Chapter 8

Dynamics of Open Chains

In this chapter we study once again the motions of open-chain robots, but this time taking into account the forces and torques that cause them; this is the subject of **robot dynamics**. The associated dynamic equations – also referred to as the **equations of motion** – are a set of second-order differential equations of the form

$$\tau = M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}), \tag{8.1}$$

where $\theta \in \mathbb{R}^n$ is the vector of joint variables, $\tau \in \mathbb{R}^n$ is the vector of joint forces and torques, $M(\theta) \in \mathbb{R}^{n \times n}$ is a symmetric positive-definite **mass matrix**, and $h(\theta, \dot{\theta}) \in \mathbb{R}^n$ are forces that lump together centripetal, Coriolis, gravity, and friction terms that depend on θ and $\dot{\theta}$. One should not be deceived by the apparent simplicity of these equations; even for "simple" open chains, e.g., those with joint axes that are either orthogonal or parallel to each other, $M(\theta)$ and $h(\theta, \dot{\theta})$ can be extraordinarily complex.

Just as a distinction was made between a robot's forward and inverse kinematics, it is also customary to distinguish between a robot's **forward** and **inverse dynamics**. The forward problem is the problem of determining the robot's acceleration $\ddot{\theta}$ given the state $(\theta, \dot{\theta})$ and the joint forces and torques,

$$\ddot{\theta} = M^{-1}(\theta)\left(\tau - h(\theta, \dot{\theta})\right), \tag{8.2}$$

and the inverse problem is finding the joint forces and torques τ corresponding to the robot's state and a desired acceleration, i.e., Equation (8.1).

A robot's dynamic equations are typically derived in one of two ways: by a direct application of Newton's and Euler's dynamic equations for a rigid body (often called the **Newton–Euler formulation**) or by the **Lagrangian dynamics** formulation derived from the kinetic and potential energy of the robot.

The Lagrangian formalism is conceptually elegant and quite effective for robots with simple structures, e.g., with three or fewer degrees of freedom. The calculations can quickly become cumbersome for robots with more degrees of freedom, however. For general open chains, the Newton–Euler formulation leads to efficient recursive algorithms for both the inverse and forward dynamics that can also be assembled into closed-form analytic expressions for, e.g., the mass matrix $M(\theta)$ and the other terms in the dynamics equation (8.1). The Newton–Euler formulation also takes advantage of tools we have already developed in this book.

In this chapter we study both the Lagrangian and Newton–Euler dynamics formulations for an open-chain robot. While we usually express the dynamics in terms of the joint space variables θ, it is sometimes convenient to express it in terms of the configuration, twist, and rate of change of the twist of the end-effector. This is the task-space dynamics, studied in Section 8.6. Sometimes robots are subject to a set of constraints on their motion, such as when the robot makes contact with a rigid environment. This leads to a formulation of the constrained dynamics (Section 8.7), whereby the space of joint torques and forces is divided into a subspace that causes motion of the robot and a subspace that causes forces against the constraints. The URDF file format for specifying robot inertial properties is described in Section 8.8. Finally, some practical issues that arise in the derivation of robot dynamics, such as the effect of motor gearing and friction, are described in Section 8.9.

8.1 Lagrangian Formulation

8.1.1 Basic Concepts and Motivating Examples

The first step in the Lagrangian formulation of dynamics is to choose a set of independent coordinates $q \in \mathbb{R}^n$ that describes the system's configuration. The coordinates q are called **generalized coordinates**. Once generalized coordinates have been chosen, these then define the **generalized forces** $f \in \mathbb{R}^n$. The forces f and the coordinate rates $\dot{q}$ are dual to each other in the sense that the inner product $f^{\mathrm{T}}\dot{q}$ corresponds to power. A Lagrangian function $\mathcal{L}(q, \dot{q})$ is then defined as the overall system's kinetic energy $\mathcal{K}(q, \dot{q})$ minus the potential energy $\mathcal{P}(q)$,

$$\mathcal{L}(q, \dot{q}) = \mathcal{K}(q, \dot{q}) - \mathcal{P}(q).$$

The equations of motion can now be expressed in terms of the Lagrangian as follows:

$$f = \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q}, \tag{8.3}$$

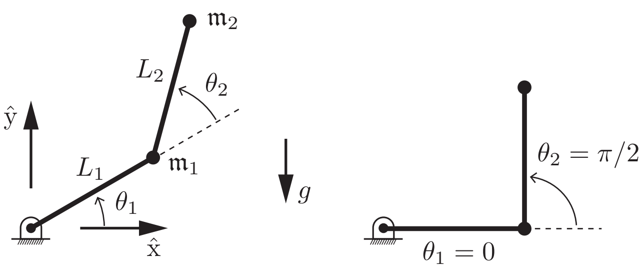


Figure 8.1: (Left) A 2R open chain under gravity. (Right) At $\theta = (0, \pi/2)$.

These equations are also referred to as the **Euler–Lagrange equations with external forces**.[1] The derivation can be found in dynamics texts.

We illustrate the Lagrangian dynamics formulation through two examples. In the first example, consider a particle of mass $\mathfrak{m}$ constrained to move on a vertical line. The particle's configuration space is this vertical line, and a natural choice for a generalized coordinate is the height of the particle, which we denote by the scalar variable $x \in \mathbb{R}$. Suppose that the gravitational force $\mathfrak{m}g$ acts downward, and an external force f is applied upward. By Newton's second law, the equation of motion for the particle is

$$f - \mathfrak{m}g = \mathfrak{m}\ddot{x}. \tag{8.4}$$

We now apply the Lagrangian formalism to derive the same result. The kinetic energy is $\mathfrak{m}\dot{x}^2/2$, the potential energy is $\mathfrak{m}gx$, and the Lagrangian is

$$\mathcal{L}(x, \dot{x}) = \mathcal{K}(x, \dot{x}) - \mathcal{P}(x) = \frac{1}{2}\mathfrak{m}\dot{x}^2 - \mathfrak{m}gx. \tag{8.5}$$

The equation of motion is then given by

$$f = \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{x}} - \frac{\partial \mathcal{L}}{\partial x} = \mathfrak{m}\ddot{x} + \mathfrak{m}g, \tag{8.6}$$

which matches Equation (8.4).

We now derive the dynamic equations for a planar 2R open chain moving in the presence of gravity (Figure 8.1). The chain moves in the $\hat{\mathrm{x}}$–$\hat{\mathrm{y}}$-plane, with gravity g acting in the $-\hat{\mathrm{y}}$-direction. Before the dynamics can be derived, the mass and inertial properties of all the links must be specified. To keep things simple the two links are modeled as point masses $\mathfrak{m}_1$ and $\mathfrak{m}_2$ concentrated at

[1] The external force f is zero in the standard form of the Euler–Lagrange equations.

the ends of each link. The position and velocity of the link-1 mass are then given by

$$\begin{aligned} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} &= \begin{bmatrix} L_1 \cos\theta_1 \\ L_1 \sin\theta_1 \end{bmatrix}, \\ \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \end{bmatrix} &= \begin{bmatrix} -L_1 \sin\theta_1 \\ L_1 \cos\theta_1 \end{bmatrix} \dot{\theta}_1, \end{aligned}$$

while those of the link-2 mass are given by

$$\begin{aligned} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} &= \begin{bmatrix} L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2) \\ L_1 \sin\theta_1 + L_2 \sin(\theta_1+\theta_2) \end{bmatrix}, \\ \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} &= \begin{bmatrix} -L_1 \sin\theta_1 - L_2 \sin(\theta_1+\theta_2) & -L_2 \sin(\theta_1+\theta_2) \\ L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2) & L_2 \cos(\theta_1+\theta_2) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}. \end{aligned}$$

We choose the joint coordinates $\theta = (\theta_1, \theta_2)$ as the generalized coordinates. The generalized forces $\tau = (\tau_1, \tau_2)$ then correspond to joint torques (since $\tau^{\mathrm{T}}\dot{\theta}$ corresponds to power). The Lagrangian $\mathcal{L}(\theta, \dot{\theta})$ is of the form

$$\mathcal{L}(\theta, \dot{\theta}) = \sum_{i=1}^{2} (\mathcal{K}_i - \mathcal{P}_i), \tag{8.7}$$

where the link kinetic energy terms $\mathcal{K}_1$ and $\mathcal{K}_2$ are

$$\begin{aligned} \mathcal{K}_1 &= \frac{1}{2}\mathfrak{m}_1(\dot{x}_1^2 + \dot{y}_1^2) = \frac{1}{2}\mathfrak{m}_1 L_1^2 \dot{\theta}_1^2 \\ \mathcal{K}_2 &= \frac{1}{2}\mathfrak{m}_2(\dot{x}_2^2 + \dot{y}_2^2) \\ &= \frac{1}{2}\mathfrak{m}_2 \left((L_1^2 + 2L_1L_2\cos\theta_2 + L_2^2)\dot{\theta}_1^2 + 2(L_2^2 + L_1L_2\cos\theta_2)\dot{\theta}_1\dot{\theta}_2 + L_2^2\dot{\theta}_2^2 \right), \end{aligned}$$

and the link potential energy terms $\mathcal{P}_1$ and $\mathcal{P}_2$ are

$$\begin{aligned} \mathcal{P}_1 &= \mathfrak{m}_1 g y_1 = \mathfrak{m}_1 g L_1 \sin\theta_1, \\ \mathcal{P}_2 &= \mathfrak{m}_2 g y_2 = \mathfrak{m}_2 g (L_1 \sin\theta_1 + L_2 \sin(\theta_1+\theta_2)). \end{aligned}$$

The Euler–Lagrange equations (8.3) for this example are of the form

$$\tau_i = \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i} - \frac{\partial \mathcal{L}}{\partial \theta_i}, \qquad i = 1, 2. \tag{8.8}$$

The dynamic equations for the 2R planar chain follow from explicit evaluation of the right-hand side of (8.8) (we omit the detailed calculations, which are straightforward but tedious):

$$\left.\begin{aligned}
\tau_1 &= \big(\mathfrak{m}_1 L_1^2 + \mathfrak{m}_2(L_1^2 + 2L_1L_2\cos\theta_2 + L_2^2)\big)\,\ddot{\theta}_1 \\
&\quad + \mathfrak{m}_2(L_1L_2\cos\theta_2 + L_2^2)\ddot{\theta}_2 - \mathfrak{m}_2L_1L_2\sin\theta_2(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2) \\
&\quad + (\mathfrak{m}_1 + \mathfrak{m}_2)L_1 g\cos\theta_1 + \mathfrak{m}_2 g L_2\cos(\theta_1 + \theta_2), \\
\tau_2 &= \mathfrak{m}_2(L_1L_2\cos\theta_2 + L_2^2)\ddot{\theta}_1 + \mathfrak{m}_2L_2^2\ddot{\theta}_2 + \mathfrak{m}_2L_1L_2\dot{\theta}_1^2\sin\theta_2 \\
&\quad + \mathfrak{m}_2 g L_2\cos(\theta_1 + \theta_2).
\end{aligned}\right\} \tag{8.9}$$

We can gather terms together into an equation of the form

$$\tau = M(\theta)\ddot{\theta} + \underbrace{c(\theta,\dot{\theta}) + g(\theta)}_{h(\theta,\dot{\theta})}, \tag{8.10}$$

with

$$\begin{aligned}
M(\theta) &= \begin{bmatrix} \mathfrak{m}_1L_1^2 + \mathfrak{m}_2(L_1^2 + 2L_1L_2\cos\theta_2 + L_2^2) & \mathfrak{m}_2(L_1L_2\cos\theta_2 + L_2^2) \\ \mathfrak{m}_2(L_1L_2\cos\theta_2 + L_2^2) & \mathfrak{m}_2L_2^2 \end{bmatrix}, \\
c(\theta,\dot{\theta}) &= \begin{bmatrix} -\mathfrak{m}_2L_1L_2\sin\theta_2(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2) \\ \mathfrak{m}_2L_1L_2\dot{\theta}_1^2\sin\theta_2 \end{bmatrix}, \\
g(\theta) &= \begin{bmatrix} (\mathfrak{m}_1 + \mathfrak{m}_2)L_1 g\cos\theta_1 + \mathfrak{m}_2 g L_2\cos(\theta_1 + \theta_2) \\ \mathfrak{m}_2 g L_2\cos(\theta_1 + \theta_2) \end{bmatrix},
\end{aligned}$$

where $M(\theta)$ is the symmetric positive-definite mass matrix, $c(\theta,\dot{\theta})$ is the vector containing the Coriolis and centripetal torques, and $g(\theta)$ is the vector containing the gravitational torques. These reveal that the equations of motion are linear in $\ddot{\theta}$, quadratic in $\dot{\theta}$, and trigonometric in θ. This is true in general for serial chains containing revolute joints, not just for the 2R robot.

