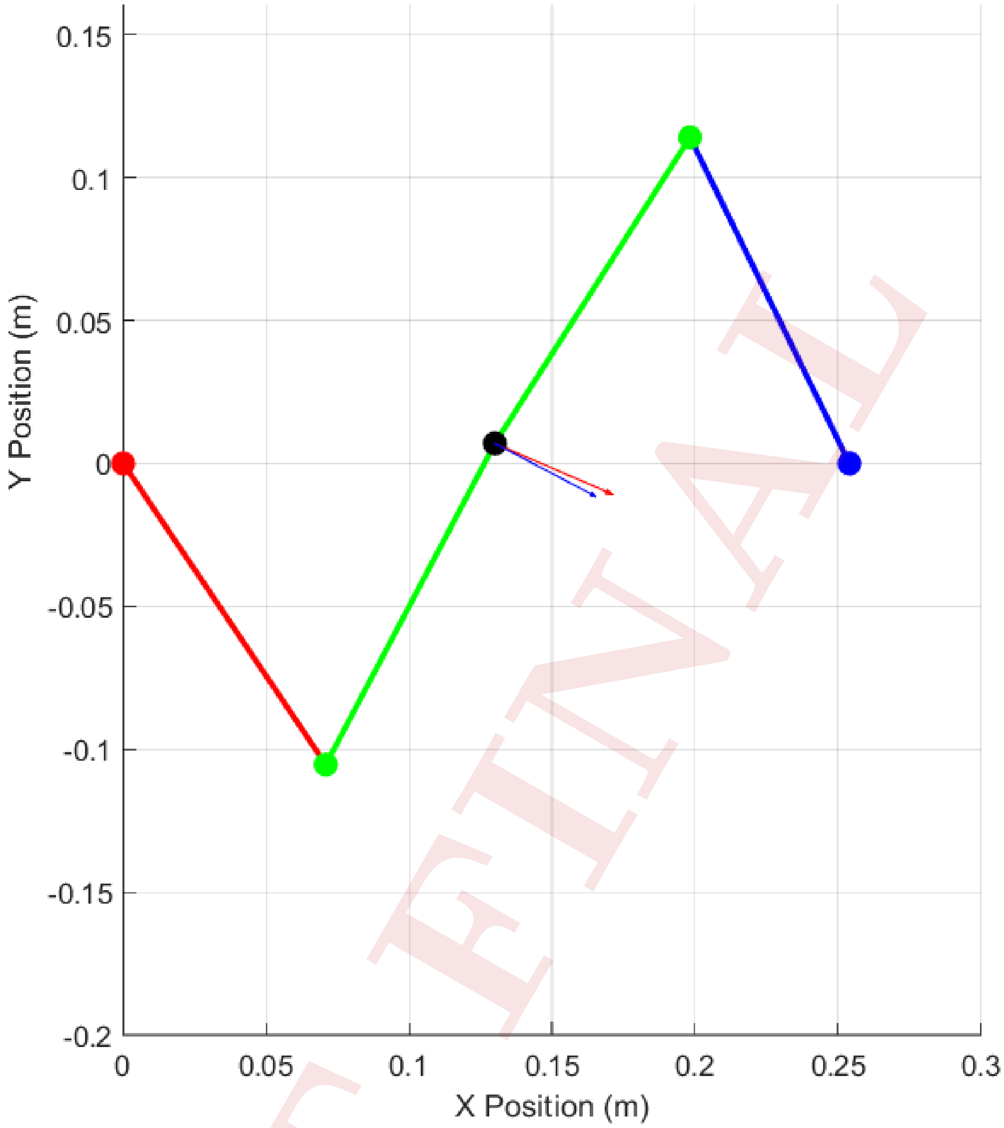
$\left.\begin{matrix} \lambda \gg 1 \\ \lambda \ll 0 \end{matrix}\right\} \implies$ FORTE INTERAZIONE

Analysis Approach

Let's plot λ_{rga} on the Cartesian workspace. When $\lambda_{\text{rga}} \approx 1$ it means we have high decoupling, otherwise we have cross-coupling between the active links:



We can see how we have good decoupling everywhere but in the singularity region, which is great and makes sense if you think about it:



When near the singularity, every torque applied on one axle is fully transferred to the other axle = high RGA!

