Contents

6 Analytical Dynamics

6.1 Generalized Coordinates and Constraints

Consider a system of N particles: $\{S\} = \{P_1,\ P_2,\ ...\ P_i\ ...\ P_N\}$. The position vector of the ith particle in cartesian reference frame is $\mathbf{r}_i = \mathbf{r}_i(x_i,\ y_i, z_i)$ and can be expressed as

$$\mathbf{r}_i = x_i\imath + y_i\jmath + z_i\mathbf{k}, \quad i = 1, 2, ..., N.$$

The system of N particles requires $n = 3\,N$ physical coordinates to specify the its position. To analyze the motion of the system in many cases, it is more convenient to use a different set of variables than the physical coordinates. Let consider a set of variables $q_1,\ q_2,\ .\ .\ .\ ,\ q_{3N}$ related to the physical coordinates by

$$\begin{aligned}
x_1 &= x_1\left(q_1, q_2, ..., q_{3N}\right), \\
y_1 &= y_1\left(q_1, q_2, ..., q_{3N}\right), \\
z_1 &= z_1\left(q_1, q_2, ..., q_{3N}\right), \\
& . \\
& . \\
& . \\
x_{3N} &= x_{3\,N}\left(q_1, q_2, ..., q_{3N}\right), \\
y_{3N} &= y_{3\,N}\left(q_1, q_2, ..., q_{3N}\right), \\
z_{3N} &= z_{3\,N}\left(q_1, q_2, ..., q_{3N}\right).
\end{aligned}$$

The *generalized coordinates*, $q_1,\ q_2,\ .\ .\ .\ ,\ q_{3N}$, are the set of variables that can completely describe the position of the dynamical system.

The *configuration space* is the space extended across the generalized coordinates.

If the system of N particles has m constraint equations acting on it, the system can be represented uniquely by p *independent generalized coordinates* q_k, $(k = 1,\ 2\ ,\ .\ .\ .\ ,\ p)$, where

$$p = 3N - m = n - m.$$

The number p is called the number of degrees of freedom of the system.

The number of *degrees of freedom* is the minimum number of independent coordinates necessary to describe the dynamical system uniquely.

The *generalized velocities*, denoted by $\dot{q}_k(t)$ $(k = 1, 2, \ldots, n)$, represent the rate of change of the generalized coordinates with respect to time.

The *state space* is the $2n$-dimensional space spanned by the generalized coordinates and generalized velocities.

The constraints are generally preponderant as a result of contact between bodies, and they limit the motion of the bodies upon which they act. A *constraint equation* and a *constraint force* are related with a constraint. The constraint force is the joint reaction force and the constraint equation represents the kinematics of the contact.

Consider a smooth surface of equation

$$f\,(x, y, z, t) = 0, \tag{6.1}$$

where f has continuous second derivatives in all its variables. A particle P is subjected to a constraint of moving on the smooth surface described by Eq. (6.1). The constraint equation $f\ (x,\ y,\ z,\ t) = 0$ represents a *configuration constraint.*

The motion of the particle over the surface can be viewed as the motion of an otherwise free particle subjected to the constraint of moving on that particular surface. Hence, $f\ (x,\ y,\ z,\ t) = 0$ represents a constraint equation.

For a dynamical system with n generalized coordinates, a configuration constraint can be described as

$$f\,(q_1, q_2, ..., q_n, t) = 0. \tag{6.2}$$

The differential of the constraint f, given by Eq. (6.1), in terms of physical coordinates is

$$df = \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy + \frac{\partial f}{\partial z}dz + \frac{\partial f}{\partial t}dt = 0. \tag{6.3}$$

The differential of the constraint f, given by Eq. (6.2), in terms of the generalized coordinates is

$$df = \frac{\partial f}{\partial q_1}dq_1 + \frac{\partial f}{\partial q_2}dq_2 + ... + \frac{\partial f}{\partial q_n}dq_n + \frac{\partial f}{\partial t}dt = 0. \tag{6.4}$$

Equations (6.3) and (6.4) are called *constraint relations in Pfaffian form.* A constraint in Pfaffian form is a constraint that is represented in the form of differentials.

The *constraint equations in velocity form* (or *velocity constraints* or *motion constraints*) are obtained dividing Eqs. (6.3) and (6.4) by dt

$$\frac{df}{dt} = \frac{\partial f}{\partial x}\dot{x} + \frac{\partial f}{\partial y}\dot{y} + \frac{\partial f}{\partial z}\dot{z} + \frac{\partial f}{\partial t} = 0, \tag{6.5}$$

$$\frac{df}{dt} = \frac{\partial f}{\partial q_1}\dot{q}_1 + \frac{\partial f}{\partial q_2}\dot{q}_2 + ... + \frac{\partial f}{\partial q_n}\dot{q}_n + \frac{\partial f}{\partial t} = 0. \tag{6.6}$$

The velocity constraint given by Eq. (6.5) can be represented as

$$a_x\dot{x} + \ a_y\dot{y} + \ a_z\dot{z} + \ a_0 = 0. \tag{6.7}$$

For a dynamical system with n generalized coordinates subjected to m constraints the velocity constraint given by Eq. (6.6) can be expressed as

$$\sum_{k=1}^{n} a_{jk}\dot{q}_k + a_{j0} = 0, \quad j = 1, 2, ..., m, \tag{6.8}$$

where a_{jk} and a_{j0} (j = 1, 2, . . . , m; k = 1, 2, . . . , n) are functions of the generalized coordinates and time.

A *holonomic* constraint is a constraint that can be represented as both a configuration constraint as well as in velocity constraint.

Constraints that do not have this property are called *nonholonomic* (a nonholonomic constraints cannot be expressed as configuration constraints). When the constraint is nonholonomic, it can only be expressed in the form Eqs. (6.7) or (6.8), as an integrating factor does not exist to allow expression in the form of Eqs. (6.1) or (6.2).

A *scleronomic* system, $f(q_1,\ q_1,\ .\ \ .\ \ .,\ q_n) = 0$, is an unconstrained dynamical system or a system subjected to a holonomic constraint that is not an explicit function of time.

A *rhenomic* system is a system subjected to a holonomic constraint that is an explicit function of time.

6.2 Laws of Motion

Consider the motion of a system $\{S\}$ of ν particles $P_1, ..., P_\nu$ ($\{S\} = \{P_1, ..., P_\nu\}$) in an inertial reference frame (0). The equation of motion for the ith particle is

$$\mathbf{F}_i = m_i\,\mathbf{a}_i, \tag{6.9}$$

where $\mathbf{F}_i$ is the resultant of all contact and distance forces acting on P_i; m_i is the mass of P_i; and $\mathbf{a}_i$ is the acceleration of P_i in (0). Equation (6.9) is the expression of Newton's second law.

The inertia force $\mathbf{F}_{in\,i}$ for P_i in (0) is defined as

$$\mathbf{F}_{in\,i} = -\,m_i\,\mathbf{a}_i, \tag{6.10}$$

then Eq. (6.9) is written as

$$\mathbf{F}_i + \mathbf{F}_{in\,i} = \mathbf{0}. \tag{6.11}$$

Equation (6.11) is the expression of D'Alembert's principle.

If $\{S\}$ is a holonomic system possessing n degrees of freedom, then the position vector $\mathbf{r}_i$ of P_i relative to a point O fixed in reference frame (0) is expressed as a vector function of n generalized coordinates $q_1, ..., q_n$ and time t

$$\mathbf{r}_i = \mathbf{r}_i(q_1, ..., q_n, t).$$

The velocity $\mathbf{v}_i$ of P_i in (0) has the form

$$\mathbf{v}_i = \sum_{r=1}^{n} \frac{\partial \mathbf{r}_i}{\partial q_r}\,\frac{\partial q_r}{\partial t} + \frac{\partial \mathbf{r}_i}{\partial t} = \sum_{r=1}^{n} \frac{\partial \mathbf{r}_i}{\partial q_r}\,\dot{q}_r + \frac{\partial \mathbf{r}_i}{\partial t}, \tag{6.12}$$

or as

$$\mathbf{v}_i = \sum_{r=1}^{n} (\mathbf{v}_i)_r\,\dot{q}_r + \frac{\partial \mathbf{r}_i}{\partial t},$$

where $(\mathbf{v}_i)_r$ is called the rth *partial velocity* of P_i in (0) and is defined as

$$(\mathbf{v}_i)_r = \frac{\partial \mathbf{r}_i}{\partial q_r} = \frac{\partial \mathbf{v}_i}{\partial \dot{q}_r}. \tag{6.13}$$

Next, replace Eq. (6.11) with

$$\sum_{i=1}^{\nu} (\mathbf{F}_i + \mathbf{F}_{in\,i}) \cdot (\mathbf{v}_i)_r = 0. \tag{6.14}$$

If a *generalized active force* Q_r and a *generalized inertia force* $K_{in\,r}$ are defined as

$$Q_r = \sum_{i=1}^{\nu} (\mathbf{v}_i)_r \cdot \mathbf{F}_i = \sum_{i=1}^{\nu} \frac{\partial \mathbf{r}_i}{\partial q_r} \cdot \mathbf{F}_i = \sum_{i=1}^{\nu} \frac{\partial \mathbf{v}_i}{\partial \dot{q}_r} \cdot \mathbf{F}_i, \tag{6.15}$$

and

$$K_{in\,r} = \sum_{i=1}^{\nu} (\mathbf{v}_i)_r \cdot \mathbf{F}_{in\,i} = \sum_{i=1}^{\nu} \frac{\partial \mathbf{r}_i}{\partial q_r} \cdot \mathbf{F}_{in\,i} = \sum_{i=1}^{\nu} \frac{\partial \mathbf{v}_i}{\partial \dot{q}_r} \cdot \mathbf{F}_{in\,i}, \tag{6.16}$$

then Eq. (6.14) can be written as

$$Q_r + K_{in\,r} = 0, \quad r = 1, ..., n. \tag{6.17}$$

Equations (6.17) are *Kane's dynamical equations.*
Consider the generalized inertia force $K_{in\,r}$

$$K_{in\,r} = \sum_{i=1}^{\nu} \mathbf{F}_{in\,i} \cdot (\mathbf{v}_i)_r = -\sum_{i=1}^{\nu} m_i\, \mathbf{a}_i \cdot (\mathbf{v}_i)_r = -\sum_{i=1}^{\nu} m_i\, \ddot{\mathbf{r}}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_r} =$$
$$-\sum_{i=1}^{\nu} \left[\frac{d}{d\,t} \left(m_i\, \dot{\mathbf{r}}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_r} \right) - m_i\, \dot{\mathbf{r}}_i \cdot \frac{d}{d\,t} \left(\frac{\partial \mathbf{r}_i}{\partial q_r} \right) \right]. \tag{6.18}$$

Now

$$\frac{d}{d\,t} \left(\frac{\partial \mathbf{r}_i}{\partial q_r} \right) = \sum_{k=1}^{n} \frac{\partial^2 \mathbf{r}_i}{\partial q_r\, \partial q_k} \dot{q}_k + \frac{\partial^2 \mathbf{r}_i}{\partial q_r\, \partial t} = \frac{\partial \mathbf{v}_i}{\partial q_r}, \tag{6.19}$$

and furthermore using Eq. (6.12)

$$\frac{\partial \mathbf{v}_i}{\partial \dot{q}_r} = \frac{\partial \mathbf{r}_i}{\partial q_r}. \tag{6.20}$$

Substitution of Eq. (6.19) and Eq. (6.20) in Eq. (6.18) leads to

$$K_{in\,r} = -\sum_{i=1}^{\nu} \left[\frac{d}{d\,t} \left(m_i\, \mathbf{v}_i \cdot \frac{\partial \mathbf{v}_i}{\partial \dot{q}_r} \right) - m_i\, \mathbf{v}_i \cdot \frac{\partial \mathbf{v}_i}{\partial q_r} \right] =$$
$$-\left[\frac{d}{d\,t} \frac{\partial}{\partial \dot{q}_r} \left(\sum_{i=1}^{\nu} \frac{1}{2} m_i\, \mathbf{v}_i \cdot \mathbf{v}_i \right) - \frac{\partial}{\partial q_r} \left(\sum_{i=1}^{\nu} \frac{1}{2} m_i\, \mathbf{v}_i \cdot \mathbf{v}_i \right) \right].$$

The *kinetic energy* T of $\{S\}$ in reference frame (0) is defined as

$$T = \frac{1}{2} \sum_{i=1}^{\nu} m_i\, \mathbf{v}_i \cdot \mathbf{v}_i.$$

Therefore, the generalized inertia forces $K_{in\,r}$ are written as

$$K_{in\,r} = -\frac{d}{d\,t}\left(\frac{\partial T}{\partial \dot{q}_r}\right) + \frac{\partial T}{\partial q_r},$$

and Kane's dynamical equations can be written as

$$Q_r - \frac{d}{d\,t}\left(\frac{\partial T}{\partial \dot{q}_r}\right) + \frac{\partial T}{\partial q_r} = 0,$$

or

$$\frac{d}{d\,t}\left(\frac{\partial T}{\partial \dot{q}_r}\right) - \frac{\partial T}{\partial q_r} = Q_r.$$

The equations

$$\frac{d}{d\,t}\left(\frac{\partial T}{\partial \dot{q}_r}\right) - \frac{\partial T}{\partial q_r} = \sum_{i=1}^{\nu} \frac{\partial \mathbf{r}_i}{\partial q_r} \cdot \mathbf{F}_i, \quad r = 1, ..., n. \tag{6.21}$$

or

$$\frac{d}{d\,t}\left(\frac{\partial T}{\partial \dot{q}_r}\right) - \frac{\partial T}{\partial q_r} = \sum_{i=1}^{\nu} \frac{\partial \mathbf{v}_i}{\partial \dot{q}_r} \cdot \mathbf{F}_i, \quad r = 1, ..., n. \tag{6.22}$$

are known as *Lagrange's equations of motion* of the first kind.