The $M(\theta)\ddot{\theta} + c(\theta,\dot{\theta})$ terms in Equation (8.10) could have been derived by writing $f_i = \mathfrak{m}_i a_i$ for each point mass, where the accelerations a_i are written in terms of θ, by differentiating the expressions for $(\dot{x}_1, \dot{y}_1)$ and $(\dot{x}_2, \dot{y}_2)$ given above:

$$f_1 = \begin{bmatrix} f_{x1} \\ f_{y1} \\ f_{z1} \end{bmatrix} = \mathfrak{m}_1 \begin{bmatrix} \ddot{x}_1 \\ \ddot{y}_1 \\ \ddot{z}_1 \end{bmatrix} = \mathfrak{m}_1 \begin{bmatrix} -L_1\dot{\theta}_1^2 \mathrm{c}_1 - L_1\ddot{\theta}_1\mathrm{s}_1 \\ -L_1\dot{\theta}_1^2\mathrm{s}_1 + L_1\ddot{\theta}_1\mathrm{c}_1 \\ 0 \end{bmatrix}, \tag{8.11}$$

$$f_2 = \mathfrak{m}_2 \begin{bmatrix} -L_1\dot{\theta}_1^2\mathrm{c}_1 - L_2(\dot{\theta}_1 + \dot{\theta}_2)^2\mathrm{c}_{12} - L_1\ddot{\theta}_1\mathrm{s}_1 - L_2(\ddot{\theta}_1 + \ddot{\theta}_2)\mathrm{s}_{12} \\ -L_1\dot{\theta}_1^2\mathrm{s}_1 - L_2(\dot{\theta}_1 + \dot{\theta}_2)^2\mathrm{s}_{12} + L_1\ddot{\theta}_1\mathrm{c}_1 + L_2(\ddot{\theta}_1 + \ddot{\theta}_2)\mathrm{c}_{12} \\ 0 \end{bmatrix}, \tag{8.12}$$

where s_{12} indicates $\sin(\theta_1+\theta_2)$, etc. Defining r_{11} as the vector from joint 1 to $\mathfrak{m}_1$, r_{12} as the vector from joint 1 to $\mathfrak{m}_2$, and r_{22} as the vector from joint 2 to $\mathfrak{m}_2$, the moments in world-aligned frames $\{i\}$ attached to joints 1 and 2 can be expressed as $m_1 = r_{11}\times f_1 + r_{12}\times f_2$ and $m_2 = r_{22}\times f_2$. (Note that joint 1 must provide torques to move both $\mathfrak{m}_1$ and $\mathfrak{m}_2$, but joint 2 only needs to provide torque to move $\mathfrak{m}_2$.) The joint torques τ_1 and τ_2 are just the third elements of m_1 and m_2, i.e., the moments about the $\hat{z}_i$ axes out of the page, respectively.

In (x, y) coordinates, the accelerations of the masses are written simply as second time-derivatives of the coordinates, e.g., $(\ddot{x}_2, \ddot{y}_2)$. This is because the $\hat{x}$-$\hat{y}$ frame is an inertial frame. The joint coordinates (θ_1, θ_2) are not in an inertial frame, however, so accelerations are expressed as a sum of terms that are linear in the second derivatives of joint variables, $\ddot{\theta}$, and quadratic of the first derivatives of joint variables, $\dot{\theta}^{\mathrm{T}}\dot{\theta}$, as seen in Equations (8.11) and (8.12). Quadratic terms containing $\dot{\theta}_i^2$ are called **centripetal** terms, and quadratic terms containing $\dot{\theta}_i\dot{\theta}_j, i \neq j$, are called **Coriolis** terms. In other words, $\ddot{\theta}=0$ does not mean zero acceleration of the masses, due to the centripetal and Coriolis terms.

To better understand the centripetal and Coriolis terms, consider the arm at the configuration $(\theta_1, \theta_2) = (0, \pi/2)$, i.e., $\cos\theta_1 = \sin(\theta_1+\theta_2) = 1$, $\sin\theta_1 = \cos(\theta_1+\theta_2) = 0$. Assuming $\ddot{\theta}=0$, the acceleration $(\ddot{x}_2, \ddot{y}_2)$ of $\mathfrak{m}_2$ from Equation (8.12) can be written

$$\begin{bmatrix} \ddot{x}_2 \\ \ddot{y}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} -L_1\dot{\theta}_1^2 \\ -L_2\dot{\theta}_1^2 - L_2\dot{\theta}_2^2 \end{bmatrix}}_{\text{centripetal terms}} + \underbrace{\begin{bmatrix} 0 \\ -2L_2\dot{\theta}_1\dot{\theta}_2 \end{bmatrix}}_{\text{Coriolis terms}}.$$

Figure 8.2 shows the centripetal acceleration $a_{\mathrm{cent1}} = (-L_1\dot{\theta}_1^2, -L_2\dot{\theta}_1^2)$ when $\dot{\theta}_2 = 0$, the centripetal acceleration $a_{\mathrm{cent2}} = (0, -L_2\dot{\theta}_2^2)$ when $\dot{\theta}_1 = 0$, and the Coriolis acceleration $a_{\mathrm{cor}} = (0, -2L_2\dot{\theta}_1\dot{\theta}_2)$ when both $\dot{\theta}_1$ and $\dot{\theta}_2$ are positive. As illustrated in Figure 8.2, each centripetal acceleration $a_{\mathrm{cent}i}$ pulls $\mathfrak{m}_2$ toward joint i to keep $\mathfrak{m}_2$ rotating about the center of the circle defined by joint i.[2] Therefore $a_{\mathrm{cent}i}$ creates zero torque about joint i. The Coriolis acceleration a_{cor} in this example passes through joint 2, so it creates zero torque about joint 2 but it creates negative torque about joint 1; the torque about joint 1 is negative because $\mathfrak{m}_2$ gets closer to joint 1 (due to joint 2's motion). Therefore the inertia due to $\mathfrak{m}_2$ about the $\hat{z}_1$-axis is dropping, meaning that the positive momentum about joint 1 drops while joint 1's speed $\dot{theta}_1$ is constant. Therefore joint 1 must apply a negative torque, since torque is defined as the rate of change of

[2]Without this centripetal acceleration, and therefore centripetal force, the mass $\mathfrak{m}_2$ would fly off along a tangent to the circle.

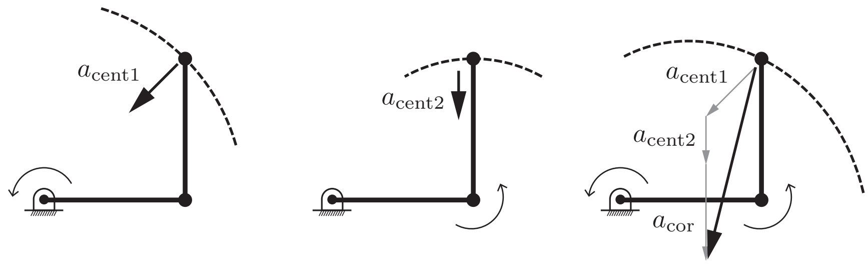


Figure 8.2: Accelerations of $\mathfrak{m}_2$ when $\theta = (0, \pi/2)$ and $\ddot{\theta} = 0$. (Left) The centripetal acceleration $a_{\text{cent1}} = (-L_1\dot{\theta}_1^2, -L_2\dot{\theta}_1^2)$ of $\mathfrak{m}_2$ when $\dot{\theta}_2 = 0$. (Middle) The centripetal acceleration $a_{\text{cent2}} = (0, -L_2\dot{\theta}_2^2)$ of $\mathfrak{m}_2$ when $\dot{\theta}_1 = 0$. (Right) When both joints are rotating with $\dot{\theta}_i > 0$, the acceleration is the vector sum of a_{cent1}, a_{cent2}, and the Coriolis acceleration $a_{\text{cor}} = (0, -2L_2\dot{\theta}_1\dot{\theta}_2)$.

angular momentum. Otherwise $\dot{\theta}_1$ would increase as $\mathfrak{m}_2$ gets closer to joint 1, just as a skater's rotation speed increases as she pulls in her outstretched arms while doing a spin.

8.1.2 General Formulation

We now describe the Lagrangian dynamics formulation for general n-link open chains. The first step is to select a set of generalized coordinates $\theta \in \mathbb{R}^n$ for the configuration space of the system. For open chains all of whose joints are actuated, it is convenient and always possible to choose θ to be the vector of the joint values. The generalized forces will be denoted $\tau \in \mathbb{R}^n$. If θ_i is a revolute joint then τ_i will correspond to a torque, while if θ_i is a prismatic joint then τ_i will correspond to a force.

Once θ has been chosen and the generalized forces τ identified, the next step is to formulate the Lagrangian $\mathcal{L}(\theta, \dot{\theta})$ as follows:

$$\mathcal{L}(\theta, \dot{\theta}) = \mathcal{K}(\theta, \dot{\theta}) - \mathcal{P}(\theta), \tag{8.13}$$

where $\mathcal{K}(\theta, \dot{\theta})$ is the kinetic energy and $\mathcal{P}(\theta)$ is the potential energy of the overall system. For rigid-link robots the kinetic energy can always be written in the form

$$\mathcal{K}(\theta) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} m_{ij}(\theta)\dot{\theta}_i\dot{\theta}_j = \frac{1}{2}\dot{\theta}^{\mathrm{T}} M(\theta)\dot{\theta}, \tag{8.14}$$

where $m_{ij}(\theta)$ is the (i, j)th element of the $n \times n$ mass matrix $M(\theta)$; a constructive proof of this assertion is provided when we examine the Newton–Euler formulation.