Appendix

Jacobian terms

```
jac(4,1) = - L*sin(ta1) + (sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(5,1) = L*cos(ta1) - (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(6,1) = -(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2)))/(4*L*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
```

```
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(7,1) = - L*sin(ta1) + (sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) -
(sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1)
+ 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2));

jac(8,1) = L*cos(ta1) - (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) +
(cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1)
+ 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2));

jac(9,1) = (2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2)))/(4*L*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(4,2) = -(sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(5,2) = (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));
jac(6,2) = (2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2)))/(4*L*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(7,2) = - (sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) + (sin(angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)) - acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(8,2) = (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1)
```

```
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) - (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)) - acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));

jac(9,2) = -(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2)))/(4*L*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2));
```

Generalized Mass

```
mm(1,1) = Ja + M*(L*cos(ta1) - (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i))
+ acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))
+ (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)))*(L*cos(ta1) -
(cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) -
cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2)
- cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)) + (cos(conj(acos((L^2*(cos(ta2)
- cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)^(1/2))) + M*(L*cos(ta1) - (cos(angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) +
2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2)))*(L*cos(ta1) - (cos(conj(acos((L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1)
- sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)^(1/2))) + (L^2*M*cos(ta1)^2)/4 + (L^2*M*sin(ta1)^2)/4 - M*(L*sin(ta1)
- (sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) -
cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) + (sin(angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) +
2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2)))*(- L*sin(ta1) + (sin(conj(acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
```

```
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)) + (sin(conj(acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1)
- sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)^(1/2))) + M*(L*sin(ta1) - (sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 -
1i)) + acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/
(4*L^2))^(1/2)))*(L*sin(ta1) - (sin(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1)
- sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2))) + (Jp*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta1)*(sin(ta1) - sin(ta2)))^2)/(8*L^2*conj((1 - (L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2));
```

```
mm(1,2) = - M*((sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(4*(L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) - (sin(angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) +
2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2)))*(- L*sin(ta1) + (sin(conj(acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)) + (sin(conj(acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1)
- sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2))) + M*((cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))
- (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) -
```

```
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)))*(L*cos(ta1) -
(cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) -
cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)) + (cos(conj(acos((L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1)
- sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2))) + (M*cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1)
- sin(ta2)))*(L*cos(ta1) - (cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1)
- sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2)) + (M*sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i))
+ acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1)
- sin(ta2)))*(L*sin(ta1) - (sin(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1)
- sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2)) - (Jp*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta1)*(sin(ta1) - sin(ta2)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*L^2*conj((1 - (L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2));


mm(2,1) = M*((sin(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1) - sin(ta2))*(- 1 -
1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2)) + (sin(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)^(1/2)))*(L*sin(ta1) - (sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i))
+ acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) +
(sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1)
```

```
+ 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))) + M*((cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1)
- sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2)) - (cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)^(1/2)))*(L*cos(ta1) - (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i))
+ acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) +
(cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1)
+ 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))) + (M*cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1)
- sin(ta2)))*(L*cos(ta1) - (cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta1)*(sin(ta1) -
sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))))/
(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/
(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)) +
(M*sin(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2)))*(L*sin(ta1) -
(sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1)
+ 2) + 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)^(1/2)) - (Jp*(2*L^2*sin(ta1)*(cos(ta2) - cos(ta1) + 2)
+ 2*L^2*cos(ta1)*(sin(ta1) - sin(ta2)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*L^2*conj((1 - (L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2)) ;
```

```
mm(2,2) = Ja + M*((cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(4*(L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) - (cos(angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
```

```
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) +
2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2)))*((cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1)
- sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2)) - (cos(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)^(1/2))) + (L^2*M*cos(ta2)^2)/4 + M*((sin(conj(acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) - angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2
+ L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)) + (sin(conj(acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(-
1 - 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1)
- sin(ta2))))/(4*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2)))*((sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) +
acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/
(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2))))/(4*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) -
(sin(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) - acos((L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1)
+ 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2))))/(8*(L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)/(4*L^2))^(1/2))) + (L^2*M*sin(ta2)^2)/4 + (Jp*(2*L^2*sin(ta2)*(cos(ta2)
- cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) - sin(ta2)))^2)/(8*L^2*conj((1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/
(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)*(1
- (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) +
(M*cos(angle(L*(sin(ta1) - sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L)))*cos(conj(acos((L^2*(cos(ta2) -
cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1)
- sin(ta2))*(- 1 - 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) +
2*L^2*cos(ta2)*(sin(ta1) - sin(ta2)))^2)/(64*conj((1 - (L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1)
+ 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)*(1 - (L^2*(cos(ta2) - cos(ta1) +
2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/(4*L^2))^(1/2)) + (M*sin(angle(L*(sin(ta1) -
sin(ta2))*(- 1 - 1i)) + acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1)
- sin(ta2))^2)^(1/2)/(2*L)))*sin(conj(acos((L^2*(cos(ta2) - cos(ta1) + 2)^2 +
L^2*(sin(ta1) - sin(ta2))^2)^(1/2)/(2*L))) + angle(L*(sin(ta1) - sin(ta2))*(- 1
- 1i)))*(2*L^2*sin(ta2)*(cos(ta2) - cos(ta1) + 2) + 2*L^2*cos(ta2)*(sin(ta1) -
sin(ta2)))^2)/(64*conj((1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)/(4*L^2))^(1/2))*(L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) -
sin(ta2))^2)*(1 - (L^2*(cos(ta2) - cos(ta1) + 2)^2 + L^2*(sin(ta1) - sin(ta2))^2)/
(4*L^2))^(1/2));
```