6.3 Lagrange's Equations for Two-Link Robot Arm

A two-link robot arm is considered in Fig. 5.7. The bars 1 and 2 are homogenuos and have the lengths $L_1 = L_2 = L = 1$ m. The masses of the rigid links are $m_1 = m_2 = m = 1$ kg and the gravitational acceleration is $g = 9.81$ m/s^2. To characterize the instantaneous configuration of the system, two generalized coordinates $q_1(t)$ and $q_2(t)$ are employed. The generalized coordinates q_1 and q_2 denote the radian measure of the angles between the link 1 and 2 and the horizontal x-axis. The set of contact forces transmitted from 0 to 1 is replaced with a couple of torque $\mathbf{T}_{01} = T_{01z}\,\mathbf{k}$ applied to 1 at A, and the set of contact forces transmitted from 1 to 2 is replaced with a couple of torque $\mathbf{T}_{12} = T_{12z}\,\mathbf{k}$ applied to 2 at B.

The initial conditions, at $t = 0$ s, are $q_1(0) = -\pi/18$ rad, $\dot{q}_1(0) = 0$ rad/s, $q_2(0) = \pi/6$ rad, and $\dot{q}_2(0) = 0$ rad/s. The robot arm can be brought from an initial state of rest to a final state of rest in such a way that q_1 and q_2 have the specified values $q_{1f} = \pi/6$ rad and $q_{2f} = \pi/3$ rad.
Find the Lagrange's equations of motion.

I. Direct dynamics

The following feedback control laws are given

$$
\begin{aligned}
T_{01z} &= -\beta_{01}\,\dot{q}_1 - \gamma_{01}\,(q_1 - q_{1f}) + 0.5\,g\,L_1\,m_1\,\cos q_1 + g\,L_1\,m_2\,\cos q_1,\\
T_{12z} &= -\beta_{12}\,\dot{q}_2 - \gamma_{12}\,(q_2 - q_{2f}) + 0.5\,g\,L_2\,m_2\,\cos q_2.
\end{aligned}
$$

The constant gains are: $\beta_{01} = 450$ N·m·s/rad, $\gamma_{01} = 300$ N·m/rad, $\beta_{12} = 200$ N·m·s/rad, and $\gamma_{12} = 300$ N·m/rad.
Write a MATLAB program for solving the equations of motion.

II. Inverse dynamics

A desired motion of the robot arm is specified for a time interval $0 \le t \le T_p = 15$ s. The generalized coordinates can be established explicitly

$$
q_r(t) = q_r(0) + \frac{q_r(T_p) - q_r(0)}{T_p}\left[t - \frac{T_p}{2\,\pi}\sin\left(\frac{2\,\pi\,t}{T_p}\right)\right], \quad r = 1, 2.
$$

with $q_r(T_p) = q_{rf}$.
Find $T_{01z}(t)$ and $T_{12z}(t)$ for $0 \le t \le T_p = 15$ s.

I. Direct dynamics
Kinematics

The position vector of the mass center of link 1 is

$$\mathbf{r}_{C_1} = 0.5\,L\,\cos q_1\,\boldsymbol{\imath} + 0.5\,L\,\sin q_1\,\boldsymbol{\jmath},$$

and the position vector of the mass center of link 2 is

$$\mathbf{r}_{C_2} = (L\,\cos q_1 + 0.5\,L\,\cos q_2)\,\boldsymbol{\imath} + (L\,\sin q_1 + 0.5\,L\,\sin q_2)\,\boldsymbol{\imath}.$$

The velocity of C_1 is

$$\mathbf{v}_{C_1} = \frac{d\,\mathbf{r}_{C_1}}{d\,t} = \dot{\mathbf{r}}_{C_1} = -0.5\,L\,\dot{q}_1\,\sin q_1\,\boldsymbol{\imath} + 0.5\,L\,\dot{q}_1\,\cos q_1\,\boldsymbol{\jmath},$$

and the velocity of C_2 is

$$\mathbf{v}_{C_2} = \frac{d\,\mathbf{r}_{C_2}}{d\,t} = \dot{\mathbf{r}}_{C_2} =$$
$$(-L\,\dot{q}_1\,\sin q_1 - 0.5\,L\,\dot{q}_2\,\sin q_2)\,\boldsymbol{\imath} + (L\,\dot{q}_1\,\cos q_1 + 0.5\,L\,\dot{q}_2\,\cos q_2)\,\boldsymbol{\jmath}.$$

The angular velocity vectors of the links 1 and 2 are

$$\boldsymbol{\omega}_1 = \dot{q}_1\mathbf{k} \quad \text{and} \quad \boldsymbol{\omega}_2 = \dot{q}_2\mathbf{k}.$$

The MATLAB commands for the kinematics are

```
syms t L1 L2 m1 m2 m3 g real
q1 = sym('q1(t)');
q2 = sym('q2(t)');
c1 = cos(q1); s1 = sin(q1);
c2 = cos(q2); s2 = sin(q2);
xB = L1*c1; yB = L1*s1; rB = [xB yB 0];
rC1 = rB/2; vc1 = diff(rC1,t);
xD = xB + L2*c2; yD = yB + L2*s2; rD = [xD yD 0];
rC2 = (rB + rD)/2; vc2 = diff(rC2,t);
omega1 = [0 0 diff(q1,t)];
omega2 = [0 0 diff(q2,t)];
```

Kinetic energy

The kinetic energy of the link 1 which is in rotational motion is

$$T_1 = \frac{1}{2}\, I_A\, \boldsymbol{\omega}_1 \cdot \boldsymbol{\omega}_1 = \frac{1}{2}\, I_A\, \dot{q}_1^2 = \frac{1}{2}\, \frac{mL^2}{3}\, \dot{q}_1^2 = \frac{mL^2}{6}\, \dot{q}_1^2,$$

where I_A is the mass moment of inertia about the center of rotation A, $I_A = mL^2/3$.

The kinetic energy of the bar 2 is due to the translation and rotation and can be expressed as

$$T_2 = \frac{1}{2} I_{C_2}\, \boldsymbol{\omega}_1 \cdot \boldsymbol{\omega}_1 + \frac{1}{2} m_2\, \mathbf{v}_{C_2} \cdot \mathbf{v}_{C_2} = \frac{1}{2} I_{C_2} \dot{q}_2^2 + \frac{1}{2} m_2\, \mathbf{v}_{C_2} \cdot \mathbf{v}_{C_2},$$

where I_{C_2} is the mass moment of inertia about the center of mass C_2, $I_{C_2} = mL^2/12$, and

$$\mathbf{v}_{C_2} \cdot \mathbf{v}_{C_2} = \mathbf{v}_{C_2}^2 = L^2\, \dot{q}_1^2 + \frac{1}{4}\, L^2\, \dot{q}_2^2 + L^2\, \dot{q}_1\, \dot{q}_2 \cos(q_2 - q_1).$$

Equation (6.23) becomes

$$T_2 = \frac{1}{2}\, \frac{mL^2}{12}\, \dot{q}_2^2 + \frac{1}{2} mL^2 \left[\dot{q}_1^2 + \frac{1}{4}\, \dot{q}_2^2 + \dot{q}_1\, \dot{q}_2 \cos(q_2 - q_1)\right].$$

The total kinetic energy of the system is

$$T = T_1 + T_2 = \frac{mL^2}{6}\, \left[4\dot{q}_1^2 + 3\, \dot{q}_1\, \dot{q}_2\, \cos(q_2 - q_1) + \dot{q}_2^2\right].$$

The MATLAB commands for the kinetic energy are

```
IA = Im1*L1^2/3; IC2 = m2*L2^2/12;
T1 = IA*omega1*omega1.'/2       % .'  array transpose
T2 = m2*vC2*vC2.'/2 + IC2*omega2*omega2.'/2;
T2 = simple(T2)
T = expand(T1 + T2);      % total kinetic energy
```

The MATLAB statement `A.'` is the array transpose of `A`.

The left hand sides of Lagrange's equations $\partial T/\partial \dot{q}_i,\ i = 1, 2$ are

$$\frac{\partial T}{\partial \dot{q}_1} = \frac{mL^2}{6}\, \left[8\dot{q}_1 + 3\dot{q}_2\, \cos(q_2 - q_1)\right],$$

$$\frac{\partial T}{\partial \dot{q}_2} = \frac{mL^2}{6}\, \left[3\dot{q}_1\, \cos(q_2 - q_1) + 2\dot{q}_2\right].$$

To calculate the partial derivative of the kinetic energy `T` with respect to `diff('q1(t)',t)` a MATLAB function, `deriv`, is created

```
function fout = deriv(f, g)
% deriv differentiates f with respect to g=g(t)
% the variable g=g(t) is a function of time
x = sym('x');
f1 = subs(f, g, x);
f2 = diff(f1, x);
fout = subs(f2, x, g);
```

The function `deriv(f, g)` differentiates a symbolic expression `f` with respect to the variable `g`, where the variable `g` is a function of time `g = g(t)`. The statement `diff(f,'x')` differentiates `f` with respect to the free variable `x`. In MATLAB the free variable `x` cannot be a function of time and that is why the function `deriv` was introduced.

The partial derivatives of the kinetic energy T with respect to $\dot{q}_i$ or in MATLAB the partial derivatives of `T` with respect to `diff('q1(t)',t)` and `diff('q2(t)',t)` are calculated with

```
Tdq1 = deriv(T, 'dq_1');
Tdq2 = deriv(T, 'dq_2');
```

Next the derivative of $\partial T/\partial \dot{q}_i$ with respect to time is calculated

$$\frac{d}{d\,t}\left(\frac{\partial T}{\partial \dot{q}_1}\right) = \frac{mL^2}{6}\left[8\ddot{q}_1 + 3\ddot{q}_2\,\cos(q_2-q_1) - 3\dot{q}_2\,(\dot{q}_2-\dot{q}_1)\,\sin(q_2-q_1)\right],$$
$$\frac{d}{d\,t}\left(\frac{\partial T}{\partial \dot{q}_2}\right) = \frac{mL^2}{6}\left[3\ddot{q}_1\,\cos(q_2-q_1) - 3\dot{q}_1\,(\dot{q}_2-\dot{q}_1)\,\sin(q_2-q_1) + 2\ddot{q}_2\right],$$

and in MATLAB the terms $\dfrac{d}{d\,t}\left(\dfrac{\partial T}{\partial \dot{q}_i}\right)$ are

```
Tt1 = diff(Tdq1, t);
Tt2 = diff(Tdq2, t);
```

The partial derivative of the kinetic energy with respect to q_i are

$$\frac{\partial T}{\partial q_1} = \frac{mL^2}{6}\,3\,\dot{q}_1\,\dot{q}_2\,\sin(q_2-q_1) = \frac{mL^2}{2}\,\dot{q}_1\,\dot{q}_2\,\sin(q_2-q_1);$$

$$\frac{\partial T}{\partial q_2} = -\frac{mL^2}{6}\, 3\,\dot{q}_1\,\dot{q}_2\,\sin(q_2 - q_1) = -\frac{mL^2}{2}\,\dot{q}_1\,\dot{q}_2\,\sin(q_2 - q_1),$$

and with MATLAB

```
Tq1 = deriv(T, q1);
Tq2 = deriv(T, q2);
```

The left hand side of Lagrange's equations, $\dfrac{d}{dt}\left(\dfrac{\partial T}{\partial \dot{q}_i}\right) - \dfrac{\partial T}{\partial q_i}$, with MATLAB are

```
LHS1 = Tt1 - Tq1;
LHS2 = Tt2 - Tq2;
```

Generalized active forces

The gravity forces on links 1 and 2 at the mass centers C_1 and C_2

$$\mathbf{G}_1 = -m_1\, g\, \mathbf{j} = -m\, g\, \mathbf{j} \quad \text{and} \quad \mathbf{G}_2 = -m_2\, g\, \mathbf{j} = -m\, g\, \mathbf{j}.$$

The torque transmitted from 0 to 1 at A is $\mathbf{T}_{01} = T_{01z}\,\mathbf{k}$ and the torque transmitted from 1 to 2 at B is $\mathbf{T}_{12} = T_{12z}\,\mathbf{k}$. The MATLAB commands for the net forces and moments are

```
G1 = [0 -m1*g 0]; G2 = [0 -m2*g 0];
syms T01z T12z real
T01 = [0 0 T01z];
T12 = [0 0 T12z];
```

There are two generalized forces. The generalized force associated to q_1 is

$$Q_1 = \mathbf{G}_1 \cdot \frac{\partial \mathbf{r}_{C_1}}{\partial q_1} + \mathbf{T}_{01} \cdot \frac{\partial \boldsymbol{\omega}_1}{\partial \dot{q}_1} - \mathbf{T}_{12} \cdot \frac{\partial \boldsymbol{\omega}_1}{\partial \dot{q}_1} + \mathbf{G}_2 \cdot \frac{\partial \mathbf{r}_{C_2}}{\partial q_1} + \mathbf{T}_{12} \cdot \frac{\partial \boldsymbol{\omega}_2}{\partial \dot{q}_1} =$$
$$-m\, g\, \mathbf{j} \cdot (-0.5\, L\, \sin q_1\, \mathbf{i} + 0.5\, L\, \cos q_1\, \mathbf{j}) + T_{01z} - T_{12z} - m\, g\, \mathbf{j} \cdot (-L\, \sin q_1\, \mathbf{i} + L\, \cos q_1\, \mathbf{j})$$
$$= -1.5\, m\, g\, L\, \cos q_1 + T_{01z} - T_{12z}.$$