The dynamic equations are analytically obtained by evaluating the right-hand side of

$$\tau_i = \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i} - \frac{\partial \mathcal{L}}{\partial \theta_i}, \qquad i = 1, \ldots, n. \tag{8.15}$$

With the kinetic energy expressed as in Equation (8.14), the dynamics can be written explicitly as

$$\tau_i = \sum_{j=1}^{n} m_{ij}(\theta)\ddot{\theta}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} \Gamma_{ijk}(\theta)\dot{\theta}_j\dot{\theta}_k + \frac{\partial \mathcal{P}}{\partial \theta_i}, \qquad i = 1, \ldots, n, \tag{8.16}$$

where the $\Gamma_{ijk}(\theta)$, known as the **Christoffel symbols of the first kind**, are defined as follows:

$$\Gamma_{ijk}(\theta) = \frac{1}{2}\left(\frac{\partial m_{ij}}{\partial \theta_k} + \frac{\partial m_{ik}}{\partial \theta_j} - \frac{\partial m_{jk}}{\partial \theta_i}\right). \tag{8.17}$$

This shows that the Christoffel symbols, which generate the Coriolis and centripetal terms $c(\theta, \dot{\theta})$, are derived from the mass matrix $M(\theta)$.

As we have already seen, the equations (8.16) are often gathered together in the form

$$\tau = M(\theta)\ddot{\theta} + c(\theta, \dot{\theta}) + g(\theta) \qquad \text{or} \qquad M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}),$$

where $g(\theta)$ is simply $\partial \mathcal{P}/\partial \theta$.

We can see explicitly that the Coriolis and centripetal terms are quadratic in the velocity by using the form

$$\tau = M(\theta)\ddot{\theta} + \dot{\theta}^{\mathrm{T}}\Gamma(\theta)\dot{\theta} + g(\theta), \tag{8.18}$$

where $\Gamma(\theta)$ is an $n \times n \times n$ matrix and the product $\dot{\theta}^{\mathrm{T}}\Gamma(\theta)\dot{\theta}$ should be interpreted as follows:

$$\dot{\theta}^{\mathrm{T}}\Gamma(\theta)\dot{\theta} = \begin{bmatrix} \dot{\theta}^{\mathrm{T}}\Gamma_1(\theta)\dot{\theta} \\ \dot{\theta}^{\mathrm{T}}\Gamma_2(\theta)\dot{\theta} \\ \vdots \\ \dot{\theta}^{\mathrm{T}}\Gamma_n(\theta)\dot{\theta} \end{bmatrix},$$

where $\Gamma_i(\theta)$ is an $n \times n$ matrix with (j, k)th entry Γ_{ijk}.

It is also common to see the dynamics written as

$$\tau = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta),$$

where $C(\theta, \dot{\theta}) \in \mathbb{R}^{n\times n}$ is called the **Coriolis matrix**, with (i,j)th entry

$$c_{ij}(\theta, \dot{\theta}) = \sum_{k=1}^{n} \Gamma_{ijk}(\theta)\dot{\theta}_k. \tag{8.19}$$

The Coriolis matrix is used to prove the following **passivity property** (Proposition 8.1), which can be used to prove the stability of certain robot control laws, as we will see in Section 11.4.2.2.

Proposition 8.1. *The matrix $\dot{M}(\theta) - 2C(\theta, \dot{\theta}) \in \mathbb{R}^{n\times n}$ is skew symmetric, where $M(\theta) \in \mathbb{R}^{n\times n}$ is the mass matrix, $\dot{M}(\theta)$ its time derivative, and $C(\theta, \dot{\theta}) \in \mathbb{R}^{n\times n}$ is the Coriolis matrix as defined in Equation (8.19).*

Proof. The (i,j)th component of $\dot{M} - 2C$ is

$$\begin{aligned}
\dot{m}_{ij}(\theta) - 2c_{ij}(\theta, \dot{\theta}) &= \sum_{k=1}^{n} \frac{\partial m_{ij}}{\partial \theta_k}\dot{\theta}_k - \frac{\partial m_{ij}}{\partial \theta_k}\dot{\theta}_k - \frac{\partial m_{ik}}{\partial \theta_j}\dot{\theta}_k + \frac{\partial m_{kj}}{\partial \theta_i}\dot{\theta}_k \\
&= \sum_{k=1}^{n} \frac{\partial m_{kj}}{\partial \theta_i}\dot{\theta}_k - \frac{\partial m_{ik}}{\partial \theta_j}\dot{\theta}_k.
\end{aligned}$$

By switching the indices i and j, it can be seen that

$$\dot{m}_{ji}(\theta) - 2c_{ji}(\theta, \dot{\theta}) = -(\dot{m}_{ij}(\theta) - 2c_{ij}(\theta, \dot{\theta})),$$

thus proving that $(\dot{M} - 2C)^{\mathrm{T}} = -(\dot{M} - 2C)$ as claimed. □

8.1.3 Understanding the Mass Matrix

The kinetic energy $\frac{1}{2}\dot{\theta}^{\mathrm{T}} M(\theta)\dot{\theta}$ is a generalization of the familiar expression $\frac{1}{2}mv^{\mathrm{T}}v$ for a point mass. The fact that the mass matrix $M(\theta)$ is positive definite, meaning that $\dot{\theta}^{\mathrm{T}} M(\theta)\dot{\theta} > 0$ for all $\dot{\theta} \neq 0$, is a generalization of the fact that the mass of a point mass is always positive, $m > 0$. In both cases, if the velocity is nonzero, the kinetic energy must be positive.

On the one hand, for a point mass with dynamics expressed in Cartesian coordinates as $f = m\ddot{x}$, the mass is independent of the direction of acceleration, and the acceleration $\ddot{x}$ is always "parallel" to the force, in the sense that $\ddot{x}$ is a scalar multiple of f. A mass matrix $M(\theta)$, on the other hand, presents a different effective mass in different acceleration directions, and $\ddot{\theta}$ is not generally a scalar multiple of τ even when $\dot{\theta} = 0$. To visualize the direction dependence of the effective mass, we can map a unit ball of joint accelerations $\{\ddot{\theta} \mid \ddot{\theta}^{\mathrm{T}}\ddot{\theta} = 1\}$ through the mass matrix $M(\theta)$ to generate a joint force–torque ellipsoid when

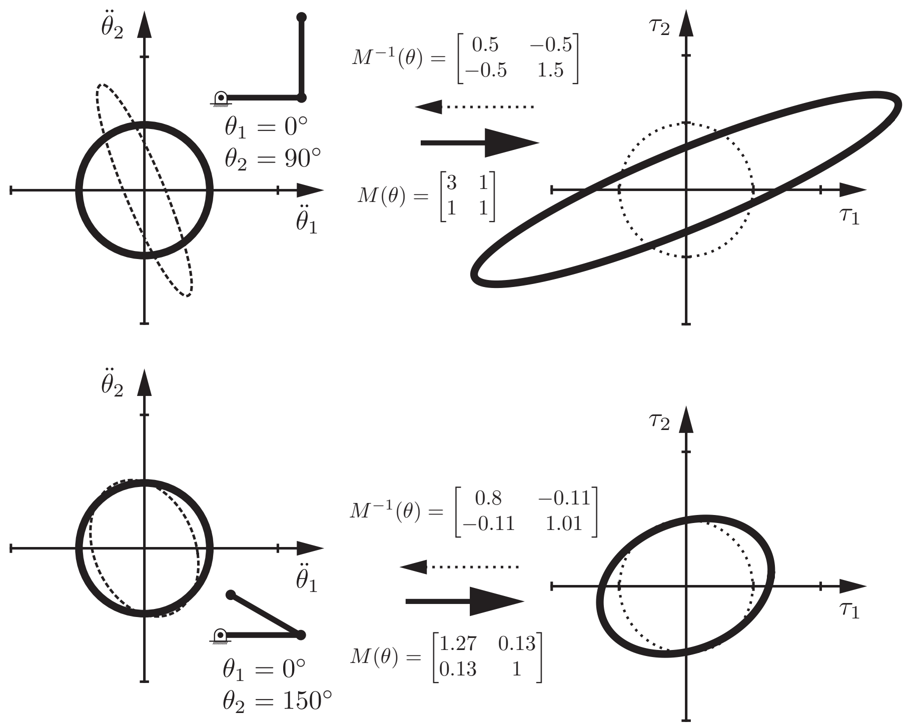


Figure 8.3: (Bold lines) A unit ball of accelerations in $\ddot{\theta}$ maps through the mass matrix $M(\theta)$ to a torque ellipsoid that depends on the configuration of the 2R arm. These torque ellipsoids may be interpreted as mass ellipsoids. The mapping is shown for two arm configurations: $(0°, 90°)$ and $(0°, 150°)$. (Dotted lines) A unit ball in τ maps through $M^{-1}(\theta)$ to an acceleration ellipsoid.

the mechanism is at rest ($\dot{\theta} = 0$). An example is shown in Figure 8.3 for the 2R arm of Figure 8.1, with $L_1 = L_2 = \mathfrak{m}_1 = \mathfrak{m}_2 = 1$, at two different joint configurations: $(\theta_1, \theta_2) = (0°, 90°)$ and $(\theta_1, \theta_2) = (0°, 150°)$. The torque ellipsoid can be interpreted as a direction-dependent mass ellipsoid: the same joint acceleration magnitude $\|\ddot{\theta}\|$ requires different joint torque magnitudes $\|\tau\|$ depending on the acceleration direction. The directions of the principal axes of the mass ellipsoid are given by the eigenvectors v_i of $M(\theta)$ and the lengths of the principal semi-axes are given by the corresponding eigenvalues λ_i. The acceleration $\ddot{\theta}$ is only a scalar multiple of τ when τ is along a principal axis of the ellipsoid.

It is easier to visualize the mass matrix if it is represented as an effective mass of the end-effector, since it is possible to feel this mass directly by grabbing and moving the end-effector. If you grabbed the endpoint of the 2R robot, depending on the direction you applied force to it, how massy would it feel? Let us denote the effective mass matrix at the end-effector as $\Lambda(\theta)$, and the velocity of the end-effector as $V = (\dot{x}, \dot{y})$. We know that the kinetic energy of the robot must be the same regardless of the coordinates we use, so

$$\frac{1}{2}\dot{\theta}^{\mathrm{T}} M(\theta)\dot{\theta} = \frac{1}{2} V^{\mathrm{T}} \Lambda(\theta) V. \tag{8.20}$$

Assuming the Jacobian $J(\theta)$ satisfying $V = J(\theta)\dot{\theta}$ is invertible, Equation (8.20) can be rewritten as follows:

$$\begin{aligned} V^{\mathrm{T}} \Lambda V &= (J^{-1}V)^{\mathrm{T}} M (J^{-1}V) \\ &= V^{\mathrm{T}} (J^{-T} M J^{-1}) V. \end{aligned}$$

In other words, the end-effector mass matrix is

$$\Lambda(\theta) = J^{-T}(\theta) M(\theta) J^{-1}(\theta). \tag{8.21}$$

Figure 8.4 shows the end-effector mass ellipsoids, with principal-axis directions given by the eigenvectors of $\Lambda(\theta)$ and principal semi-axis lengths given by its eigenvalues, for the same two 2R robot configurations as in Figure 8.3. The endpoint acceleration $(\ddot{x}, \ddot{y})$ is a scalar multiple of the force (f_x, f_y) applied at the endpoint only if the force is along a principal axis of the ellipsoid. Unless $\Lambda(\theta)$ is of the form cI, where $c > 0$ is a scalar and I is the identity matrix, the mass at the endpoint feels different from a point mass.

The change in apparent endpoint mass as a function of the configuration of the robot is an issue for robots used as haptic displays. One way to reduce the sensation of a changing mass to the user is to make the mass of the links as small as possible.

Note that the ellipsoidal interpretations of the relationship between forces and accelerations defined here are only relevant at zero velocity, where there are no Coriolis or centripetal terms.

8.1.4 Lagrangian Dynamics vs. Newton–Euler Dynamics

In the rest of this chapter, we focus on the Newton–Euler recursive method for calculating robot dynamics. Using the tools we have developed so far, the Newton–Euler formulation allows computationally efficient computer implementation, particularly for robots with many degrees of freedom, without the need

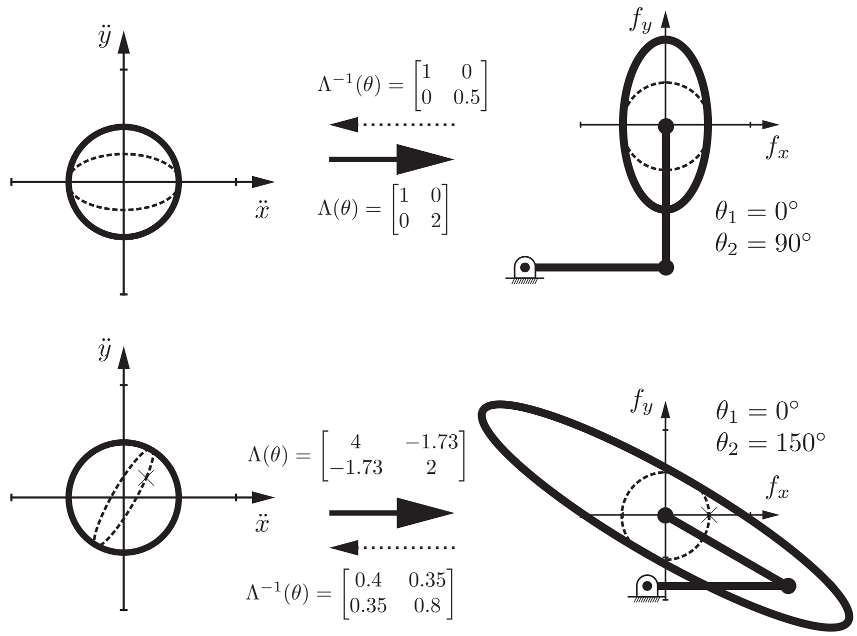


Figure 8.4: (Bold lines) A unit ball of accelerations in $(\ddot{x}, \ddot{y})$ maps through the end-effector mass matrix $\Lambda(\theta)$ to an end-effector force ellipsoid that depends on the configuration of the 2R arm. For the configuration $(\theta_1, \theta_2) = (0^\circ, 90^\circ)$, a force in the f_y-direction exactly feels both masses $\mathfrak{m}_1$ and $\mathfrak{m}_2$, while a force in the f_x-direction feels only $\mathfrak{m}_2$. (Dotted lines) A unit ball in f maps through $\Lambda^{-1}(\theta)$ to an acceleration ellipsoid. The $\times$ symbols for $(\theta_1, \theta_2) = (0^\circ, 150^\circ)$ indicate an example endpoint force $(f_x, f_y) = (1, 0)$ and its corresponding acceleration $(\ddot{x}, \ddot{y}) = (0.4, 0.35)$, showing that the force and acceleration at the endpoint are not aligned.

for differentiation. The resulting equations of motion are, and must be, identical with those derived using the energy-based Lagrangian method.

The Newton–Euler method builds on the dynamics of a single rigid body, so we begin there.

8.2 Dynamics of a Single Rigid Body

8.2.1 Classical Formulation

Consider a rigid body consisting of a number of rigidly connected point masses, where point mass i has mass $\mathfrak{m}_i$ and the total mass is $\mathfrak{m} = \sum_i \mathfrak{m}_i$. Let $r_i = (x_i, y_i, z_i)$ be the fixed location of mass i in a body frame {b}, where the origin of this frame is the unique point such that

$$\sum_i \mathfrak{m}_i r_i = 0.$$

This point is known as the **center of mass**. If some other point happens to be inconveniently chosen as the origin, then the frame {b} should be moved to the center of mass at $(1/\mathfrak{m}) \sum_i \mathfrak{m}_i r_i$ (in the inconvenient frame) and the r_i recalculated in the center-of-mass frame.

Now assume that the body is moving with a body twist $\mathcal{V}_b = (\omega_b, v_b)$, and let $p_i(t)$ be the time-varying position of $\mathfrak{m}_i$, initially located at r_i, in the inertial frame {b}. Then

$$\begin{aligned}
\dot{p}_i &= v_b + \omega_b \times p_i, \\
\ddot{p}_i &= \dot{v}_b + \frac{d}{dt}\omega_b \times p_i + \omega_b \times \frac{d}{dt} p_i \\
&= \dot{v}_b + \dot{\omega}_b \times p_i + \omega_b \times (v_b + \omega_b \times p_i).
\end{aligned}$$

Substituting r_i for p_i on the right-hand side and using our skew-symmetric notation (see Equation (3.30)), we get

$$\ddot{p}_i = \dot{v}_b + [\dot{\omega}_b] r_i + [\omega_b] v_b + [\omega_b]^2 r_i.$$

Taking as a given that $f_i = \mathfrak{m}_i \ddot{p}_i$ for a point mass, the force acting on $\mathfrak{m}_i$ is

$$f_i = \mathfrak{m}_i (\dot{v}_b + [\dot{\omega}_b] r_i + [\omega_b] v_b + [\omega_b]^2 r_i),$$

which implies a moment

$$m_i = [r_i] f_i.$$

The total force and moment acting on the body is expressed as the wrench $\mathcal{F}_b$:

$$\mathcal{F}_b = \begin{bmatrix} m_b \\ f_b \end{bmatrix} = \begin{bmatrix} \sum_i m_i \\ \sum_i f_i \end{bmatrix}.$$

To simplify the expressions for f_b and m_b, keep in mind that $\sum_i \mathfrak{m}_i r_i = 0$ (and therefore $\sum_i \mathfrak{m}_i [r_i] = 0$) and, for $a, b \in \mathbb{R}^3$, $[a] = -[a]^{\mathrm{T}}$, $[a]b = -[b]a$, and

$[a][b] = ([b][a])^{\mathrm{T}}$. Focusing on the linear dynamics,

$$\begin{aligned}
f_b &= \sum_i \mathfrak{m}_i(\dot{v}_b + [\dot{\omega}_b]r_i + [\omega_b]v_b + [\omega_b]^2 r_i) \\
&= \sum_i \mathfrak{m}_i(\dot{v}_b + [\omega_b]v_b) - \cancelto{0}{\sum_i \mathfrak{m}_i[r_i]\dot{\omega}_b} + \cancelto{0}{\sum_i \mathfrak{m}_i[r_i][\omega_b]\omega_b} \\
&= \sum_i \mathfrak{m}_i(\dot{v}_b + [\omega_b]v_b) \\
&= \mathfrak{m}(\dot{v}_b + [\omega_b]v_b).
\end{aligned} \tag{8.22}$$

The velocity product term $\mathfrak{m}[\omega_b]v_b$ arises from the fact that, for $\omega_b \neq 0$, a constant $v_b \neq 0$ corresponds to a changing linear velocity in an inertial frame.

Now focusing on the rotational dynamics,

$$\begin{aligned}
m_b &= \sum_i \mathfrak{m}_i[r_i](\dot{v}_b + [\dot{\omega}_b]r_i + [\omega_b]v_b + [\omega_b]^2 r_i) \\
&= \cancelto{0}{\sum_i \mathfrak{m}_i[r_i]\dot{v}_b} + \cancelto{0}{\sum_i \mathfrak{m}_i[r_i][\omega_b]v_b} \\
&\quad + \sum_i \mathfrak{m}_i[r_i]([\dot{\omega}_b]r_i + [\omega_b]^2 r_i) \\
&= \sum_i \mathfrak{m}_i\left(-[r_i]^2\dot{\omega}_b - [r_i]^{\mathrm{T}}[\omega_b]^{\mathrm{T}}[r_i]\omega_b\right) \\
&= \sum_i \mathfrak{m}_i\left(-[r_i]^2\dot{\omega}_b - [\omega_b][r_i]^2\omega_b\right) \\
&= \left(-\sum_i \mathfrak{m}_i[r_i]^2\right)\dot{\omega}_b + [\omega_b]\left(-\sum_i \mathfrak{m}_i[r_i]^2\right)\omega_b \\
&= \mathcal{I}_b\dot{\omega}_b + [\omega_b]\mathcal{I}_b\omega_b,
\end{aligned} \tag{8.23}$$

where $\mathcal{I}_b = -\sum_i \mathfrak{m}_i[r_i]^2 \in \mathbb{R}^{3\times 3}$ is the body's **rotational inertia matrix**. Equation (8.23) is known as **Euler's equation** for a rotating rigid body.

In Equation (8.23), note the presence of a term linear in the angular acceleration, $\mathcal{I}_b\dot{\omega}_b$, and a term quadratic in the angular velocities, $[\omega_b]\mathcal{I}_b\omega_b$, just as we saw for the mechanisms in Section 8.1. Also, $\mathcal{I}_b$ is symmetric and positive definite, just like the mass matrix for a mechanism, and the rotational kinetic energy is given by the quadratic

$$\mathcal{K} = \frac{1}{2}\omega_b^{\mathrm{T}}\mathcal{I}_b\omega_b.$$

One difference is that $\mathcal{I}_b$ is constant whereas the mass matrix $M(\theta)$ changes with the configuration of the mechanism.