The generalized force associated to q_2 is

$$Q_2 = \mathbf{G}_1 \cdot \frac{\partial \mathbf{r}_{C_1}}{\partial q_2} + \mathbf{T}_{01} \cdot \frac{\partial \boldsymbol{\omega}_1}{\partial \dot{q}_2} - \mathbf{T}_{12} \cdot \frac{\partial \boldsymbol{\omega}_1}{\partial \dot{q}_2} + \mathbf{G}_2 \cdot \frac{\partial \mathbf{r}_{C_2}}{\partial q_2} + \mathbf{T}_{12} \cdot \frac{\partial \boldsymbol{\omega}_2}{\partial \dot{q}_2} =$$
$$-mg\, \mathbf{j} \cdot (-0.5\, L\, \sin q_2\, \mathbf{i} + 0.5\, L\, \cos q_2\, \mathbf{j}) + T_{12z}$$
$$= -0.5\, m\, g\, L\, \cos q_2 + T_{12z}.$$

The MATLAB commands for the partial derivatives of the position vectors of the mass centers,

$$\frac{\partial \mathbf{r}_{C_1}}{\partial q_1},\ \frac{\partial \mathbf{r}_{C_2}}{\partial q_1},\ \frac{\partial \mathbf{r}_{C_1}}{\partial q_2},\ \frac{\partial \mathbf{r}_{C_2}}{\partial q_2}$$

are

```
rC1_1 = deriv(rC1, q1); rC2_1 = deriv(rC2, q1);
rC1_2 = deriv(rC1, q2); rC2_2 = deriv(rC2, q2);
```

The MATLAB commands for the partial angular velocities,

$$\frac{\partial \boldsymbol{\omega}_1}{\partial \dot{q}_1},\ \frac{\partial \boldsymbol{\omega}_2}{\partial \dot{q}_1},\ \frac{\partial \boldsymbol{\omega}_1}{\partial \dot{q}_2},\ \frac{\partial \boldsymbol{\omega}_2}{\partial \dot{q}_2}$$

are

```
w1_1 = deriv(omega1, diff(q1,t)); w2_1 = deriv(omega2, diff(q1,t));
w1_2 = deriv(omega1, diff(q2,t)); w2_2 = deriv(omega2, diff(q2,t));
```

The generalized active forces are calculated with the MATLAB commands

```
Q1 = rC1_1*G1.'+w1_1*T01.'+w1_1*(-T12.')+rC2_1*G2.'+w2_1*T12.'
Q2 = rC1_2*G1.'+w1_2*T01.'+w1_2*(-T12.')+rC2_2*G2.'+w2_2*T12.'
```

The two Lagrange's equations are

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_1}\right) - \frac{\partial T}{\partial q_1} = Q_1,$$
$$1.333\, m\, L^2\, \ddot{q}_1 + 0.5\, m\, L^2\, \ddot{q}_2\, \cos(q_2 - q_1) - 0.5 m\, L^2 \dot{q}_2^2\, \sin(q_2 - q_1)$$
$$+1.5\, m\, g\, L\, \cos q_1 - T_{01z} + T_{12z} = 0;$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_2}\right) - \frac{\partial T}{\partial q_2} = Q_2,$$
$$0.5\, m\, L^2\, \ddot{q}_1 \cos(q_2 - q_1) + 0.333\, m\, L^2\, \ddot{q}_2 + 0.5\, m\, L^2\, \dot{q}_1^2\, \sin(q_2 - q_1)$$
$$+0.5\, m\, g\, L\, \cos q_2 - T_{12z} = 0, \tag{6.23}$$

or in MATLAB

```
Lagrange1 = LHS1-Q1;
Lagrange2 = LHS2-Q2;
```

The feedback control laws are

$$
\begin{aligned}
T_{01z} &= -\beta_{01}\,\dot{q}_1 - \gamma_{01}\,(q_1 - q_{1f}) + 0.5\,g\,L_1\,m_1\,\cos q_1 + g\,L_1\,m_2\,\cos q_1,\\
T_{12z} &= -\beta_{12}\,\dot{q}_2 - \gamma_{12}\,(q_2 - q_{2f}) + 0.5\,g\,L_2\,m_2\,\cos q_2,
\end{aligned}
$$

with $\beta_{01} = 450$ N·m·s/rad, $\gamma_{01} = 300$ N·m/rad, $\beta_{12} = 200$ N·m·s/rad, and $\gamma_{12} = 300$ N·m/rad.

The feedback control torques using MATLAB commands are

```
b01 = 450; g01 = 300;
b12 = 200; g12 = 300;
q1f = pi/6;
q2f = pi/3;
T01zco = -b01*diff(q1,t)-g01*(q1-q1f)+0.5*g*L1*m1*c1+g*L1*m2*c1;
T01zc = subs(T01zco, tlist, nlist);
T12zco = -b12*diff(q2,t)-g12*(q2-q2f)+0.5*g*L2*m2*c2;
T12zc = subs(T12zco, tlist, nlist);
tor = {T01z, T12z};
torf = {T01zc, T12zc};
```

The feedback control torques are introduced into Lagrange's equations

```
Lagrang1 = subs(Lagrange1, tor, torf);
Lagrang2 = subs(Lagrange2, tor, torf);
```

The numerical data for L_1, L_2, m_1, m_2, and g are introduced in MATLAB with the lists

```
data = {L1, L2, m1, m2, g };
datn = {1 , 1 , 1 , 1 , 9.81};
```

and are substituted into Lagrange's equations

```
Lagran1 = subs(Lagrang1, data, datn);
Lagran2 = subs(Lagrang2, data, datn);
```

The two second order Lagrange's equations have to be rewritten your as a first order system and two MATLAB lists are created

```
ql = {diff(q1,t,2), diff(q2,t,2), diff(q1,t), diff(q2,t), q1, q2};
qf = {'ddq1', 'ddq2', 'x(2)', 'x(4)', 'x(1)', 'x(3)'};

% ql                   qf
% ----------------------------
% diff('q1(t)',t,2) -> 'ddq1'
% diff('q2(t)',t,2) -> 'ddq2'
%   diff('q1(t)',t) -> 'x(2)'
%   diff('q2(t)',t) -> 'x(4)'
%           'q1(t)' -> 'x(1)'
%           'q2(t)' -> 'x(3)'
```

In the expression of the Lagrange's equations
`diff('q1(t)',t,2)` is replaced by `'ddq1'`,
`diff('q2(t)',t,2)` is replaced by `'ddq2'`,
`diff('q1(t)',t)` is replaced by `'x(2)'`,
`diff('q2(t)',t)` is replaced by `'x(4)'`,
`'q1(t)'` is replaced by `'x(1)'`, and
`'q2(t)'` is replaced by `'x(3)'`

or

```
Lagra1 = subs(Lagran1, ql, qf);
Lagra2 = subs(Lagran2, ql, qf);
```

The Lagrange's equations are solved in terms of `'ddq1'` and `'ddq2'`

```
sol = solve(Lagra1,Lagra2,'ddq1, ddq2');
Lagr1 = sol.ddq1;
Lagr2 = sol.ddq2;
```

The system of differential equations is solved numerically by m-file functions. The function file, `RR_Lagr.m` is created using the statements

```
dx2dt = char(Lagr1);
```

```
dx4dt = char(Lagr2);

fid = fopen('RR_Lagr.m','w+');
fprintf(fid,'function dx = RR_Lagr(t,x)\n');
fprintf(fid,'dx = zeros(4,1);\n');
fprintf(fid,'dx(1) = x(2);\n');
fprintf(fid,'dx(2) = ');
fprintf(fid,dx2dt);
fprintf(fid,';\n');
fprintf(fid,'dx(3) = x(4);\n');
fprintf(fid,'dx(4) = ');
fprintf(fid,dx4dt);
fprintf(fid,';');
fclose(fid);
```

The `ode45` solver is used for the system of differential equations

```
t0 = 0; tf = 15; time = [0 tf];
x0 = [-pi/18 0 pi/6 0];
[t,xs] = ode45(@RR_Lagr, time, x0);
x1 = xs(:,1);
x2 = xs(:,2);
x3 = xs(:,3);
x4 = xs(:,4);
subplot(2,1,1),plot(t,x1*180/pi,'r'),...
xlabel('t (s)'),ylabel('q1 (deg)'),grid,...
subplot(2,1,2),plot(t,x3*180/pi,'b'),...
xlabel('t (s)'),ylabel('q2 (deg)'),grid
[ts,xs] = ode45(@RR_Lagr,0:1:5,x0);
fprintf('Results \n'); fprintf('\n');
fprintf(' t(s) q1(rad) dq1(rad/s) q2(rad) dq2(rad/s) \n');
[ts,xs]
```

A MATLAB computer program for the direct dynamics is given in the Program 6.1.

II. Inverse dynamics

The generalized coordinates are given explicitly for $0 \le t \le T_p = 15$ s

$$q_r(t) = q_r(0) + \frac{q_{rf}(T_p) - q_r(0)}{T_p}\left[t - \frac{T_p}{2\,\pi}\sin\left(\frac{2\,\pi\,t}{T_p}\right).\right], \quad r = 1, 2. \quad (6.24)$$

The initial conditions, at $t = 0$ s, are $q_1(0) = -\pi/18$ rad and $q_2(0) = \pi/6$ rad. The robot arm is brought from an initial state of rest to a final state of rest in such a way that q_1 and q_2 have the specified values $q_{1f}(T_p) = \pi/6$ rad and $q_{2f}(T_p) = \pi/3$ rad.

Figure 6.1 shows the plots of $q_1(t)$ and $q_2(t)$ rad.

The MATLAB commands for finding the Lagrage's equations are identical with the commands presented at **Direct dynamics**

```
........................

syms T01z T12z real
T01 = [0 0 T01z]; T12 = [0 0 T12z];
Q1 = rC1_1*G1.'+w1_1*T01.'+w1_1*(-T12.')+rC2_1*G2.'+w2_1*T12.';
Q2 = rC1_2*G1.'+w1_2*T01.'+w1_2*(-T12.')+rC2_2*G2.'+w2_2*T12.';
Lagrange1 = LHS1-Q1; Lagrange2 = LHS2-Q2;
data = {L1, L2, m1, m2, g};
datn = {1 , 1 , 1 , 1 , 9.81};
Lagr1 = subs(Lagrange1, data, datn);
Lagr2 = subs(Lagrange2, data, datn);
```

From the Lagrange's equations of motions the torques T_{01z} and T_{12z} are calculated

```
sol = solve(Lagr1,Lagr2,'T01z, T12z');
T01zc = sol.T01z;
T12zc = sol.T12z;
```

The generalized coordinates, q_1 and q_2, given by Eq. (6.24) and their derivatives, $\dot{q}_1$, $\dot{q}_2$, $\ddot{q}_1$, $\ddot{q}_2$, are substituted in the expressions of T_{01z} and T_{12z}

```
q1f = pi/6 ; q2f = pi/3;
q1s = pi/18; q2s = pi/6;
```

```
Tp=15.;
q1n = q1s + (q1f-q1s)/Tp*(t - Tp/(2*pi)*sin(2*pi/Tp*t));
q2n = q2s + (q2f-q2s)/Tp*(t - Tp/(2*pi)*sin(2*pi/Tp*t));
dq1n = diff(q1n,t);
dq2n = diff(q2n,t);
ddq1n = diff(dq1n,t);
ddq2n = diff(dq2n,t);

ql = {diff(q1,t,2), diff(q2,t,2), diff(q1,t), diff(q2,t), q1, q2};
qn = {ddq1n, ddq2n, dq1n, dq2n, q1n, q2n};

% ql                    qn
% ----------------------------
% diff('q1(t)',t,2) -> ddq1n
% diff('q2(t)',t,2) -> ddq2n
%   diff('q1(t)',t) -> dq1n
%   diff('q2(t)',t) -> dq2n
%           'q1(t)' -> q1n
%           'q2(t)' -> q1n

T01zt = subs(T01zc, ql, qn);
T12zt = subs(T12zc, ql, qn);
```

The plots of $T_{01z}(t)$ and $T_{12z}(t)$ are obtained with the MATLAB commands

```
time = 0:1:Tp;
T01t = subs(T01zt,'t',time);
T12t = subs(T12zt,'t',time);
subplot(2,1,1),plot(time,T01t),...
xlabel('t (s)'),ylabel('T01z (N m)'),grid,...
subplot(2,1,2),plot(time,T12t),...
xlabel('t (s)'),ylabel('T12z (N m)'),grid
```

Figure 6.2 shows the control torques and the MATLAB program is given in Program 6.2.

6.4 Rotation Transformation

Two orthogonal reference frames, $Oxyz$ and $O'x'y'z'$, are considered. The unit vectors of the reference frame $Oxyz$ are $\mathbf{ı}$, $\mathbf{ȷ}$, $\mathbf{k}$ and the unit vectors of the reference frame $O'x'y'z'$ are $\mathbf{ı}'$, $\mathbf{ȷ}'$, $\mathbf{k}'$. The origins of the reference frames may coincide because only the orientation of the axes is of interest $O = O'$.