Writing out the individual entries of $\mathcal{I}_b$, we get

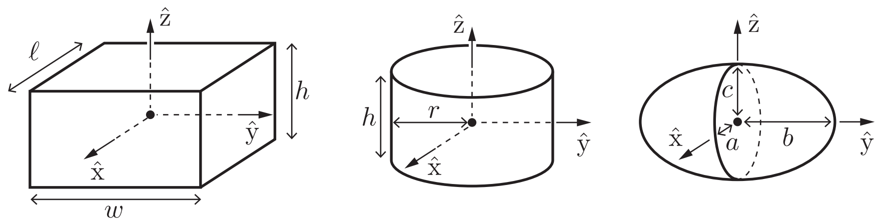
$$\begin{aligned}
\mathcal{I}_b &= \begin{bmatrix} \sum \mathfrak{m}_i(y_i^2+z_i^2) & -\sum \mathfrak{m}_i x_i y_i & -\sum \mathfrak{m}_i x_i z_i \\ -\sum \mathfrak{m}_i x_i y_i & \sum \mathfrak{m}_i(x_i^2+z_i^2) & -\sum \mathfrak{m}_i y_i z_i \\ -\sum \mathfrak{m}_i x_i z_i & -\sum \mathfrak{m}_i y_i z_i & \sum \mathfrak{m}_i(x_i^2+y_i^2) \end{bmatrix} \\
&= \begin{bmatrix} \mathcal{I}_{xx} & \mathcal{I}_{xy} & \mathcal{I}_{xz} \\ \mathcal{I}_{xy} & \mathcal{I}_{yy} & \mathcal{I}_{yz} \\ \mathcal{I}_{xz} & \mathcal{I}_{yz} & \mathcal{I}_{zz} \end{bmatrix}.
\end{aligned}$$

The summations can be replaced by volume integrals over the body $\mathcal{B}$, using the differential volume element dV, with the point masses $\mathfrak{m}_i$ replaced by a mass density function $\rho(x,y,z)$:

$$\left.\begin{aligned}
\mathcal{I}_{xx} &= \int_{\mathcal{B}} (y^2+z^2)\rho(x,y,z)\,dV \\
\mathcal{I}_{yy} &= \int_{\mathcal{B}} (x^2+z^2)\rho(x,y,z)\,dV \\
\mathcal{I}_{zz} &= \int_{\mathcal{B}} (x^2+y^2)\rho(x,y,z)\,dV \\
\mathcal{I}_{xy} &= -\int_{\mathcal{B}} xy\rho(x,y,z)\,dV \\
\mathcal{I}_{xz} &= -\int_{\mathcal{B}} xz\rho(x,y,z)\,dV \\
\mathcal{I}_{yz} &= -\int_{\mathcal{B}} yz\rho(x,y,z)\,dV.
\end{aligned}\right\} \tag{8.24}$$

If the body has uniform density, $\mathcal{I}_b$ is determined exclusively by the shape of the rigid body (see Figure 8.5).

Given an inertia matrix $\mathcal{I}_b$, the **principal axes of inertia** are given by the eigenvectors and eigenvalues of $\mathcal{I}_b$. Let v_1, v_2, v_3 be the eigenvectors of $\mathcal{I}_b$ and $\lambda_1, \lambda_2, \lambda_3$ be the corresponding eigenvalues. Then the principal axes of inertia are in the directions of v_1, v_2, v_3, and the scalar moments of inertia about these axes, the **principal moments of inertia**, are $\lambda_1, \lambda_2, \lambda_3 > 0$. One principal axis maximizes the moment of inertia among all axes passing through the center of mass, and another minimizes the moment of inertia. For bodies with symmetry, often the principal axes of inertia are apparent. They may not be unique; for a uniform-density solid sphere, for example, any three orthogonal axes intersecting at the center of mass constitute a set of principal axes, and



rectangular parallelepiped:	circular cylinder:	ellipsoid:
volume $= abc$,	volume $= \pi r^2 h$,	volume $= 4\pi abc/3$,
$\mathcal{I}_{xx} = \mathfrak{m}(w^2 + h^2)/12$,	$\mathcal{I}_{xx} = \mathfrak{m}(3r^2 + h^2)/12$,	$\mathcal{I}_{xx} = \mathfrak{m}(b^2 + c^2)/5$,
$\mathcal{I}_{yy} = \mathfrak{m}(\ell^2 + h^2)/12$,	$\mathcal{I}_{yy} = \mathfrak{m}(3r^2 + h^2)/12$,	$\mathcal{I}_{yy} = \mathfrak{m}(a^2 + c^2)/5$,
$\mathcal{I}_{zz} = \mathfrak{m}(\ell^2 + w^2)/12$	$\mathcal{I}_{zz} = \mathfrak{m}r^2/2$	$\mathcal{I}_{zz} = \mathfrak{m}(a^2 + b^2)/5$

Figure 8.5: The principal axes and the inertia about the principal axes for uniform-density bodies of mass $\mathfrak{m}$. Note that the $\hat{\mathrm{x}}$ and $\hat{\mathrm{y}}$ principal axes of the cylinder are not unique.

the minimum principal moment of inertia is equal to the maximum principal moment of inertia.

If the principal axes of inertia are aligned with the axes of {b}, the off-diagonal terms of $\mathcal{I}_b$ are all zero, and the eigenvalues are the scalar moments of inertia $\mathcal{I}_{xx}$, $\mathcal{I}_{yy}$, and $\mathcal{I}_{zz}$ about the $\hat{\mathrm{x}}$-, $\hat{\mathrm{y}}$-, and $\hat{\mathrm{z}}$-axes, respectively. In this case, the equations of motion (8.23) simplify to

$$m_b = \begin{bmatrix} \mathcal{I}_{xx}\dot{\omega}_x + (\mathcal{I}_{zz} - \mathcal{I}_{yy})\omega_y\omega_z \\ \mathcal{I}_{yy}\dot{\omega}_y + (\mathcal{I}_{xx} - \mathcal{I}_{zz})\omega_x\omega_z \\ \mathcal{I}_{zz}\dot{\omega}_z + (\mathcal{I}_{yy} - \mathcal{I}_{xx})\omega_x\omega_y \end{bmatrix}, \tag{8.25}$$

where $\omega_b = (\omega_x, \omega_y, \omega_z)$. When possible, we choose the axes of {b} to be aligned with the principal axes of inertia, in order to reduce the number of nonzero entries in $\mathcal{I}_b$ and to simplify the equations of motion.

Examples of common uniform-density solid bodies, their principal axes of inertia, and the principal moments of inertia obtained by solving the integrals (8.24), are given in Figure 8.5.

An inertia matrix $\mathcal{I}_b$ can be expressed in a rotated frame {c} described by the rotation matrix R_{bc}. Denoting this inertia matrix as $\mathcal{I}_c$, and knowing that the kinetic energy of the rotating body is independent of the chosen frame, we have

$$\begin{aligned}\frac{1}{2}\omega_c^{\mathrm{T}}\mathcal{I}_c\omega_c &= \frac{1}{2}\omega_b^{\mathrm{T}}\mathcal{I}_b\omega_b \\ &= \frac{1}{2}(R_{bc}\omega_c)^{\mathrm{T}}\mathcal{I}_b(R_{bc}\omega_c) \\ &= \frac{1}{2}\omega_c^{\mathrm{T}}(R_{bc}^{\mathrm{T}}\mathcal{I}_b R_{bc})\omega_c.\end{aligned}$$

In other words,

$$\mathcal{I}_c = R_{bc}^{\mathrm{T}}\mathcal{I}_b R_{bc}. \tag{8.26}$$

If the axes of {b} are not aligned with the principal axes of inertia then we can diagonalize the inertia matrix by expressing it instead in the rotated frame {c}, where the columns of R_{bc} correspond to the eigenvalues of $\mathcal{I}_b$.

Sometimes it is convenient to represent the inertia matrix in a frame at a point not at the center of mass of the body, for example at a joint. **Steiner's theorem** can be stated as follows.

Theorem 8.2. *The inertia matrix $\mathcal{I}_q$ about a frame aligned with* {b}*, but at a point $q = (q_x, q_y, q_z)$ in* {b}*, is related to the inertia matrix $\mathcal{I}_b$ calculated at the center of mass by*

$$\mathcal{I}_q = \mathcal{I}_b + \mathfrak{m}(q^{\mathrm{T}}qI - qq^{\mathrm{T}}), \tag{8.27}$$

where I is the 3×3 identity matrix and $\mathfrak{m}$ is the mass of the body.

Steiner's theorem is a more general statement of the parallel-axis theorem, which states that the scalar inertia $\mathcal{I}_d$ about an axis parallel to, but a distance d from, an axis through the center of mass is related to the scalar inertia $\mathcal{I}_{\mathrm{cm}}$ about the axis through the center of mass by

$$\mathcal{I}_d = \mathcal{I}_{\mathrm{cm}} + \mathfrak{m}d^2. \tag{8.28}$$

Equations (8.26) and (8.27) are useful for calculating the inertia of a rigid body consisting of component rigid bodies. First we calculate the inertia matrices of the n component bodies in terms of frames at their individual centers of mass. Then we choose a common frame {common} (e.g., at the center of mass of the composite rigid body) and use Equations (8.26) and (8.27) to express each inertia matrix in this common frame. Once the individual inertia matrices are expressed in {common}, they can be summed to get the inertia matrix $\mathcal{I}_{\mathrm{common}}$ for the composite rigid body.

In the case of motion confined to the x̂–ŷ-plane, where $\omega_b = (0, 0, \omega_z)$ and the inertia of the body about the ẑ-axis through the center of mass is given by the scalar $\mathcal{I}_{zz}$, the spatial rotational dynamics (8.23) reduces to the planar rotational dynamics

$$m_z = \mathcal{I}_{zz}\dot{\omega}_z,$$

and the rotational kinetic energy is

$$\mathcal{K} = \frac{1}{2}\mathcal{I}_{zz}\omega_z^2.$$

8.2.2 Twist–Wrench Formulation

The linear dynamics (8.22) and the rotational dynamics (8.23) can be written in the following combined form:

$$\begin{bmatrix} m_b \\ f_b \end{bmatrix} = \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix} \begin{bmatrix} \dot{\omega}_b \\ \dot{v}_b \end{bmatrix} + \begin{bmatrix} [\omega_b] & 0 \\ 0 & [\omega_b] \end{bmatrix} \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix} \begin{bmatrix} \omega_b \\ v_b \end{bmatrix}, \quad (8.29)$$

where I is the 3×3 identity matrix. With the benefit of hindsight, and also making use of the fact that $[v]v = v \times v = 0$ and $[v]^{\mathrm{T}} = -[v]$, we can write Equation (8.29) in the following equivalent form:

$$\begin{aligned} \begin{bmatrix} m_b \\ f_b \end{bmatrix} &= \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix} \begin{bmatrix} \dot{\omega}_b \\ \dot{v}_b \end{bmatrix} + \begin{bmatrix} [\omega_b] & [v_b] \\ 0 & [\omega_b] \end{bmatrix} \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix} \begin{bmatrix} \omega_b \\ v_b \end{bmatrix} \\ &= \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix} \begin{bmatrix} \dot{\omega}_b \\ \dot{v}_b \end{bmatrix} - \begin{bmatrix} [\omega_b] & 0 \\ [v_b] & [\omega_b] \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix} \begin{bmatrix} \omega_b \\ v_b \end{bmatrix}. \end{aligned} \quad (8.30)$$

Written this way, each term can now be identified with six-dimensional spatial quantities as follows:

(a) The vectors (ω_b, v_b) and (m_b, f_b) can be respectively identified with the body twist $\mathcal{V}_b$ and body wrench $\mathcal{F}_b$,

$$\mathcal{V}_b = \begin{bmatrix} \omega_b \\ v_b \end{bmatrix}, \qquad \mathcal{F}_b = \begin{bmatrix} m_b \\ f_b \end{bmatrix}. \quad (8.31)$$

(b) The **spatial inertia matrix** $\mathcal{G}_b \in \mathbb{R}^{6\times 6}$ is defined as

$$\mathcal{G}_b = \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix}. \quad (8.32)$$

As an aside, the kinetic energy of the rigid body can be expressed in terms of the spatial inertia matrix as

$$\text{kinetic energy} = \frac{1}{2}\omega_b^{\mathrm{T}}\mathcal{I}_b\omega_b + \frac{1}{2}\mathfrak{m}v_b^{\mathrm{T}}v_b = \frac{1}{2}\mathcal{V}_b^{\mathrm{T}}\mathcal{G}_b\mathcal{V}_b. \quad (8.33)$$