The angles between the x'-axis and each of the xyz axes are the direction angles α, β, and γ ($O < \alpha, \beta, \gamma < \pi$) as shown in Fig. 6.3. The unit vector $\mathbf{ı}'$ can be expressed in terms of $\mathbf{ı}$, $\mathbf{ȷ}$, $\mathbf{k}$ and the direction angles

$$\mathbf{ı}' = (\mathbf{ı}' \cdot \mathbf{ı})\mathbf{ı} + (\mathbf{ı}' \cdot \mathbf{ȷ})\mathbf{ȷ} + (\mathbf{ı}' \cdot \mathbf{k})\mathbf{k} = \cos\alpha\, \mathbf{ı} + \cos\beta\, \mathbf{ȷ} + \cos\gamma\, \mathbf{k}.$$

The cosines of the direction angles are the direction cosines and
$\cos^2\alpha + \cos^2\beta + \cos^2\gamma = 1$.
With the notations $\cos\alpha = a_{x'x}$, $\cos\beta = a_{x'y}$, and $\cos\gamma = a_{x'z}$ the unit vector $\mathbf{ı}'$ is

$$\mathbf{ı}' = a_{x'x}\, \mathbf{ı} + a_{x'y}\, \mathbf{ȷ} + a_{x'z}\, \mathbf{k}.$$

In a similar way the unit vectors $\mathbf{ȷ}'$ and $\mathbf{k}'$ are

$$\mathbf{ȷ}' = a_{y'x}\, \mathbf{ı} + a_{y'y}\, \mathbf{ȷ} + a_{y'z}\, \mathbf{k},$$
$$\mathbf{ı}' = a_{z'x}\, \mathbf{ı} + a_{z'y}\, \mathbf{ȷ} + a_{z'z}\, \mathbf{k},$$

where $a_{r's} = a_{rs'}$ are the cosine of the angle between axis r' and axis s, with r and r representing x, y, or z. In matrix form

$$\begin{bmatrix} \mathbf{ı}' \\ \mathbf{ȷ}' \\ \mathbf{k}' \end{bmatrix} = \mathbf{R} \begin{bmatrix} \mathbf{ı} \\ \mathbf{ȷ} \\ \mathbf{k} \end{bmatrix},$$

where

$$\mathbf{R} = \begin{bmatrix} a_{x'x} & a_{x'y} & a_{x'z} \\ a_{y'x} & a_{y'y} & a_{y'z} \\ a_{z'x} & a_{z'y} & a_{z'z} \end{bmatrix}.$$

The matrix $\mathbf{R}$ is the *rotation transformation matrix* from xyz to $x'y'z'$.

The unit vectors $\mathbf{ı}, \mathbf{ȷ}, \mathbf{k}$ are an orthogonal set of unit vectors and the unit vectors $\mathbf{ı}', \mathbf{ȷ}', \mathbf{k}'$ are an orthogonal too. Using these properties it results

$$\mathbf{R} \cdot \mathbf{R}^T = \mathbf{I},$$

where $\mathbf{I}$ is the identity matrix. Multiplication of Eq. (6.25) by $\mathbf{R}^{-1}$ gives

$$\mathbf{R}^{-1} = \mathbf{R}^T.$$

The matrix R is an orthonormal matrix because $\mathbf{R}^{-1} = \mathbf{R}^T$.
Let $\mathbf{R}'$ be the transformation matrix from $\boldsymbol{\imath}, \boldsymbol{\jmath}, \mathbf{k}$ to $\boldsymbol{\imath}', \boldsymbol{\jmath}', \mathbf{k}'$

$$\begin{bmatrix} \boldsymbol{\imath} \\ \boldsymbol{\jmath} \\ \mathbf{k} \end{bmatrix} = \mathbf{R}' \begin{bmatrix} \boldsymbol{\imath}' \\ \boldsymbol{\jmath}' \\ \mathbf{k}' \end{bmatrix}. \tag{6.25}$$

The matrix $\mathbf{R}'$ is the inverse of the original transformation matrix $\mathbf{R}$, which is identical to the transpose of $\mathbf{R}$.

$$\mathbf{R}' = \mathbf{R}^{-1} = \mathbf{R}^T.$$

Any vector $\mathbf{p}$ is independent of the reference frame used to describe its components, so

$$\mathbf{p} = p_x\ \boldsymbol{\imath} + p_y\ \boldsymbol{\jmath} + p_z\ \mathbf{k} = p_{x'}\ \boldsymbol{\imath}' + p_{y'}\ \boldsymbol{\jmath}' + p_{z'}\ \mathbf{k}',$$

or in matrix form as

$$\begin{bmatrix} \boldsymbol{\imath}' & \boldsymbol{\jmath}' & \mathbf{k}' \end{bmatrix} \begin{bmatrix} p_{x'} \\ p_{y'} \\ p_{z'} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\imath} & \boldsymbol{\jmath} & \mathbf{k} \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}.$$

Using Eq. (6.25) and the fact that the transpose of a product is the product of the transposes the following relation is obtained

$$\begin{bmatrix} \boldsymbol{\imath}' & \boldsymbol{\jmath}' & \mathbf{k}' \end{bmatrix} \begin{bmatrix} p_{x'} \\ p_{y'} \\ p_{z'} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\imath}' & \boldsymbol{\jmath}' & \mathbf{k}' \end{bmatrix} [\mathbf{R}']^T \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}.$$

With $[\mathbf{R}']^T = \mathbf{R}$ the above equation leads to

$$\begin{bmatrix} p_{x'} \\ p_{y'} \\ p_{z'} \end{bmatrix} = \mathbf{R} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}.$$

When the reference frame $x'y'z'$ is the result of a simple rotation about one of the axes of the reference frame xyz the following transformation matrices

are obtained (Fig. 6.4):
- the reference frame xyz is rotated by an angle θ_x about x-axis

$$\mathbf{R}(x,\theta_x) = \mathbf{R}(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & \sin\theta_x \\ 0 & -\sin\theta_x & \cos\theta_x \end{bmatrix},$$

- the reference frame xyz is rotated by an angle θ_y about y-axis

$$\mathbf{R}(y,\theta_y) = \mathbf{R}(\theta_y) = \begin{bmatrix} \cos\theta_y & 0 & -\sin\theta_y \\ 0 & 1 & 0 \\ \sin\theta_y & 0 & \cos\theta_y \end{bmatrix},$$

- the reference frame xyz is rotated by an angle θ_y about z-axis

$$\mathbf{R}(z,\theta_z) = \mathbf{R}(\theta_z) = \begin{bmatrix} \cos\theta_z & \sin\theta_z & 0 \\ -\sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The following property holds

$$\mathbf{R}(s,-\theta_s) = \mathbf{R}^T(s,\theta_s), \quad s = x,\, y,\, z.$$

6.5 RRT Robot Arm

Figure 6.5(a) is a schematic representation of a RRT robot arm consisting of three links 1, 2, and 3. Let m_1, m_2, m_3 be the masses of 1, 2, 3, respectively. Link 1 can be rotated at A in a "fixed" reference frame (0) of unit vectors $[\mathbf{\imath}_0, \mathbf{\jmath}_0, \mathbf{k}_0]$ about a vertical axis $\mathbf{\imath}_0$. The unit vector $\mathbf{\imath}_0$ is fixed in 1. The link 1 is connected to link 2 at the pin joint B. The element 2 rotates relative to 1 about an axis fixed in both 1 and 2, passing through B, and perpendicular to the axis of 1. The last link 3 is connected to 2 by means of a slider joint. The mass centers of links 1, 2, and 3 are C_1, C_2, and C_3, respectively. The distances $L_1 = AC_1$, $L_B = AB = 2\,L_1$, and $L_2 = BC_2$ are indicated in Fig. 6.5(a) The length of link 1 is $2\,L_1$ and the length of link 2 is $2\,L_2$. The reference frame (1) of the unit vectors $[\mathbf{\imath}_1, \mathbf{\jmath}_1, \mathbf{k}_1]$ is attached to link 1, and the reference frame (2) of the unit vectors $[\mathbf{\imath}_2, \mathbf{\jmath}_2, \mathbf{k}_2]$ is attached to link 2, as shown in Fig. 6.5(b).

The generalized coordinates (quantities associated with the the instantaneous position of the system) are $q_1(t)$, $q_2(t)$, $q_3(t)$.

The first generalized coordinate q_1 denotes the radian measure of the angle between the axes of (1) and (0). The unit vectors $\mathbf{\imath}_1$, $\mathbf{\jmath}_1$, and $\mathbf{k}_1$ can be expressed as functions of $\mathbf{\imath}_0$, $\mathbf{\jmath}_0$, and $\mathbf{k}_0$

$$\begin{aligned}
\mathbf{\imath}_1 &= \mathbf{\imath}_0,\\
\mathbf{\jmath}_1 &= c_1\,\mathbf{\jmath}_0 + s_1\,\mathbf{k}_0,\\
\mathbf{k}_1 &= -s_1\,\mathbf{\jmath}_0 + c_1\,\mathbf{k}_0,
\end{aligned} \tag{6.26}$$

or

$$\begin{bmatrix} \mathbf{\imath}_1 \\ \mathbf{\jmath}_1 \\ \mathbf{k}_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_1 & s_1 \\ 0 & -s_1 & c_1 \end{bmatrix} \begin{bmatrix} \mathbf{\imath}_0 \\ \mathbf{\jmath}_0 \\ \mathbf{k}_0 \end{bmatrix},$$

where $s_1 = \sin q_1$ and $c_1 = \cos q_1$. The transformation matrix from (1) to (0) is

$$R_{10} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_1 & s_1 \\ 0 & -s_1 & c_1 \end{bmatrix}. \tag{6.27}$$

The second generalized coordinate designates also a radian measure of the rotation angle between (1) and (2). The unit vectors $\mathbf{\imath}_2$, $\mathbf{\jmath}_2$ and $\mathbf{k}_2$ can be expressed as

$$
\begin{aligned}
\mathbf{\imath}_2 &= c_2\mathbf{\imath}_1 - s_2\mathbf{k}_1\\
&= c_2\mathbf{\imath}_0 + s_1 s_2\,\mathbf{\jmath}_0 - c_1 s_2\,\mathbf{k}_0,\\
\mathbf{\jmath}_2 &= \mathbf{\jmath}_1,\\
&= c_1\,\mathbf{\jmath}_0 + s_1\,\mathbf{k}_0,\\
\mathbf{k}_2 &= s_2\mathbf{\imath}_1 + c_2\mathbf{k}_1\\
&= s_2\mathbf{\imath}_0 - c_2 s_1\,\mathbf{\jmath}_0 + c_1 c_2\,\mathbf{k}_0,
\end{aligned}
\tag{6.28}
$$

where $s_2 = \sin q_2$ and $c_2 = \cos q_2$. The transformation matrix from (2) to (1) is

$$
R_{21} = \begin{bmatrix} c_2 & 0 & -s_2 \\ 0 & 1 & 0 \\ s_2 & 0 & c_2 \end{bmatrix}. \tag{6.29}
$$

The last generalized coordinate q_3 is the distance from C_2 to C_3.

The MATLAB commands for the transformation matrices are

```
q1 = sym('q1(t)');
q2 = sym('q2(t)');
q3 = sym('q3(t)');
c1 = cos(q1);
s1 = sin(q1);
c2 = cos(q2);
s2 = sin(q2);
% transformation matrix from RF1 to RF0
R10 = [[1 0 0]; [0 c1 s1]; [0 -s1 c1]];
% transformation matrix from RF2 to RF1
R21 = [[c2 0 -s2]; [0 1 0]; [s2 0 c2]];
```

Angular velocities

Next the angular velocity of the links 1, 2, and 3 will be expressed in the fixed reference frame (0). The angular velocity of 1 in (0) is

$$
\boldsymbol{\omega}_{10} = \dot{q}_1\,\mathbf{\imath}_1. \tag{6.30}
$$

The angular velocity of the link 2 with respect to (1) is

$$
\boldsymbol{\omega}_{21} = \dot{q}_2\,\mathbf{\jmath}_2.
$$

The angular velocity of the link 2 with respect to the fixed reference frame (0) is

$$\boldsymbol{\omega}_{20} = \boldsymbol{\omega}_{10} + \boldsymbol{\omega}_{21} = \dot{q}_1\, \mathbf{\imath}_1 + \dot{q}_2 \mathbf{\jmath}_2.$$

With $\mathbf{\imath}_0 = \mathbf{\imath}_1 = c_2\, \mathbf{\imath}_2 + s_2\, \mathbf{k}_2$ the angular velocity of the link 2 in the reference frame (0) written in terms of the reference frame (2) is

$$\boldsymbol{\omega}_{20} = \dot{q}_1 (c_2\, \mathbf{\imath}_2 + s_2\, \mathbf{k}_2) + \dot{q}_2\, \mathbf{\jmath}_2 = \dot{q}_1\, c_2\, \mathbf{\imath}_2 + \dot{q}_2\, \mathbf{\jmath}_2 + \dot{q}_1\, s_2\, \mathbf{k}_2. \tag{6.31}$$

The link 3 has the same rotational motion as link 2, i.e., $\boldsymbol{\omega}_{30} = \boldsymbol{\omega}_{20}$.

Angular accelerations

The angular acceleration of the link 1 in the reference frame (0) is

$$\boldsymbol{\alpha}_{10} = \ddot{q}_1\, \mathbf{\imath}_1. \tag{6.32}$$

The angular acceleration of the link 2 with respect to the reference frame (0) is

$$\boldsymbol{\alpha}_{20} = \frac{d}{dt}\boldsymbol{\omega}_{20} = \frac{^{(2)}d}{dt}\boldsymbol{\omega}_{20} + \boldsymbol{\omega}_{20} \times \boldsymbol{\omega}_{20} = \frac{^{(2)}d}{dt}\boldsymbol{\omega}_{20}.$$

where $\dfrac{^{(2)}d}{dt}$ represents the derivative with respect to time in reference frame (2), $[\mathbf{\imath}_2, \mathbf{\jmath}_2, \mathbf{k}_2]$. The angular acceleration of the link 2 is

$$\begin{aligned}\boldsymbol{\alpha}_{20} &= \frac{^{(2)}d}{dt}(\dot{q}_1\, c_2\, \mathbf{\imath}_2 + \dot{q}_2\, \mathbf{\jmath}_2 + \dot{q}_1\, s_2\, \mathbf{k}_2) = \\ &(\ddot{q}_1\, c_2 - \dot{q}_1\, \dot{q}_2\, s_2)\, \mathbf{\imath}_2 + \ddot{q}_2\, \mathbf{\jmath}_2 + (\ddot{q}_1\, s_2 + \dot{q}_1\, \dot{q}_2\, c_2)\, \mathbf{k}_2.\end{aligned} \tag{6.33}$$

The link 3 has the same angular acceleration as link 2, i.e., $\boldsymbol{\alpha}_{30} = \boldsymbol{\alpha}_{20}$.