(c) The **spatial momentum** $\mathcal{P}_b \in \mathbb{R}^6$ is defined as

$$\mathcal{P}_b = \begin{bmatrix} \mathcal{I}_b \omega_b \\ \mathfrak{m} v_b \end{bmatrix} = \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m} I \end{bmatrix} \begin{bmatrix} \omega_b \\ v_b \end{bmatrix} = \mathcal{G}_b \mathcal{V}_b. \tag{8.34}$$

Observe that the term involving $\mathcal{P}_b$ in Equation (8.30) is left-multiplied by the matrix

$$-\begin{bmatrix} [\omega_b] & 0 \\ [v_b] & [\omega_b] \end{bmatrix}^{\mathrm{T}}. \tag{8.35}$$

We now explain the origin and geometric significance of this matrix. First, recall that the cross product of two vectors $\omega_1, \omega_2 \in \mathbb{R}^3$ can be calculated, using the skew-symmetric matrix notation, as follows:

$$[\omega_1 \times \omega_2] = [\omega_1][\omega_2] - [\omega_2][\omega_1]. \tag{8.36}$$

The matrix in (8.35) can be thought of as a generalization of the cross-product operation to six-dimensional twists. Specifically, given two twists $\mathcal{V}_1 = (\omega_1, v_1)$ and $\mathcal{V}_2 = (\omega_2, v_2)$, we perform a calculation analogous to (8.36):

$$\begin{aligned} [\mathcal{V}_1][\mathcal{V}_2] - [\mathcal{V}_2][\mathcal{V}_1] &= \begin{bmatrix} [\omega_1] & v_1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} [\omega_2] & v_2 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} [\omega_2] & v_2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} [\omega_1] & v_1 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} [\omega_1][\omega_2] - [\omega_2][\omega_1] & [\omega_1]v_2 - [\omega_2]v_1 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} [\omega'] & v' \\ 0 & 0 \end{bmatrix}, \end{aligned}$$

which can be written more compactly in vector form as

$$\begin{bmatrix} \omega' \\ v' \end{bmatrix} = \begin{bmatrix} [\omega_1] & 0 \\ [v_1] & [\omega_1] \end{bmatrix} \begin{bmatrix} \omega_2 \\ v_2 \end{bmatrix}.$$

This generalization of the cross product to two twists $\mathcal{V}_1$ and $\mathcal{V}_2$ is called the **Lie bracket** of $\mathcal{V}_1$ and $\mathcal{V}_2$.

Definition 8.3. Given two twists $\mathcal{V}_1 = (\omega_1, v_1)$ and $\mathcal{V}_2 = (\omega_2, v_2)$, the **Lie bracket** of $\mathcal{V}_1$ and $\mathcal{V}_2$, written either as $[\mathrm{ad}_{\mathcal{V}_1}]\mathcal{V}_2$ or $\mathrm{ad}_{\mathcal{V}_1}(\mathcal{V}_2)$, is defined as follows:

$$\begin{bmatrix} [\omega_1] & 0 \\ [v_1] & [\omega_1] \end{bmatrix} \begin{bmatrix} \omega_2 \\ v_2 \end{bmatrix} = [\mathrm{ad}_{\mathcal{V}_1}]\mathcal{V}_2 = \mathrm{ad}_{\mathcal{V}_1}(\mathcal{V}_2) \in \mathbb{R}^6, \tag{8.37}$$

where

$$[\mathrm{ad}_{\mathcal{V}}] = \begin{bmatrix} [\omega] & 0 \\ [v] & [\omega] \end{bmatrix} \in \mathbb{R}^{6\times 6}. \tag{8.38}$$

Definition 8.4. Given a twist $\mathcal{V} = (\omega, v)$ and a wrench $\mathcal{F} = (m, f)$, define the mapping

$$\mathrm{ad}_{\mathcal{V}}^{\mathrm{T}}(\mathcal{F}) = [\mathrm{ad}_{\mathcal{V}}]^{\mathrm{T}}\mathcal{F} = \begin{bmatrix} [\omega] & 0 \\ [v] & [\omega] \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} m \\ f \end{bmatrix} = \begin{bmatrix} -[\omega]m - [v]f \\ -[\omega]f \end{bmatrix}. \tag{8.39}$$

Using the notation and definitions above, the dynamic equations for a single rigid body can now be written as

$$\begin{aligned} \mathcal{F}_b &= \mathcal{G}_b\dot{\mathcal{V}}_b - \mathrm{ad}_{\mathcal{V}_b}^{\mathrm{T}}(\mathcal{P}_b) \\ &= \mathcal{G}_b\dot{\mathcal{V}}_b - [\mathrm{ad}_{\mathcal{V}_b}]^{\mathrm{T}}\mathcal{G}_b\mathcal{V}_b. \end{aligned} \tag{8.40}$$

Note the analogy between Equation (8.40) and the moment equation for a rotating rigid body:

$$m_b = \mathcal{I}_b\dot{\omega}_b - [\omega_b]^{\mathrm{T}}\mathcal{I}_b\omega_b. \tag{8.41}$$

Equation (8.41) is simply the rotational component of (8.40).

8.2.3 Dynamics in Other Frames

The derivation of the dynamic equations (8.40) relies on the use of a center-of-mass frame {b}. It is straightforward to express the dynamics in other frames, however. Let's call one such frame {a}.

Since the kinetic energy of the rigid body must be independent of the frame of representation,

$$\begin{aligned} \frac{1}{2}\mathcal{V}_a^{\mathrm{T}}\mathcal{G}_a\mathcal{V}_a &= \frac{1}{2}\mathcal{V}_b^{\mathrm{T}}\mathcal{G}_b\mathcal{V}_b \\ &= \frac{1}{2}([\mathrm{Ad}_{T_{ba}}]\mathcal{V}_a)^{\mathrm{T}}\mathcal{G}_b[\mathrm{Ad}_{T_{ba}}]\mathcal{V}_a \\ &= \frac{1}{2}\mathcal{V}_a^{\mathrm{T}}\underbrace{[\mathrm{Ad}_{T_{ba}}]^{\mathrm{T}}\mathcal{G}_b[\mathrm{Ad}_{T_{ba}}]}_{\mathcal{G}_a}\mathcal{V}_a; \end{aligned}$$

for the adjoint representation Ad (see Definition 3.20). In other words, the spatial inertia matrix $\mathcal{G}_a$ in {a} is related to $\mathcal{G}_b$ by

$$\mathcal{G}_a = [\mathrm{Ad}_{T_{ba}}]^{\mathrm{T}}\mathcal{G}_b[\mathrm{Ad}_{T_{ba}}]. \tag{8.42}$$

This is a generalization of Steiner's theorem.

Using the spatial inertia matrix $\mathcal{G}_a$, the equations of motion (8.40) in the {b} frame can be expressed equivalently in the {a} frame as

$$\mathcal{F}_a = \mathcal{G}_a\dot{\mathcal{V}}_a - [\mathrm{ad}_{\mathcal{V}_a}]^{\mathrm{T}}\mathcal{G}_a\mathcal{V}_a, \tag{8.43}$$

where $\mathcal{F}_a$ and $\mathcal{V}_a$ are the wrench and twist written in the {a} frame. (See Exercise 8.3.) Thus the form of the equations of motion is independent of the frame of representation.

8.3 Newton–Euler Inverse Dynamics

We now consider the inverse dynamics problem for an n-link open chain connected by one-dof joints. Given the joint positions $\theta \in \mathbb{R}^n$, velocities $\dot{\theta} \in \mathbb{R}^n$, and accelerations $\ddot{\theta} \in \mathbb{R}^n$, the objective is to calculate the right-hand side of the dynamics equation

$$\tau = M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}).$$

The main result is a recursive inverse dynamics algorithm consisting of a forward and a backward iteration stage. In the former, the positions, velocities, and accelerations of each link are propagated from the base to the tip while in the backward iterations the forces and moments experienced by each link are propagated from the tip to the base.

8.3.1 Derivation

A body-fixed reference frame $\{i\}$ is attached to the center of mass of each link i, $i = 1, \ldots, n$. The base frame is denoted $\{0\}$, and a frame at the end-effector is denoted $\{n+1\}$. This frame is fixed in $\{n\}$.

When the manipulator is at the home position, with all joint variables zero, we denote the configuration of frame $\{j\}$ in $\{i\}$ as $M_{i,j} \in SE(3)$, and the configuration of $\{i\}$ in the base frame $\{0\}$ using the shorthand $M_i = M_{0,i}$. With these definitions, $M_{i-1,i}$ and $M_{i,i-1}$ can be calculated as

$$M_{i-1,i} = M_{i-1}^{-1} M_i \qquad \text{and} \qquad M_{i,i-1} = M_i^{-1} M_{i-1}.$$

The screw axis for joint i, expressed in the link frame $\{i\}$, is $\mathcal{A}_i$. This same screw axis is expressed in the space frame $\{0\}$ as $\mathcal{S}_i$, where the two are related by

$$\mathcal{A}_i = \mathrm{Ad}_{M_i^{-1}}(\mathcal{S}_i).$$

Defining $T_{i,j} \in SE(3)$ to be the configuration of frame $\{j\}$ in $\{i\}$ for arbitrary joint variables θ then $T_{i-1,i}(\theta_i)$, the configuration of $\{i\}$ relative to $\{i-1\}$ given the joint variable θ_i, and $T_{i,i-1}(\theta_i) = T_{i-1,i}^{-1}(\theta_i)$ are calculated as

$$T_{i-1,i}(\theta_i) = M_{i-1,i} e^{[\mathcal{A}_i]\theta_i} \qquad \text{and} \qquad T_{i,i-1}(\theta_i) = e^{-[\mathcal{A}_i]\theta_i} M_{i,i-1}.$$

We further adopt the following notation:

(a) The twist of link frame $\{i\}$, expressed in frame-$\{i\}$ coordinates, is denoted $\mathcal{V}_i = (\omega_i, v_i)$.

(b) The wrench transmitted through joint i to link frame $\{i\}$, expressed in frame-$\{i\}$ coordinates, is denoted $\mathcal{F}_i = (m_i, f_i)$.

(c) Let $\mathcal{G}_i \in \mathbb{R}^{6\times 6}$ denote the spatial inertia matrix of link i, expressed relative to link frame $\{i\}$. Since we are assuming that all link frames are situated at the link center of mass, $\mathcal{G}_i$ has the block-diagonal form

$$\mathcal{G}_i = \begin{bmatrix} \mathcal{I}_i & 0 \\ 0 & \mathfrak{m}_i I \end{bmatrix}, \tag{8.44}$$

where $\mathcal{I}_i$ denotes the 3×3 rotational inertia matrix of link i and $\mathfrak{m}_i$ is the link mass.