The MATLAB commands for the angular velocities and accelerations are

```
% angular velocity of link 1 in RF0
% expressed in terms of RF1 {i1,j1,k1}
w10 = [diff(q1,t) 0 0 ];

% angular velocity of link 2 in RF0
% expressed in terms of RF1 {i1,j1,k1}
w201 = [diff(q1,t) diff(q2,t) 0];

% angular velocity of link 2 in RF0
```

```
% expressed in terms of RF2 {i2,j2,k2}
w20 = w201 * transpose(R21);

% angular acceleration of link 1
% in RF0 expressed in terms of RF1 {i1,j1,k1}
alpha10 = diff(w10,t);

% angular acceleration of link 2 in RF0
% expressed in terms of RF2 {i2,j2,k2}
alpha20 = diff(w20,t);
```

Linear velocities

The position vector of C_1, the mass center of link 1, is

$$\mathbf{r}_{C_1} = L_1\mathbf{k}_1,$$

and the velocity of C_1 in (0) is

$$\begin{aligned} \mathbf{v}_{C_1} &= \frac{d}{dt}\mathbf{r}_{C_1} = \frac{^{(1)}d}{dt}\mathbf{r}_{C_1} + \boldsymbol{\omega}_{10} \times \mathbf{r}_{C_1} \\ &= \mathbf{0} + \begin{vmatrix} \mathbf{\imath}_1 & \mathbf{\jmath}_1 & \mathbf{k}_1 \\ \dot{q}_1 & 0 & 0 \\ 0 & 0 & L_1 \end{vmatrix} = -\dot{q}_1 L_1 \mathbf{\jmath}_1. \end{aligned} \tag{6.34}$$

With MATLAB the position and velocity vectors of C_1 are

```
% position vector of mass center C1 of link 1
% in RF0 expressed in terms of RF1 {i1,j1,k1}
rC1 = [0 0 L1];

% linear velocity of mass center C1 of link 1
% in RF0 expressed in terms of RF1 {i1,j1,k1}
vC1 = diff(rC1,t) + cross(w10, rC1);
```

The position vector of C_2, the mass center of link 2, is

$$\begin{aligned} \mathbf{r}_{C_2} &= L_B\mathbf{k}_1 + L_2\mathbf{k}_2 = L_B(-s_2\mathbf{\imath}_2 + c_2\mathbf{k}_2) + L_2\mathbf{k}_2 \\ &= -L_B s_2 \mathbf{\imath}_2 + (L_B c_2 + L_2)\mathbf{k}_2, \end{aligned}$$

where $L_B = 2\,L_1$. The velocity of C_2 in (0) is

$$\begin{aligned}
\mathbf{v}_{C_2} &= \frac{d}{dt}\mathbf{r}_{C_2} = \frac{{}^{(2)}d}{dt}\mathbf{r}_{C_2} + \boldsymbol{\omega}_{20} \times \mathbf{r}_{C_2} \\
&= -L_B c_1 \dot{q}_2 \mathbf{\imath}_2 - L_B c_2 \dot{q}_2 \mathbf{k}_2 + \begin{vmatrix} \mathbf{\imath}_2 & \mathbf{\jmath}_2 & \mathbf{k}_2 \\ \dot{q}_1 c_2 & \dot{q}_2 & \dot{q}_1 s_2 \\ -L_B s_2 & 0 & L_B c_2 + L_2 \end{vmatrix} \\
&= L_2 \dot{q}_2 \mathbf{\imath}_2 - (L_B + L_2 c_2)\dot{q}_1 \mathbf{\jmath}_2. \qquad (6.35)
\end{aligned}$$

The position and velocity vectors of C_2, with MATLAB, are

```
% position vector of mass center C2 of link 2
% in RF0 expressed in terms of RF2 {i2,j2,k2}
rC2 = [0 0 2*L1]*transpose(R21) + [0 0 L2];

% linear velocity of mass center C2 of link 2
% in RF0 expressed in terms of RF2 {i2,j2,k2}
vC2 = simple(diff(rC2,t) + cross(w20,rC2));
```

The position vector of C_3 with respect to reference frame (0) is

$$\begin{aligned}
\mathbf{r}_{C_3} &= \mathbf{r}_{C_2} + q_3 \mathbf{k}_2 \\
&= -L_B s_2 \mathbf{\imath}_2 + (L_B c_2 + L_2 + q_3)\mathbf{k}_2,
\end{aligned}$$

and the velocity of this mass center in (0) is

$$\begin{aligned}
\mathbf{v}_{C_3} &= \frac{d}{dt}\mathbf{r}_{C_3} = \frac{{}^{(2)}d}{dt}\mathbf{r}_{C_3} + \boldsymbol{\omega}_{20} \times \mathbf{r}_{C_3} \\
&= -L_B c_2 \dot{q}_2 \mathbf{\imath}_2 - (L_B c_2 \dot{q}_2 + \dot{q}_3)\mathbf{k}_2 + \begin{vmatrix} \mathbf{\imath}_2 & \mathbf{\jmath}_2 & \mathbf{k}_2 \\ \dot{q}_1 c_2 & \dot{q}_2 & \dot{q}_1 s_2 \\ -L_B s_2 & 0 & L_B c_2 + L_2 + q_3 \end{vmatrix} \\
&= (L_2 + q_3)\dot{q}_2 \mathbf{\imath}_2 - (L_B + L_2 c_2 + c_2 q_2)\dot{q}_1 \mathbf{\jmath}_2 + \dot{q}_3 \mathbf{k}_2. \qquad (6.36)
\end{aligned}$$

The position and velocity vectors of C_3, with MATLAB, are

```
% position vector of mass center C3 of link 3 in RF0
% expressed in terms of RF2 {i2,j2,k2}
rC3 = rC2 + [0 0 q3];
```

```
% linear velocity of mass center C3 of link 3 in RF0
% expressed in terms of RF2 {i2,j2,k2}
vC3 = simple(diff(rC3,t) + cross(w20,rC3));
```

There is a point C_{32} on link 2 ($C_{32} \in$ link2) that instantaneously coincides with C_3, ($C_3 \in$ link3). The velocity of point C_{32} is

$$\begin{aligned} \mathbf{v}_{C_{32}} &= \mathbf{v}_{C_2} + \boldsymbol{\omega}_{20} \times \mathbf{r}_{C_2C_3} = \mathbf{v}_{C_2} + \boldsymbol{\omega}_{20} \times q_3\mathbf{k}_2 = \\ & (L_2 + q_3)\dot{q}_2\mathbf{\imath}_2 - (L_B + L_2c_2 + c_2q_2)\dot{q}_1\mathbf{\jmath}_2. \end{aligned} \tag{6.37}$$

The point C_{32} of link 2 is superposed with the point C_3 of link 3. The velocity of mass center C_3 of link 3 in (0) can be computed in terms of the velocity of C_{32} using the relation

$$\mathbf{v}_{C_3} = \mathbf{v}_{C_{32}} + \dot{q}_3\mathbf{k}_2.$$

The velocity vector of C_{32}, with MATLAB, is

```
vC32 = simple(vC2 + cross(w20,[0 0 q3]));
```

Linear accelerations

The acceleration of C_1 is

$$\begin{aligned} \mathbf{a}_{C_1} &= \frac{d}{dt}\mathbf{v}_{C_1} = \frac{^{(1)}d}{d\,t}\mathbf{v}_{C_1} + \boldsymbol{\omega}_{10} \times \mathbf{v}_{C_1} \\ &= -L_1\ddot{q}_1\mathbf{\jmath} + \begin{vmatrix} \mathbf{\imath}_1 & \mathbf{\jmath}_1 & \mathbf{k}_1 \\ \dot{q}_1 & 0 & 0 \\ 0 & -L_1\dot{q}_1 & 0 \end{vmatrix} \\ &= -L_1\ddot{q}_1\mathbf{\jmath}_1 - L_1\dot{q}_1^2\mathbf{k}_1. \end{aligned} \tag{6.38}$$

The linear acceleration of the mass center C_2 is

$$\mathbf{a}_{C_2} = \frac{d}{dt}\mathbf{v}_{C_2} = \frac{^{(2)}d}{d\,t}\mathbf{v}_{C_2} + \boldsymbol{\omega}_{20} \times \mathbf{v}_{C_2}. \tag{6.39}$$

The linear acceleration of C_2 is symbolically calculated in the program *RRTrobot.nb*.

The acceleration of C_3 is

$$\mathbf{a}_{C_3} = \frac{d}{dt}\mathbf{v}_{C_3} = \frac{^{(2)}d}{d\,t}\mathbf{v}_{C_3} + \boldsymbol{\omega}_{20} \times \mathbf{v}_{C_3}. \tag{6.40}$$

The linear acceleration of C_1, C_2 and C_3 are symbolically calculated with MATLAB

```
aC1 = simple(diff(vC1,t)+cross(w10,vC1));
aC2 = simple(diff(vC2,t)+cross(w20,vC2));
aC3 = simple(diff(vC3,t)+cross(w20,vC3));
```

Generalized forces

Remark: If a set of contact and/or body forces acting on a rigid body is equivalent to a couple of torque $\mathbf{T}$ together with force $\mathbf{R}$ applied at a point P of the rigid body, then the contribution of this set of forces to the generalized force, Q_r, is given by

$$Q_r = \frac{\partial \boldsymbol{\omega}}{\partial \dot{q}_r} \cdot \mathbf{T} + \frac{\partial \mathbf{v}_P}{\partial \dot{q}_r} \cdot \mathbf{R}, \quad r = 1, 2, ...,$$

where $\boldsymbol{\omega}$ is the angular velocity of the rigid body in (0), $\mathbf{v}_P$ is the velocity of P in (0), and r represents the generalized coordinates.

In the case of the robotic arm, there are two kinds of forces that contribute to the generalized forces Q_1, Q_2, and Q_3 namely, contact forces applied in order to drive the links 1, 2, and 3, and gravitational forces exerted on 1, 2, and 3 by the Earth.

The set of contact forces transmitted from 0 to 1 can be replaced with a couple of torque $\mathbf{T}_{01}$ applied to 1 at A.

Similarly, the set of contact forces transmitted from 1 to 2 can be replaced with a couple of torque $\mathbf{T}_{12}$ applied to 2 at B. The law of action and reaction then guarantees that the set of contact forces transmitted from 1 to 2 is equivalent to a couple of torque $-\mathbf{T}_{12}$ to 1 at B.

Next, the set of contact forces exerted by link 2 on link 3 can be replaced with a force $\mathbf{F}_{23}$ applied to 3 at C_3. The law of action and reaction guarantees that the set of contact forces transmitted from 3 to 2 is equivalent to a force $-\mathbf{F}_{23}$ applied to 2 at C_{32}.

The point C_{32} ($C_{32} \in$ link2) instantaneously coincides with C_3, ($C_3 \in$ link3).

The expressions $\mathbf{T}_{01}$, $\mathbf{T}_{12}$, and $\mathbf{F}_{23}$ are

$$\mathbf{T}_{01} = T_{01x}\mathbf{ı}_1 + T_{01y}\mathbf{J}_1 + T_{01z}\mathbf{k}_1, \quad \mathbf{T}_{12} = T_{12x}\mathbf{ı}_2 + T_{12y}\mathbf{J}_2 + T_{12z}\mathbf{k}_2, \quad \text{and}$$

$$\mathbf{F}_{23} = F_{23x}\boldsymbol{\imath}_2 + F_{23y}\boldsymbol{\jmath}_2 + F_{23z}\mathbf{k}_2.$$

The external gravitational forces exerted on the links 1, 2, and 3 by the Earth, can be denoted by $\mathbf{G}_1$, $\mathbf{G}_2$, and $\mathbf{G}_3$ respectively, and can be expressed as

$$\begin{aligned}
\mathbf{G}_1 &= -m_1\, g \boldsymbol{\imath}_1, \\
\mathbf{G}_2 &= -m_2\, g\, \boldsymbol{\imath}_1 = -m_2\, g\, (c_2\, \boldsymbol{\imath}_2 + s_2\, \mathbf{k}_2), \\
\mathbf{G}_3 &= -m_3\, g\, \boldsymbol{\imath}_1 = -m_3\, g\, (c_2\, \boldsymbol{\imath}_2 + s_2\, \mathbf{k}_2).
\end{aligned}$$

The reason for replacing $\boldsymbol{\imath}_1$ with $c_2\, \boldsymbol{\imath}_2 + s_2\, \mathbf{k}_2$ in connection with the forces $\mathbf{G}_2$ and $\mathbf{G}_3$ is that they are soon to be dot-multiplied with $\dfrac{\partial \mathbf{v}_{C_2}}{\partial \dot{q}_r}$ and $\dfrac{\partial \mathbf{v}_{C_3}}{\partial \dot{q}_r}$ which have been expressed in terms of $\boldsymbol{\imath}_2, \boldsymbol{\jmath}_2,$ and $\mathbf{k}_2$.
One can express $(Q_r)_1$, the contribution to the generalized active force Q_r of all the forces and torques acting on the particles of the link 1, as

$$(Q_r)_1 = \frac{\partial \boldsymbol{\omega}_{10}}{\partial \dot{q}_r} \cdot (\mathbf{T}_{01} - \mathbf{T}_{12}) + \frac{\partial \mathbf{v}_{C_1}}{\partial \dot{q}_r} \cdot \mathbf{G}_1, \;\; r = 1, 2, 3.$$

The contribution $(Q_r)_2$ to the generalized active force of all the forces and torques acting on the link 2 is

$$(Q_r)_2 = \frac{\partial \boldsymbol{\omega}_{20}}{\partial \dot{q}_r} \cdot \mathbf{T}_{12} + \frac{\partial \mathbf{v}_{C_2}}{\partial \dot{q}_r} \cdot \mathbf{G}_2 + \frac{\partial \mathbf{v}_{C_{32}}}{\partial \dot{q}_r} \cdot (-\mathbf{F}_{23})\,, \;\; r = 1, 2, 3,$$

where $\mathbf{v}_{C_{32}} = \mathbf{v}_{C_3} - \dot{q}_3 \mathbf{k}_2$.

The contribution $(Q_r)_3$, to the generalized active force of all the forces and torques acting on the link 3 is

$$(Q_r)_3 = \frac{\partial \mathbf{v}_{C_3}}{\partial \dot{q}_r} \cdot \mathbf{G}_3 + \frac{\partial \mathbf{v}_{C_3}}{\partial \dot{q}_r} \cdot \mathbf{F}_{23}, \;\; r = 1, 2, 3.$$

The generalized active force Q_r of all the forces and torques acting on the links 1, 2, and 3 are

$$Q_r = (Q_r)_1 + (Q_r)_2 + (Q_r)_3. \;\; r = 1, 2, 3,$$

The generalized forces Q_r, $r = 1, 2, 3$ are symbolically calculated in the Program 6.3 and have the values

$$\begin{aligned}
&Q_1 = T_{01x}, \\
&Q_2 = T_{12y} - g\, m_2\, L_2\, c_2 - g\, m_3\, c_2\, (L_2 + q_3), \\
&Q_3 = F_{23z} - g\, m_3\, s_2.
\end{aligned} \tag{6.41}$$

The MATLAB statements for the generalized active forces are

```
% gravitational force that acts on link 1 at C1
% RF0 expressed in terms of RF1 {i1,j1,k1}
G1 = [-m1*g 0 0]
% gravitational force that acts on link 2 at C2
% in RF0 expressed in terms of RF2 {i2,j2,k2}
G2 = [-m2*g 0 0]*transpose(R21)
% gravitational force that acts on link 3 at C3
% in RF0 expressed in terms of RF2 {i2,j2,k2}
G3 = [-m3*g 0 0]*transpose(R21)
syms T01x T01y T01z T12x T12y T12z F23x F23y F23z real
% contact torque of 0 that acts on link 1
% in RF0 expressed in terms of RF1 {i1,j1,k1}
T01 = [T01x T01y T01z];
% contact torque of link 1 that acts on link 2
% in RF0 expressed in terms of RF2 {i2,j2,k2}
T12 = [T12x T12y T12z];
% contact force of link 2 that acts on link 3 at C3
% in RF0 expressed in terms of RF2 {i2,j2,k2}
F23 = [F23x F23y F23z];

w1_1 = deriv(w10, diff(q1,t)); w2_1 = deriv(w20, diff(q1,t));
w1_2 = deriv(w10, diff(q2,t)); w2_2 = deriv(w20, diff(q2,t));
w1_3 = deriv(w10, diff(q3,t)); w2_3 = deriv(w20, diff(q3,t));
vC1_1 = deriv(vC1, diff(q1,t)); vC2_1 = deriv(vC2, diff(q1,t));
vC1_2 = deriv(vC1, diff(q2,t)); vC2_2 = deriv(vC2, diff(q2,t));
vC1_3 = deriv(vC1, diff(q3,t)); vC2_3 = deriv(vC2, diff(q3,t));
vC32_1 = deriv(vC32, diff(q1,t)); vC3_1 = deriv(vC3, diff(q1,t));
vC32_2 = deriv(vC32, diff(q2,t)); vC3_2 = deriv(vC3, diff(q2,t));
vC32_3 = deriv(vC32, diff(q3,t)); vC3_3 = deriv(vC3, diff(q3,t));

% generalized active force Q1
Q1 = w1_1*T01.'  + vC1_1*G1.'  + w1_1*transpose(R21)*(-T12.')  +...
     w2_1*T12.'  + vC2_1*G2.'  + vC32_1*(-F23.')  +...
     vC3_1*F23.'  + vC3_1*G3.'

% generalized active force Q2
```

```
Q2 = w1_2*T01.'  + vC1_2*G1.'  + w1_2*transpose(R21)*(-T12.')  +...
     w2_2*T12.'  + vC2_2*G2.'  + vC32_2*(-F23.')  +...
     vC3_2*F23.'  + vC3_2*G3.'

% generalized active force Q3
Q3 = w1_3*T01.'  + vC1_3*G1.'  + w1_3*transpose(R21)*(-T12.')  +...
     w2_3*T12.'  + vC2_3*G2.'  + vC32_3*(-F23.')  +...
     vC3_3*F23.'  + vC3_3*G3.'
```

Lagrange's Equations of Motion

Kinetic energy

The total kinetic energy of the robot arm in the reference frame (0) is

$$T = \sum_{i=1}^{3} T_i.$$

The kinetic energy of the link i, $i = 1, 2, 3$, is

$$T_i = \frac{1}{2} m_i \mathbf{v}_{C_i} \cdot \mathbf{v}_{C_i} + \frac{1}{2} \boldsymbol{\omega}_{i0} \cdot (\bar{I}_i \cdot \boldsymbol{\omega}_{i0}).$$

Remark: The kinetic energy for a rigid body is

$$T_{\text{rigid body}} = \frac{1}{2} m \mathbf{v}_C \cdot \mathbf{v}_C + \frac{1}{2} \boldsymbol{\omega} \cdot (\bar{I}_C \cdot \boldsymbol{\omega}),$$

where m is the mass of the rigid body, $\mathbf{v}_C$ is the velocity of the mass center of the rigid body in (0), $\boldsymbol{\omega} = \omega_x \boldsymbol{\imath} + \omega_y \boldsymbol{\jmath} + \omega_z \mathbf{k}$ is the angular velocity of the rigid body in (0), and $\bar{I} = (I_x \boldsymbol{\imath})\boldsymbol{\imath} + (I_y \boldsymbol{\jmath})\boldsymbol{\jmath} + (I_z \mathbf{k})\mathbf{k}$ is the central inertia dyadic of the rigid body. The central principal axes of the rigid body are parallel to $\boldsymbol{\imath}$, $\boldsymbol{\jmath}$, $\mathbf{k}$ and the associated moments of inertia have the values I_x, I_y, I_z, respectively. The inertia matrix associated with $\bar{I}$ is

$$\bar{I} \rightarrow \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}.$$

The dot product of the vector $\boldsymbol{\omega}$ with the dyadic $\bar{I}$ is

$$\boldsymbol{\omega} \cdot \bar{I} = \bar{I} \cdot \boldsymbol{\omega} \;\; = \;\; \omega_x I_x \, \boldsymbol{\imath} + \omega_y I_y \, \boldsymbol{\jmath} + \omega_z I_z \, \mathbf{k}.$$

The central moments of inertia of links 1 and 2 are calculated using Figure 6.6.

The central principal axes of 1 are parallel to $\mathbf{\imath}_1$, $\mathbf{\jmath}_1$, $\mathbf{k}_1$ and the associated moments of inertia have the values I_{1x}, I_{1y}, I_{1z}, respectively. The inertia matrix associated with link 1 is

$$\bar{I}_1 \rightarrow \begin{bmatrix} I_{1x} & 0 & 0 \\ 0 & I_{1y} & 0 \\ 0 & 0 & I_{1z} \end{bmatrix} = \begin{bmatrix} \dfrac{m_1(2\,L_1)^2}{12} & 0 & 0 \\ 0 & \dfrac{m_1(2\,L_1)^2}{12} & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \dfrac{m_1 L_1^2}{3} & 0 & 0 \\ 0 & \dfrac{m_1 L_1^2}{3} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The central principal axes of 2 and 3 are parallel to $\mathbf{\imath}_2$, $\mathbf{\jmath}_2$, $\mathbf{k}_2$ and the associated moments of inertia have values I_{2x}, I_{2y}, I_{2z}, and I_{3x}, I_{3y}, I_{3z} respectively. The inertia matrix associated with link 2 is

$$\bar{I}_2 \rightarrow \begin{bmatrix} I_{2x} & 0 & 0 \\ 0 & I_{2y} & 0 \\ 0 & 0 & I_{2z} \end{bmatrix} = \begin{bmatrix} \dfrac{m_2(2\,L_2)^2}{12} & 0 & 0 \\ 0 & \dfrac{m_2(2\,L_2)^2}{12} & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \dfrac{m_2 L_2^2}{3} & 0 & 0 \\ 0 & \dfrac{m_2 L_2^2}{3} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The inertia matrix associated with the slider 3 is

$$\bar{I}_3 \rightarrow \begin{bmatrix} I_{3x} & 0 & 0 \\ 0 & I_{3y} & 0 \\ 0 & 0 & I_{3z} \end{bmatrix}$$

The kinetic energy of link 1 is

$$T_1 = \frac{1}{2}m_1\mathbf{v}_{C_1}\cdot\mathbf{v}_{C_1} + \frac{1}{2}\boldsymbol{\omega}_{10}\cdot(\bar{I}_1\cdot\boldsymbol{\omega}_{10}) = \frac{1}{2}m_1 L_1\dot{q}_1^2 + \frac{1}{6}m_1 L_1\dot{q}_1^2 = \frac{2}{3}m_1 L_1\dot{q}_1^2.$$

The kinetic energy of bar 2 is

$$\begin{aligned} T_2 \;&=\; \frac{1}{2}m_2\mathbf{v}_{C_2}\cdot\mathbf{v}_{C_2} + \frac{1}{2}\boldsymbol{\omega}_{20}\cdot(\bar{I}_2\cdot\boldsymbol{\omega}_{20}) = \\ &\frac{m_2}{3}\left[\left(6\,L_1^2 + L_2^2 + 6\,L_1\,L_2\,c_2 + L_2^2\,\cos 2q_2\right)\,\dot{q}_1^2 + 2\,L_2^2\,\dot{q}_2^2\right]. \end{aligned}$$

The kinetic energy of link 3 is

$$\begin{aligned} &T_3 = \frac{1}{2}m_3\mathbf{v}_{C_3}\cdot\mathbf{v}_{C_3} + \frac{1}{2}\boldsymbol{\omega}_{20}\cdot(\bar{I}_3\cdot\boldsymbol{\omega}_{20}) = \\ &\frac{1}{2}\{I_{3x}\,c_2^2\,\dot{q}_1^2 + I_{3z}\,s_2^2\,\dot{q}_1^2 + I_{3y}\,\dot{q}_2^2 + \\ &m_3\,\left[(2\,L_1 + L_2\,c_2 + c_2\,q_3)^2\,\dot{q}_1^2 + (L_2+q_3)^2\,\dot{q}_2^2 + \dot{q}_3^2\right]\}. \end{aligned}$$

The total kinetic energy of the robot arm is

$$T = T_1 + T_2 + T_3,$$

and is symbolically calculated in the Program 6.3.

The left hand sides of Lagrange's equations are

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_r}\right) - \frac{\partial T}{\partial q_r}, \ \ r = 1,\, 2,\, 3.$$

The MATLAB commands for the left hand sides of Lagrange's equations are

```
% central inertia dyadic for link 1
% expressed in terms of RF1 {i1,j1,k1}
I1 = [m1*(2*L1)^2/12 0 0; 0 m1*(2*L1)^2/12 0; 0 0 0];
% central inertia dyadic for link 2
% expressed in terms of RF2 i2,j2,k2
I2 = [m2*(2*L2)^2/12 0 0; 0 m2*(2*L2)^2/12 0; 0 0 0];
% central inertia torque for link 3
% expressed in terms of RF2 {i2,j2,k2}
syms I3x I3y I3z real
I3 = [I3x 0 0; 0 I3y 0; 0 0 I3z];

T1 = (1/2)*m1*vC1*vC1.'  + (1/2)*w10*I1*w10.'
T2 = (1/2)*m2*vC2*vC2.'  + (1/2)*w20*I2*w20.'
T3 = (1/2)*m3*vC3*vC3.'  + (1/2)*w20*I3*w20.'
T = expand(T1 + T2 + T3);   % total kinetic energy
```

To arrive at the dynamical equations governing the robot arm, all that remains to be done is to substitute into Lagrange's equations, namely,

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_r}\right) - \frac{\partial T}{\partial q_r} = Q_r, \ \ r = 1,\, 2,\, 3.$$

The left hand side of Lagrange's equations are symbolically calculated in MATLAB with

```
% deriv(f, g(t)) differentiates f with respect to g(t)
Tdq1 = deriv(T, diff(q1,t));
```

```
Tdq2 = deriv(T, diff(q2,t));
Tdq3 = deriv(T, diff(q3,t));

Tt1 = diff(Tdq1, t);
Tt2 = diff(Tdq2, t);
Tt3 = diff(Tdq3, t);

Tq1 = deriv(T, q1);
Tq2 = deriv(T, q2);
Tq3 = deriv(T, q3);

LHS1 = Tt1 - Tq1;
LHS2 = Tt2 - Tq2;
LHS3 = Tt3 - Tq3;
```

The Lagrange's equations are symbolically calculated in MATLAB with

```
Lagrange1 = LHS1-Q1;
Lagrange2 = LHS2-Q2;
Lagrange3 = LHS3-Q3;
```

The following feedback control laws are used

$$\begin{aligned} T_{01x} &= -\beta_{01}\dot{q}_1 - \gamma_{01}(q_1 - q_{1f}), \\ T_{12y} &= -\beta_{12}\dot{q}_2 - \gamma_{12}(q_2 - q_{2f}) + g\, m_2\, L_2\, c_2 + g\, m_3\, c_2\,(L_2 + q_3), \\ F_{23z} &= -\beta_{23}\dot{q}_3 - \gamma_{23}(q_3 - q_{3f}) + g\, m_3\, s_2. \end{aligned} \tag{6.42}$$

The constant gains are: $\beta_{01} = 450$ N·m·s/rad, $\gamma_{01} = 300$ N·m/rad, $\beta_{12} = 200$ N·m·s/rad, $\gamma_{12} = 300$ N·m/rad, $\beta_{23} = 150$ N·s/m, and $\gamma_{23} = 50$ N/m.