With these definitions, we can recursively calculate the twist and acceleration of each link, moving from the base to the tip. The twist $\mathcal{V}_i$ of link i is the sum of the twist of link $i-1$, but expressed in $\{i\}$, and the added twist due to the joint rate $\dot{\theta}_i$:

$$\mathcal{V}_i = \mathcal{A}_i \dot{\theta}_i + [\mathrm{Ad}_{T_{i,i-1}}]\mathcal{V}_{i-1}. \tag{8.45}$$

The accelerations $\dot{\mathcal{V}}_i$ can also be found recursively. Taking the time derivative of Equation (8.45), we get

$$\dot{\mathcal{V}}_i = \mathcal{A}_i \ddot{\theta}_i + [\mathrm{Ad}_{T_{i,i-1}}]\dot{\mathcal{V}}_{i-1} + \frac{d}{dt}\left([\mathrm{Ad}_{T_{i,i-1}}]\right)\mathcal{V}_{i-1}. \tag{8.46}$$

To calculate the final term in this equation, we express $T_{i,i-1}$ and $\mathcal{A}_i$ as

$$T_{i,i-1} = \begin{bmatrix} R_{i,i-1} & p \\ 0 & 1 \end{bmatrix} \qquad \text{and} \qquad \mathcal{A}_i = \begin{bmatrix} \omega \\ v \end{bmatrix}.$$

Then

$$\begin{aligned} \frac{d}{dt}\left([\mathrm{Ad}_{T_{i,i-1}}]\right)\mathcal{V}_{i-1} &= \frac{d}{dt}\left(\begin{bmatrix} R_{i,i-1} & 0 \\ [p]R_{i,i-1} & R_{i,i-1} \end{bmatrix}\right)\mathcal{V}_{i-1} \\ &= \begin{bmatrix} -[\omega\dot{\theta}_i]R_{i,i-1} & 0 \\ -[v\dot{\theta}_i]R_{i,i-1} - [\omega\dot{\theta}_i][p]R_{i,i-1} & -[\omega\dot{\theta}_i]R_{i,i-1} \end{bmatrix}\mathcal{V}_{i-1} \\ &= \underbrace{\begin{bmatrix} -[\omega\dot{\theta}_i] & 0 \\ -[v\dot{\theta}_i] & -[\omega\dot{\theta}_i] \end{bmatrix}}_{-[\mathrm{ad}_{\mathcal{A}_i\dot{\theta}_i}]}\underbrace{\begin{bmatrix} R_{i,i-1} & 0 \\ [p]R_{i,i-1} & R_{i,i-1} \end{bmatrix}}_{[\mathrm{Ad}_{T_{i,i-1}}]}\mathcal{V}_{i-1} \\ &= -[\mathrm{ad}_{\mathcal{A}_i\dot{\theta}_i}]\mathcal{V}_i \\ &= [\mathrm{ad}_{\mathcal{V}_i}]\mathcal{A}_i\dot{\theta}_i. \end{aligned}$$

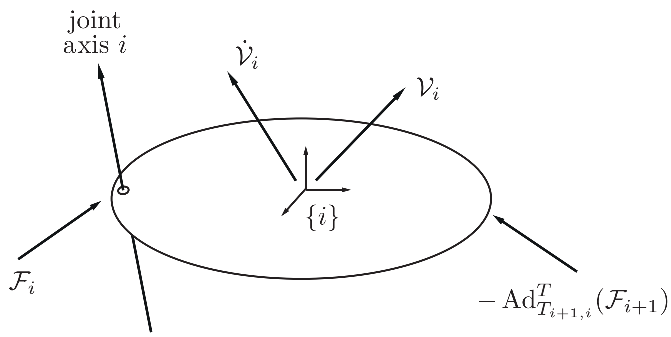


Figure 8.6: Free-body diagram illustrating the moments and forces exerted on link i.

Substituting this result into Equation (8.46), we get

$$\dot{\mathcal{V}}_i = \mathcal{A}_i\ddot{\theta}_i + [\mathrm{Ad}_{T_{i,i-1}}]\dot{\mathcal{V}}_{i-1} + [\mathrm{ad}_{\mathcal{V}_i}]\mathcal{A}_i\dot{\theta}_i, \tag{8.47}$$

i.e., the acceleration of link i is the sum of three components: a component due to the joint acceleration $\ddot{\theta}_i$, a component due to the acceleration of link $i-1$ expressed in $\{i\}$, and a velocity-product component.

Once we have determined all the link twists and accelerations moving outward from the base, we can calculate the joint torques or forces by moving inward from the tip. The rigid-body dynamics (8.40) tells us the total wrench that acts on link i given $\mathcal{V}_i$ and $\dot{\mathcal{V}}_i$. Furthermore, the total wrench acting on link i is the sum of the wrench $\mathcal{F}_i$ transmitted through joint i and the wrench applied to the link through joint $i+1$ (or, for link n, the wrench applied to the link by the environment at the end-effector frame $\{n+1\}$), expressed in the frame i. Therefore, we have the equality

$$\mathcal{G}_i\dot{\mathcal{V}}_i - \mathrm{ad}^{\mathrm{T}}_{\mathcal{V}_i}(\mathcal{G}_i\mathcal{V}_i) = \mathcal{F}_i - \mathrm{Ad}^{\mathrm{T}}_{T_{i+1,i}}(\mathcal{F}_{i+1}); \tag{8.48}$$

see Figure 8.6. Solving from the tip toward the base, at each link i we solve for the only unknown in Equation (8.48): $\mathcal{F}_i$. Since joint i has only one degree of freedom, five dimensions of the six-vector $\mathcal{F}_i$ are provided "for free" by the structure of the joint, and the actuator only has to provide the scalar force or torque in the direction of the joint's screw axis:

$$\tau_i = \mathcal{F}_i^{\mathrm{T}}\mathcal{A}_i. \tag{8.49}$$

Equation (8.49) provides the torques required at each joint, solving the inverse dynamics problem.

8.3.2 Newton-Euler Inverse Dynamics Algorithm

Initialization Attach a frame $\{0\}$ to the base, frames $\{1\}$ to $\{n\}$ to the centers of mass of links $\{1\}$ to $\{n\}$, and a frame $\{n+1\}$ at the end-effector, fixed in the frame $\{n\}$. Define $M_{i,i-1}$ to be the configuration of $\{i-1\}$ in $\{i\}$ when $\theta_i = 0$. Let $\mathcal{A}_i$ be the screw axis of joint i expressed in $\{i\}$, and $\mathcal{G}_i$ be the 6×6 spatial inertia matrix of link i. Define $\mathcal{V}_0$ to be the twist of the base frame $\{0\}$ expressed in $\{0\}$ coordinates. (This quantity is typically zero.) Let $\mathfrak{g} \in \mathbb{R}^3$ be the gravity vector expressed in base-frame coordinates, and define $\dot{\mathcal{V}}_0 = (\dot\omega_0, \dot v_0) = (0, -\mathfrak{g})$. (Gravity is treated as an acceleration of the base in the opposite direction.) Define $\mathcal{F}_{n+1} = \mathcal{F}_{\text{tip}} = (m_{\text{tip}}, f_{\text{tip}})$ to be the wrench applied to the environment by the end-effector, expressed in the end-effector frame $\{n+1\}$.

Forward iterations Given $\theta, \dot\theta, \ddot\theta$, for $i = 1$ to n do

$$\begin{aligned} T_{i,i-1} &= e^{-[\mathcal{A}_i]\theta_i} M_{i,i-1}, && (8.50)\\ \mathcal{V}_i &= \mathrm{Ad}_{T_{i,i-1}}(\mathcal{V}_{i-1}) + \mathcal{A}_i\dot\theta_i, && (8.51)\\ \dot{\mathcal{V}}_i &= \mathrm{Ad}_{T_{i,i-1}}(\dot{\mathcal{V}}_{i-1}) + \mathrm{ad}_{\mathcal{V}_i}(\mathcal{A}_i)\dot\theta_i + \mathcal{A}_i\ddot\theta_i. && (8.52) \end{aligned}$$

Backward iterations For $i = n$ to 1 do

$$\begin{aligned} \mathcal{F}_i &= \mathrm{Ad}^{\mathrm{T}}_{T_{i+1,i}}(\mathcal{F}_{i+1}) + \mathcal{G}_i\dot{\mathcal{V}}_i - \mathrm{ad}^{\mathrm{T}}_{\mathcal{V}_i}(\mathcal{G}_i\mathcal{V}_i), && (8.53)\\ \tau_i &= \mathcal{F}_i^{\mathrm{T}}\mathcal{A}_i. && (8.54) \end{aligned}$$

8.4 Dynamic Equations in Closed Form

In this section we show how the equations in the recursive inverse dynamics algorithm can be organized into a closed-form set of dynamics equations $\tau = M(\theta)\ddot\theta + c(\theta,\dot\theta) + g(\theta)$.

Before doing so, we prove our earlier assertion that the total kinetic energy $\mathcal{K}$ of the robot can be expressed as $\mathcal{K} = \frac{1}{2}\dot\theta^{\mathrm{T}}M(\theta)\dot\theta$. We do so by noting that $\mathcal{K}$ can be expressed as the sum of the kinetic energies of each link:

$$\mathcal{K} = \frac{1}{2}\sum_{i=1}^{n} \mathcal{V}_i^{\mathrm{T}}\mathcal{G}_i\mathcal{V}_i, \tag{8.55}$$

where $\mathcal{V}_i$ is the twist of link frame $\{i\}$ and $\mathcal{G}_i$ is the spatial inertia matrix of link i as defined by Equation (8.32) (both are expressed in link-frame-$\{i\}$ coordinates). Let $T_{0i}(\theta_1, \ldots, \theta_i)$ denote the forward kinematics from the base frame $\{0\}$ to link frame $\{i\}$, and let $J_{ib}(\theta)$ denote the body Jacobian obtained from $T_{0i}^{-1}\dot{T}_{0i}$. Note that J_{ib} as defined is a $6 \times i$ matrix; we turn it into a $6 \times n$ matrix by filling in all entries of the last $n - i$ columns with zeros. With this definition of J_{ib}, we can write

$$\mathcal{V}_i = J_{ib}(\theta)\dot{\theta}, \qquad i = 1, \ldots, n.$$

The kinetic energy can then be written

$$\mathcal{K} = \frac{1}{2}\dot{\theta}^{\mathrm{T}} \left(\sum_{i=1}^{n} J_{ib}^{\mathrm{T}}(\theta)\mathcal{G}_i J_{ib}(\theta) \right) \dot{\theta}. \tag{8.56}$$

The term inside the parentheses is precisely the mass matrix $M(\theta)$:

$$M(\theta) = \sum_{i=1}^{n} J_{ib}^{\mathrm{T}}(\theta)\mathcal{G}_i J_{ib}(\theta). \tag{8.57}$$

We now return to the original task of deriving a closed-form set of dynamic equations. We start by defining the following stacked vectors:

$$\mathcal{V} = \begin{bmatrix} \mathcal{V}_1 \\ \vdots \\ \mathcal{V}_n \end{bmatrix} \in \mathbb{R}^{6n}, \tag{8.58}$$

$$\mathcal{F} = \begin{bmatrix} \mathcal{F}_1 \\ \vdots \\ \mathcal{F}_n \end{bmatrix} \in \mathbb{R}^{6n}. \tag{8.59}$$