The MATLAB commands for the control torques are

```
q1f=pi/3; q2f=pi/3; q3f=0.3;
b01=450; g01=300;
b12=200; g12=300;
b23=150; g23=50;
T01xc = -b01*diff(q1,t)-g01*(q1-q1f);
T12yc = -b12*diff(q2,t)-g12*(q2-q2f)+g*(m2*L2+m3*(L2+q3))*c2;
F23zc = -b23*diff(q3,t)-g23*(q3-q3f)+g*m3*s2;
```

```
tor = {T01x, T12y, F23z};
torf = {T01xc,T12yc,F23zc};
```

The Lagrange's equations with the feedback control laws are

```
Lagrang1 = subs(Lagrange1, tor, torf);
Lagrang2 = subs(Lagrange2, tor, torf);
Lagrang3 = subs(Lagrange3, tor, torf);
```

The Lagrange's equations with the numerical values for input data are

```
data = {L1, L2, I3x, I3y, I3z, m1, m2, m3, g};
datn = {0.4, 0.4, 5, 4, 1, 90, 60, 40, 9.81};

Lagran1 = subs(Lagrang1, data, datn);
Lagran2 = subs(Lagrang2, data, datn);
Lagran3 = subs(Lagrang3, data, datn);
```

The three second order Lagrange's equations have to be rewritten your as a first order system

```
ql = {diff(q1,t,2), diff(q2,t,2), diff(q3,t,2), ...
      diff(q1,t), diff(q2,t), diff(q3,t), q1, q2, q3};
qf = {'ddq1', 'ddq2', 'ddq3',...
      'x(2)', 'x(4)', 'x(6)', 'x(1)', 'x(3)', 'x(5)'};

% ql                    qf
%----------------------------
% diff('q1(t)',t,2) -> 'ddq1'
% diff('q2(t)',t,2) -> 'ddq2'
% diff('q3(t)',t,2) -> 'ddq3'
%   diff('q1(t)',t) -> 'x(2)'
%   diff('q2(t)',t) -> 'x(4)'
%   diff('q3(t)',t) -> 'x(6)'
%           'q1(t)' -> 'x(1)'
%           'q2(t)' -> 'x(3)'
%           'q3(t)' -> 'x(5)'
```

```
Lagra1 = subs(Lagran1, ql, qf);
Lagra2 = subs(Lagran2, ql, qf);
Lagra3 = subs(Lagran3, ql, qf);

% solve e.o.m.  for ddq1, ddq2, ddq3
sol = solve(Lagra1, Lagra2, Lagra3,'ddq1, ddq2, ddq3');
Lagr1 = sol.ddq1;
Lagr2 = sol.ddq2;
Lagr3 = sol.ddq3;

dx2dt = char(Lagr1);
dx4dt = char(Lagr2);
dx6dt = char(Lagr3);
```

The system of differential equations is solved numerically by m-file functions. The function file, `RRT_Lagr.m` is created using the statements

```
fid = fopen('RRT_Lagr.m','w+');
fprintf(fid,'function dx = RRT_Lagr(t,x)\n');
fprintf(fid,'dx = zeros(6,1);\n');
fprintf(fid,'dx(1) = x(2);\n');
fprintf(fid,'dx(2) = ');
fprintf(fid,dx2dt);
fprintf(fid,';\n');
fprintf(fid,'dx(3) = x(4);\n');
fprintf(fid,'dx(4) = ');
fprintf(fid,dx4dt);
fprintf(fid,';\n');
fprintf(fid,'dx(5) = x(6);\n');
fprintf(fid,'dx(6) = ');
fprintf(fid,dx6dt);
fprintf(fid,';');
fclose(fid);
```

The `ode45` solver is used for the system of differential equations

```
t0 = 0; tf = 15; time = [0 tf];
x0 = [pi/18 0 pi/6 0 0.25 0];
```

```
[t,xs] = ode45(@RRT_Lagr, time, x0);
x1 = xs(:,1);
x2 = xs(:,2);
x3 = xs(:,3);
x4 = xs(:,4);
x5 = xs(:,5);
x6 = xs(:,6);
subplot(3,1,1),plot(t,x1*180/pi,'r'),...
xlabel('t (s)'),ylabel('q1 (deg)'),grid,...
subplot(3,1,2),plot(t,x3*180/pi,'b'),...
xlabel('t (s)'),ylabel('q2 (deg)'),grid,...
subplot(3,1,3),plot(t,x5,'g'),...
xlabel('t (s)'),ylabel('q3 (m)'),grid
[ts,xs] = ode45(@RRT_Lagr,0:1:5,x0);
fprintf('Results \n'); fprintf('\n');
fprintf('t(s) q1(rad) dq1(rad/s) q2(rad) dq2(rad/s) q3(m) dq3(m/s) \n');
[ts,xs]
```

Figure 6.7 shows the plots of $q_1(t)$, $q_2(t)$, $q_3(t)$ and a MATLAB computer program for the direct dynamics is given in the Program 6.3.

Inverse dynamics

A desired motion of the robot arm is specified for a time interval $0 \leq t \leq T_p = 15$ s. The generalized coordinates can be established explicitly

$$q_r(t) = q_r(0) + \frac{q_r(T_p) - q_r(0)}{T_p}\left[t - \frac{T_p}{2\,\pi}\sin\left(\frac{2\,\pi\,t}{T_p}\right)\right], \quad r = 1, 2, 3, \quad (6.43)$$

with $q_r(T_p) = q_{rf}$.

The initial conditions, at $t = 0$ s, are $q_1(0) = \pi/18$ rad, $q_2(0) = \pi/6$ rad, $q_3(0) = 0.25$ m, and $\dot{q}_1(0) = \dot{q}_2(0) = \dot{q}_3(0) = 0$.
The robot arm can be brought from an initial state of rest in reference frame (0) to a final state of rest in (0) in such a way that q_1, q_2, and q_3 have specified values $q_1(T_p) = q_{1f} = \pi/3$ rad, $q_2(T_p) = q_{2f} = \pi/3$ rad, and $q_3(T_p) = q_{3f} = 0.3$ m. Figure 6.8 shows the plots of $q_1(t)$, $q_2(t)$, and $q_3(t)$ rad.

The MATLAB commands for finding the Lagrage's equations are identical with the commands presented at **Direct dynamics**

```
.............................................................
syms T01x T01y T01z T12x T12y T12z F23x F23y F23z real
T01 = [T01x T01y T01z];
T12 = [T12x T12y T12z];
F23 = [F23x F23y F23z];
.............................................................

Lagrange1=LHS1-Q1; Lagrange2=LHS2-Q2; Lagrange3=LHS3-Q3;
data = L1, L2, I3x, I3y, I3z, m1, m2, m3, g;
datn = 0.4, 0.4, 5, 4, 1, 90, 60, 40, 9.81;

Lagra1 = subs(Lagrange1, data, datn);
Lagra2 = subs(Lagrange2, data, datn);
Lagra3 = subs(Lagrange3, data, datn); ;
```

From the Lagrange's equations of motions the torques T_{01x} and T_{12y} and the force F_{23z} are calculated

```
sol = solve(Lagra1,Lagra2,Lagra3,'T01x, T12y, F23z');
T01xc = simple(sol.T01x);
T12yc = simple(sol.T12y);
F23zc = simple(sol.F23z);
```

The generalized coordinates, q_1, q_2, and q_3 given by Eq. (6.43) and their derivatives, $\dot{q}_1$, $\dot{q}_2$, $\dot{q}_3$, $\ddot{q}_1$, $\ddot{q}_2$, $\ddot{q}_3$ are substituted in the expressions of T_{01x}, T_{12y}, and F_{23z}

```
q1s = pi/18; q2s = pi/6; q3s = 0.25;
q1f = pi/3 ; q2f = pi/3; q3f = 0.3;
Tp=15.;
q1t = q1s + (q1f-q1s)/Tp*(t - Tp/(2*pi)*sin(2*pi/Tp*t));
q2t = q2s + (q2f-q2s)/Tp*(t - Tp/(2*pi)*sin(2*pi/Tp*t));
q3t = q3s + (q3f-q3s)/Tp*(t - Tp/(2*pi)*sin(2*pi/Tp*t));
dq1t = diff(q1t,t);
dq2t = diff(q2t,t);
dq3t = diff(q3t,t);
ddq1t = diff(dq1t,t);
```

```
ddq2t = diff(dq2t,t);
ddq3t = diff(dq3t,t);

ql = {diff(q1,t,2), diff(q2,t,2), diff(q3,t,2), ...
      diff(q1,t), diff(q2,t), diff(q3,t), q1, q2, q3};
qn = {ddq1t, ddq2t, ddq3t, dq1t, dq2t, dq3t, q1t, q2t, q3t};

T01xt = subs(T01xc, ql, qn);
T12yt = subs(T12yc, ql, qn);
F23zt = subs(F23zc, ql, qn);
```

The plots of $T_{01x}(t)$, $T_{12y}(t)$, and $F_{23z}(t)$ are obtained with the MATLAB commands

```
time = 0:1:Tp;
T01t = subs(T01xt,'t',time);
T12t = subs(T12yt,'t',time);
F23t = subs(F23zt,'t',time);
subplot(3,1,1),plot(time,T01t),...
xlabel('t (s)'),ylabel('T01x (N m)'),grid,...
subplot(3,1,2),plot(time,T12t),...
xlabel('t (s)'),ylabel('T12y (N m)'),grid,...
subplot(3,1,3),plot(time,F23t),...
xlabel('t (s)'),ylabel('F23z (N)'),grid
```

Figure 6.9 shows the control torques and force and the MATLAB program is given in Program 6.4.

Kane's Dynamical Equations

The generalized coordinates q_i and the generalized speeds u_i are introduced

```
% generalized coordinates q1, q2, q3
q1 = sym('q1(t)'); q2 = sym('q2(t)'); q3 = sym('q3(t)');
% generalized speeds u1, u2, u3
u1 = sym('u1(t)'); u2 = sym('u2(t)'); u3 = sym('u3(t)');
```

The generalized speeds, u_1, u_2, u_3, u_4, are associated with the motion of a system, and can be introduced as $\dot{q}_i = u_i$, or

```
dq1 = u1;
dq2 = u2;
dq3 = u3;

qt = diff(q1,t), diff(q2,t), diff(q3,t);
qu = dq1, dq2, dq3;
```

The velocities and the accelerations of the robot need to be expressed in terms of q_i, u_i and $\dot{u}_i$

```
c1 = cos(q1); s1 = sin(q1); c2 = cos(q2); s2 = sin(q2);
R10 = [[1 0 0]; [0 c1 s1]; [0 -s1 c1]];
R21 = [[c2 0 -s2]; [0 1 0]; [s2 0 c2]];

w10 = [dq1, 0, 0 ]
w201 = [dq1, dq2, 0];
w20 = w201 * transpose(R21)
alpha10 = diff(w10,t)
alpha20 = subs(diff(w20,t), qt, qu)

rC1 = [0 0 L1];
vC1 = diff(rC1,t) + cross(w10, rC1)
rC2 = [0 0 2*L1]*transpose(R21) + [0 0 L2];
vC2 = subs(diff(rC2, t), qt, qu) + cross(w20,rC2)

rC3 = rC2 + [0 0 q3];
```

```
vC3 = subs(diff(rC3, t), qt, qu) + cross(w20,rC3)
vC32 = vC2 + cross(w20,[0 0 q3])

aC1 = diff(vC1,t) + cross(w10,vC1)
aC2 = diff(vC2,t) + cross(w20,vC2)
aC3 = subs(diff(vC3,t), qt, qu) + cross(w20,vC3)
```

The gravitational forces and the external moments and force are

```
G1 = [-m1*g 0 0];
G2 = [-m2*g 0 0]*transpose(R21);
G3 = [-m3*g 0 0]*transpose(R21);
T01 = [T01x T01y T01z];
T12 = [T12x T12y T12z];
F23 = [F23x F23y F23z];
```

The partial velocities with respect to `u1, u2, u3` are calculated using the function `deriv`

```
w1_1 = deriv(w10, u1); w2_1 = deriv(w20, u1);
w1_2 = deriv(w10, u2); w2_2 = deriv(w20, u2);
w1_3 = deriv(w10, u3); w2_3 = deriv(w20, u3);

vC1_1 = deriv(vC1, u1); vC2_1 = deriv(vC2, u1);
vC1_2 = deriv(vC1, u2); vC2_2 = deriv(vC2, u2);
vC1_3 = deriv(vC1, u3); vC2_3 = deriv(vC2, u3);

vC32_1 = deriv(vC32, u1); vC3_1 = deriv(vC3, u1);
vC32_2 = deriv(vC32, u2); vC3_2 = deriv(vC3, u2);
vC32_3 = deriv(vC32, u3); vC3_3 = deriv(vC3, u3);
```

The generalized active forces are

```
Q1 = w1_1*T01.'  + vC1_1*G1.'  + w1_1*transpose(R21)*(-T12.')  +...
     w2_1*T12.'  + vC2_1*G2.'  + vC32_1*(-F23.')  +...
     vC3_1*F23.'  + vC3_1*G3.';

Q2 = w1_2*T01.'  + vC1_2*G1.'  + w1_2*transpose(R21)*(-T12.')  +...
```

```
     w2_2*T12.'  + vC2_2*G2.'  + vC32_2*(-F23.')  +...
     vC3_2*F23.'  + vC3_2*G3.';

Q3 = w1_3*T01.'  + vC1_3*G1.'  + w1_3*transpose(R21)*(-T12.')  +...
     w2_3*T12.'  + vC2_3*G2.'  + vC32_3*(-F23.')  +...
     vC3_3*F23.'  + vC3_3*G3.';
```

Generalized inertia forces

To explain what the *generalized inertia forces* are, a system $\{S\}$ formed by ν particles $P_1, ..., P_\nu$ and having masses $m_1, ..., m_\nu$ is considered. Suppose that n generalized speeds u_r, $r = 1, ..., n$ have been introduced. (For the robotic arm $u_r = \dot{q}_r$, $r = 1, ..., n$.) Let $\mathbf{v}_{P_j}$ and $\mathbf{a}_{P_j}$ denote, respectively, the velocity of P_j and the acceleration of P_j in a reference frame (0).

Define $\mathbf{F}_{in\,j}$, called the inertia force for P_j, as

$$\mathbf{F}_{in\,j} = -m_j\,\mathbf{a}_{P_j}.$$

The quantities $K_{in\,1}, ..., K_{in\,n}$, defined as

$$K_{in\,r} = \sum_{j=1}^{\nu} \frac{\partial \mathbf{v}_{P_j}}{\partial u_r} \cdot \mathbf{F}_{\text{in}\,j}, \quad r = 1, ..., n,$$

are called *generalized inertia forces* for $\{S\}$.
The contribution to $K_{in\,r}$, made by the particles of a rigid body RB belonging to $\{S\}$, are

$$(K_{in\,r})_R = \frac{\partial \mathbf{v}_C}{\partial u_r} \cdot \mathbf{F}_{\text{in}} + \frac{\partial \boldsymbol{\omega}}{\partial u_r} \cdot \mathbf{M}_{\text{in}}, \quad r = 1, ..., n,$$

where $\mathbf{v}_C$ is the velocity of the center of gravity of RB in (0), and $\boldsymbol{\omega} = \omega_x\mathbf{\imath} + \omega_y\mathbf{\jmath} + \omega_z\mathbf{k}$ is the angular velocity of RB in (0).
The inertia force for the rigid body RB is

$$\mathbf{F}_{in} = -m\,\mathbf{a}_C,$$

where m is the mass of RB, and $\mathbf{a}_C$ is the acceleration of the mass center of RB in the fixed reference frame. The inertia moment $\mathbf{M}_{in}$ for RB is

$$\mathbf{M}_{in} \;\; = \;\; -\boldsymbol{\alpha} \cdot \bar{I} - \boldsymbol{\omega} \times (\bar{I} \cdot \boldsymbol{\omega}),$$

where $\boldsymbol{\alpha} = \dot{\boldsymbol{\omega}} = \alpha_x \boldsymbol{\imath} + \alpha_y \boldsymbol{\jmath} + \alpha_z \mathbf{k}$ is the angular acceleration of RB in (0), and $\bar{I} = (I_x \boldsymbol{\imath})\boldsymbol{\imath} + (I_y \boldsymbol{\jmath})\boldsymbol{\jmath} + (I_z \mathbf{k})\mathbf{k}$ is the central inertia dyadic of RB. The central principal axes of RB are parallel to $\boldsymbol{\imath}$, $\boldsymbol{\jmath}$, $\mathbf{k}$ and the associated moments of inertia have the values I_x, I_y, I_z, respectively. The inertia matrix associated with $\bar{I}$ is

$$\bar{I} \rightarrow \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}.$$

The dot product of the vector $\boldsymbol{\alpha}$ with the dyadic $\bar{I}$ is

$$\boldsymbol{\alpha} \cdot \bar{I} = \bar{I} \cdot \boldsymbol{\alpha} \quad = \quad \alpha_x I_x \, \boldsymbol{\imath} + \alpha_y I_y \, \boldsymbol{\jmath} + \alpha_z I_z \, \mathbf{k},$$

and the cross product between a vector and a dyadic is

$$\boldsymbol{\omega} \times (\bar{I} \cdot \boldsymbol{\omega}) = \begin{vmatrix} \boldsymbol{\imath} & \boldsymbol{\jmath} & \mathbf{k} \\ \omega_x & \omega_y & \omega_z \\ \omega_x I_x & \omega_y I_y & \omega_z I_z \end{vmatrix} =$$
$$-\omega_y \omega_z (I_y - I_z)\boldsymbol{\imath} - \omega_z \omega_x (I_z - I_x)\boldsymbol{\jmath} - \omega_x \omega_y (I_x - I_y)\mathbf{k}.$$

The inertia moment of 1 in (0) can be written as

$$\mathbf{M}_{in\,1} = -\boldsymbol{\alpha}_{10} \cdot \bar{I}_1 - \boldsymbol{\omega}_{10} \times (\bar{I}_1 \cdot \boldsymbol{\omega}_{10}) = -I_{1x} \ddot{q}_1 \boldsymbol{\imath}_1.$$

The inertia moment of 2 in (0) is

$$\mathbf{M}_{in\,2} = -\boldsymbol{\alpha}_{20} \cdot \bar{I}_2 - \boldsymbol{\omega}_{20} \times (\bar{I}_2 \cdot \boldsymbol{\omega}_{20}).$$

Similarly the inertia moment of 3 in (0) is

$$\mathbf{M}_{in\,3} = \boldsymbol{\alpha}_{20} \cdot \bar{I}_3 - \boldsymbol{\omega}_{20} \times (\bar{I}_3 \cdot \boldsymbol{\omega}_{20}).$$

The inertia force for link $j = 1, 2, 3$ is

$$\mathbf{F}_{in\,j} = -m_j \, \mathbf{a}_{C_j},$$

The contribution of link $j = 1, 2, 3$ to the generalized inertia force $K_{in\,r}$ is

$$(K_{in\,r})_j = \frac{\partial \mathbf{v}_{C_j}}{\partial u_r} \cdot \mathbf{F}_{\text{in}\ j} + \frac{\partial \boldsymbol{\omega}_{j0}}{\partial u_r} \cdot \mathbf{M}_{\text{in}\ j}, \quad r = 1, 2, 3.$$

The three generalized inertia forces are computed with

$$K_{in\,r} = \sum_{j=1}^{3}(K_{in\,r})_j = \sum_{j=1}^{3}\left(\frac{\partial \mathbf{v}_{C_j}}{\partial u_r}\cdot \mathbf{F}_{in\,j} + \frac{\partial \boldsymbol{\omega}_{j0}}{\partial u_r}\cdot \mathbf{M}_{in\,j}\right), \quad r = 1,\, 2,\, 3,$$

or

$$K_{in\,r} = \frac{\partial \mathbf{v}_{C_1}}{\partial u_r}\cdot(-m_1\,\mathbf{a}_{C_1}) + \frac{\partial \boldsymbol{\omega}_{10}}{\partial u_r}\cdot \mathbf{M}_{in\,1} + \frac{\partial \mathbf{v}_{C_2}}{\partial u_r}\cdot(-m_2\,\mathbf{a}_{C_2}) + \frac{\partial \boldsymbol{\omega}_{20}}{\partial u_r}\cdot \mathbf{M}_{in\,2} + \frac{\partial \mathbf{v}_{C_3}}{\partial u_r}\cdot(-m_3\,\mathbf{a}_{C_3}) + \frac{\partial \boldsymbol{\omega}_{30}}{\partial u_r}\cdot \mathbf{M}_{in\,3}, \quad r = 1,\, 2,\, 3.$$

The generalized inertia forces for the RRT robot arm are calculated with the following MATLAB commands

```
I1 = [m1*(2*L1)^2/12 0 0; 0 m1*(2*L1)^2/12 0; 0 0 0];
I2 = [m2*(2*L2)^2/12 0 0; 0 m2*(2*L2)^2/12 0; 0 0 0];
I3 = [I3x 0 0; 0 I3y 0; 0 0 I3z];

% inertia force for link 1 expressed in terms of RF1 {i1,j1,k1}
Fin1= -m1*aC1;
% inertia force for link 2 expressed in terms of RF2 {i2,j2,k2}
Fin2= -m2*aC2;
% inertia force for link 3 expressed in terms of RF2 {i2,j2,k2}
Fin3= -m3*aC3;
% inertia moment for link 1 expressed in terms of RF1 {i1,j1,k1}
Min1 = -alpha10*I1-cross(w10,w10*I1);
% inertia moment for link 2 expressed in terms of RF2 {i2,j2,k2}
Min2 = -alpha20*I2-cross(w20,w20*I2);
% inertia moment for link 3 expressed in terms of RF2 {i2,j2,k2}
Min3 = -alpha20*I3-cross(w20,w20*I3);

% generalized inertia forces corresponding to q1
Kin1 = w1_1*Min1.'  + vC1_1*Fin1.'  + ...
       w2_1*Min2.'  + vC2_1*Fin2.'  + ...
       w2_1*Min3.'  + vC3_1*Fin3.';
```

```
% generalized inertia forces corresponding to q2
Kin2 = w1_2*Min1.'  + vC1_2*Fin1.'  + ...
       w2_2*Min2.'  + vC2_2*Fin2.'  + ...
       w2_2*Min3.'  + vC3_2*Fin3.';

% generalized inertia forces corresponding to q3
Kin3 = w1_3*Min1.'  + vC1_3*Fin1.'  + ...
       w2_3*Min2.'  + vC2_3*Fin2.'  + ...
       w2_3*Min3.'  + vC3_3*Fin3.';
```

To arrive at the dynamical equations governing the robot arm, all that remains to be done is to substitute into Kane's dynamical equations, namely,

$$K_{in\,r} + Q_r = 0, \;\; r = 1,\, 2,\, 3. \tag{6.44}$$

The Kane's dynamical equations in MATLAB are

```
Kane1 = Kin1 + Q1; Kane2 = Kin2 + Q2; Kane3 = Kin3 + Q3;
```

Using the same feedback control laws (the same as the ones used at Lagrange's equations) the Kane's equations have to be rewritten

```
q1f=pi/3; q2f=pi/3; q3f=0.3;
b01=450; g01=300; b12=200; g12=300; b23=150; g23=50;

T01xc = -b01*dq1-g01*(q1-q1f);
T12yc = -b12*dq2-g12*(q2-q2f)+g*(m2*L2+m3*(L2+q3))*c2;
F23zc = -b23*dq3-g23*(q3-q3f)+g*m3*s2;

tor = T01x, T12y, F23z;
torf = T01xc,T12yc,F23zc;

Kan1 = subs(Kane1, tor, torf);
Kan2 = subs(Kane2, tor, torf);
Kan3 = subs(Kane3, tor, torf);
```

The Kane's dynamical equations can be expressed in terms of $\dot{u}_1$, $\dot{u}_2$, and $\dot{u}_3$

```
data = L1, L2, I3x, I3y, I3z, m1, m2, m3, g;
datn = 0.4, 0.4, 5, 4, 1, 90, 60, 40, 9.81;

Ka1 = subs(Kan1, data, datn);
Ka2 = subs(Kan2, data, datn);
Ka3 = subs(Kan3, data, datn);

ql = {diff(u1,t), diff(u2,t), diff(u3,t) ...
     u1, u2, u3, q1, q2, q3};
qx = {'du1', 'du2', 'du3',...
     'x(4)', 'x(5)', 'x(6)', 'x(1)', 'x(2)', 'x(3)'}

Du1 = subs(Ka1, ql, qx);
Du2 = subs(Ka2, ql, qx);
Du3 = subs(Ka3, ql, qx);

% solve for du1, du2, du3
sol = solve(Du1, Du2, Du3,'du1, du2, du3');
sdu1 = sol.du1;
sdu2 = sol.du2;
sdu3 = sol.du3; ;
```

The system of differential equations is solved numerically by m-file functions. The function file, RRT_Kane.m is created using the statements

```
% system of ODE
dx1 = char('x(4)');
dx2 = char('x(5)');
dx3 = char('x(6)');
dx4 = char(sdu1);
dx5 = char(sdu2);
dx6 = char(sdu3);

fid = fopen('RRT_Kane.m','w+');
fprintf(fid,'function dx = RRT_Kane(t,x)\n');
fprintf(fid,'dx = zeros(6,1);\n');
fprintf(fid,'dx(1) = '); fprintf(fid,dx1); fprintf(fid,';\n');
```

```
fprintf(fid,'dx(2) = '); fprintf(fid,dx2); fprintf(fid,';\n');
fprintf(fid,'dx(3) = '); fprintf(fid,dx3); fprintf(fid,';\n');
fprintf(fid,'dx(4) = '); fprintf(fid,dx4); fprintf(fid,';\n');
fprintf(fid,'dx(5) = '); fprintf(fid,dx5); fprintf(fid,';\n');
fprintf(fid,'dx(6) = '); fprintf(fid,dx6); fprintf(fid,'; ');
fclose(fid);
```

The ode45 solver is used for the system of differential equations

```
t0 = 0; tf = 15; time = [0 tf];
x0 = [pi/18 pi/6 0.25 0 0 0];
[t,xs] = ode45(@RRT_Kane, time, x0);
x1 = xs(:,1);
x2 = xs(:,2);
x3 = xs(:,3);
x4 = xs(:,4);
x5 = xs(:,5);
x6 = xs(:,6);
subplot(3,1,1), plot(t,x1*180/pi,'r'),...
xlabel('t (s)'), ylabel('q1 (deg)'), grid,...
subplot(3,1,2), plot(t,x2*180/pi,'b'),...
xlabel('t (s)'), ylabel('q2 (deg)'), grid,...
subplot(3,1,3), plot(t,x3,'g'),...
xlabel('t (s)'), ylabel('q3 (m)'), grid
```

The MATLAB computer program for the direct dynamics using Kane's dynamical equations is given in the Program 6.5.