Further, define the following matrices:

$$\mathcal{A} = \begin{bmatrix} \mathcal{A}_1 & 0 & \cdots & 0 \\ 0 & \mathcal{A}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \mathcal{A}_n \end{bmatrix} \in \mathbb{R}^{6n \times n}, \tag{8.60}$$

$$\mathcal{G} = \begin{bmatrix} \mathcal{G}_1 & 0 & \cdots & 0 \\ 0 & \mathcal{G}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \mathcal{G}_n \end{bmatrix} \in \mathbb{R}^{6n \times 6n}, \tag{8.61}$$

$$[\mathrm{ad}_{\mathcal{V}}] = \begin{bmatrix} [\mathrm{ad}_{\mathcal{V}_1}] & 0 & \cdots & 0 \\ 0 & [\mathrm{ad}_{\mathcal{V}_2}] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & [\mathrm{ad}_{\mathcal{V}_n}] \end{bmatrix} \in \mathbb{R}^{6n \times 6n}, \tag{8.62}$$

$$\left[\mathrm{ad}_{\mathcal{A}\dot{\theta}}\right] = \begin{bmatrix} [\mathrm{ad}_{\mathcal{A}_1\dot{\theta}_1}] & 0 & \cdots & 0 \\ 0 & [\mathrm{ad}_{\mathcal{A}_2\dot{\theta}_2}] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & [\mathrm{ad}_{\mathcal{A}_n\dot{\theta}_n}] \end{bmatrix} \in \mathbb{R}^{6n \times 6n}, \tag{8.63}$$

$$\mathcal{W}(\theta) = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ [\mathrm{Ad}_{T_{21}}] & 0 & \cdots & 0 & 0 \\ 0 & [\mathrm{Ad}_{T_{32}}] & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \left[\mathrm{Ad}_{T_{n,n-1}}\right] & 0 \end{bmatrix} \in \mathbb{R}^{6n \times 6n}. \tag{8.64}$$

We write $\mathcal{W}(\theta)$ to emphasize the dependence of $\mathcal{W}$ on θ. Finally, define the following stacked vectors:

$$\mathcal{V}_{\text{base}} = \begin{bmatrix} \text{Ad}_{T_{10}}(\mathcal{V}_0) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^{6n}, \tag{8.65}$$

$$\dot{\mathcal{V}}_{\text{base}} = \begin{bmatrix} \text{Ad}_{T_{10}}(\dot{\mathcal{V}}_0) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^{6n}, \tag{8.66}$$

$$\mathcal{F}_{\text{tip}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \text{Ad}^{\text{T}}_{T_{n+1,n}}(\mathcal{F}_{n+1}) \end{bmatrix} \in \mathbb{R}^{6n}. \tag{8.67}$$

Note that $\mathcal{A} \in \mathbb{R}^{6n\times n}$ and $\mathcal{G} \in \mathbb{R}^{6n\times 6n}$ are constant block-diagonal matrices, in which $\mathcal{A}$ contains only the kinematic parameters while $\mathcal{G}$ contains only the mass and inertial parameters for each link.

With the above definitions, our earlier recursive inverse dynamics algorithm can be assembled into the following set of matrix equations:

$$\mathcal{V} = \mathcal{W}(\theta)\mathcal{V} + \mathcal{A}\dot{\theta} + \mathcal{V}_{\text{base}}, \tag{8.68}$$

$$\dot{\mathcal{V}} = \mathcal{W}(\theta)\dot{\mathcal{V}} + \mathcal{A}\ddot{\theta} - [\text{ad}_{\mathcal{A}\dot{\theta}}](\mathcal{W}(\theta)\mathcal{V} + \mathcal{V}_{\text{base}}) + \dot{\mathcal{V}}_{\text{base}}, \tag{8.69}$$

$$\mathcal{F} = \mathcal{W}^{\text{T}}(\theta)\mathcal{F} + \mathcal{G}\dot{\mathcal{V}} - [\text{ad}_{\mathcal{V}}]^{\text{T}}\mathcal{G}\mathcal{V} + \mathcal{F}_{\text{tip}}, \tag{8.70}$$

$$\tau = \mathcal{A}^{\text{T}}\mathcal{F}. \tag{8.71}$$

The matrix $\mathcal{W}(\theta)$ has the property that $\mathcal{W}^n(\theta) = 0$ (such a matrix is said to be nilpotent of order n), and one consequence verifiable through direct calculation is that $(I - \mathcal{W}(\theta))^{-1} = I + \mathcal{W}(\theta) + \cdots + \mathcal{W}^{n-1}(\theta)$. Defining $\mathcal{L}(\theta) = (I - \mathcal{W}(\theta))^{-1}$, it can further be verified via direct calculation that

$$\mathcal{L}(\theta) = \begin{bmatrix} I & 0 & 0 & \cdots & 0 \\ [\text{Ad}_{T_{21}}] & I & 0 & \cdots & 0 \\ [\text{Ad}_{T_{31}}] & [\text{Ad}_{T_{32}}] & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ [\text{Ad}_{T_{n1}}] & [\text{Ad}_{T_{n2}}] & [\text{Ad}_{T_{n3}}] & \cdots & I \end{bmatrix} \in \mathbb{R}^{6n\times 6n}. \tag{8.72}$$

We write $\mathcal{L}(\theta)$ to emphasize the dependence of $\mathcal{L}$ on θ. The earlier matrix equations can now be reorganized as follows:

$$\begin{aligned}
\mathcal{V} &= \mathcal{L}(\theta)\left(\mathcal{A}\dot{\theta} + \mathcal{V}_{\rm base}\right), && (8.73)\\
\dot{\mathcal{V}} &= \mathcal{L}(\theta)\left(\mathcal{A}\ddot{\theta} + [\mathrm{ad}_{\mathcal{A}\dot{\theta}}]\mathcal{W}(\theta)\mathcal{V} + [\mathrm{ad}_{\mathcal{A}\dot{\theta}}]\mathcal{V}_{\rm base} + \dot{\mathcal{V}}_{\rm base}\right), && (8.74)\\
\mathcal{F} &= \mathcal{L}^T(\theta)\left(\mathcal{G}\dot{\mathcal{V}} - [\mathrm{ad}_{\mathcal{V}}]^{\rm T}\mathcal{G}\mathcal{V} + \mathcal{F}_{\rm tip}\right), && (8.75)\\
\tau &= \mathcal{A}^T\mathcal{F}. && (8.76)
\end{aligned}$$

If the robot applies an external wrench $\mathcal{F}_{\rm tip}$ at the end-effector, this can be included into the dynamics equation

$$\tau = M(\theta)\ddot{\theta} + c(\theta,\dot{\theta}) + g(\theta) + J^{\rm T}(\theta)\mathcal{F}_{\rm tip}, \tag{8.77}$$

where $J(\theta)$ denotes the Jacobian of the forward kinematics expressed in the same reference frame as $\mathcal{F}_{\rm tip}$, and

$$\begin{aligned}
M(\theta) &= \mathcal{A}^T\mathcal{L}^{\rm T}(\theta)\mathcal{G}\mathcal{L}(\theta)\mathcal{A}, && (8.78)\\
c(\theta,\dot{\theta}) &= -\mathcal{A}^T\mathcal{L}^T(\theta)\left(\mathcal{G}\mathcal{L}(\theta)\,[\mathrm{ad}_{\mathcal{A}\dot{\theta}}]\mathcal{W}(\theta) + [\mathrm{ad}_{\mathcal{V}}]^{\rm T}\mathcal{G}\right)\mathcal{L}(\theta)\mathcal{A}\dot{\theta}, && (8.79)\\
g(\theta) &= \mathcal{A}^T\mathcal{L}^T(\theta)\mathcal{G}\mathcal{L}(\theta)\dot{\mathcal{V}}_{\rm base}. && (8.80)
\end{aligned}$$

8.5 Forward Dynamics of Open Chains

The forward dynamics problem involves solving

$$M(\theta)\ddot{\theta} = \tau(t) - h(\theta,\dot{\theta}) - J^{\rm T}(\theta)\mathcal{F}_{\rm tip} \tag{8.81}$$

for $\ddot{\theta}$, given θ, $\dot{\theta}$, τ, and the wrench $\mathcal{F}_{\rm tip}$ applied by the end-effector (if applicable). The term $h(\theta,\dot{\theta})$ can be computed by calling the inverse dynamics algorithm with $\ddot{\theta} = 0$ and $\mathcal{F}_{\rm tip} = 0$. The inertia matrix $M(\theta)$ can be computed using Equation (8.57). An alternative is to use n calls of the inverse dynamics algorithm to build $M(\theta)$ column by column. In each of the n calls, set $\mathfrak{g} = 0$, $\dot{\theta} = 0$, and $\mathcal{F}_{\rm tip} = 0$. In the first call, the column vector $\ddot{\theta}$ is all zeros except for a 1 in the first row. In the second call, $\ddot{\theta}$ is all zeros except for a 1 in the second row, and so on. The τ vector returned by the ith call is the ith column of $M(\theta)$, and after n calls the $n \times n$ matrix $M(\theta)$ is constructed.

With $M(\theta)$, $h(\theta,\dot{\theta})$, and $\mathcal{F}_{\rm tip}$ we can use any efficient algorithm for solving Equation (8.81), which is of the form $M\ddot{\theta} = b$, for $\ddot{\theta}$.

The forward dynamics can be used to simulate the motion of the robot given its initial state, the joint forces–torques $\tau(t)$, and an optional external wrench

$\mathcal{F}_{\text{tip}}(t)$, for $t \in [0, t_f]$. First define the function *ForwardDynamics* returning the solution to Equation (8.81), i.e.,

$$\ddot{\theta} = \mathit{ForwardDynamics}(\theta, \dot{\theta}, \tau, \mathcal{F}_{\text{tip}}).$$

Defining the variables $q_1 = \theta$, $q_2 = \dot{\theta}$, the second-order dynamics (8.81) can be converted to two first-order differential equations,

$$\begin{aligned} \dot{q}_1 &= q_2, \\ \dot{q}_2 &= \mathit{ForwardDynamics}(q_1, q_2, \tau, \mathcal{F}_{\text{tip}}). \end{aligned}$$

The simplest method for numerically integrating a system of first-order differential equations of the form $\dot{q} = f(q, t)$, $q \in \mathbb{R}^n$, is the first-order Euler iteration

$$q(t + \delta t) = q(t) + \delta t f(q(t), t),$$

where the positive scalar δt denotes the timestep. The Euler integration of the robot dynamics is thus

$$\begin{aligned} q_1(t + \delta t) &= q_1(t) + q_2(t)\delta t, \\ q_2(t + \delta t) &= q_2(t) + \mathit{ForwardDynamics}(q_1, q_2, \tau, \mathcal{F}_{\text{tip}})\delta t. \end{aligned}$$

Given a set of initial values for $q_1(0) = \theta(0)$ and $q_2(0) = \dot{\theta}(0)$, the above equations can be iterated forward in time to obtain the motion $\theta(t) = q_1(t)$ numerically.

Euler Integration Algorithm for Forward Dynamics

- **Inputs**: The initial conditions $\theta(0)$ and $\dot{\theta}(0)$, the input torques $\tau(t)$ and wrenches at the end-effector $\mathcal{F}_{\text{tip}}(t)$ for $t \in [0, t_f]$, and the number of integration steps N.
- **Initialization**: Set the timestep $\delta t = t_f/N$, and set $\theta[0] = \theta(0)$, $\dot{\theta}[0] = \dot{\theta}(0)$.
- **Iteration**: For $k = 0$ to $N - 1$ do

$$\begin{aligned} \ddot{\theta}[k] &= \mathit{ForwardDynamics}(\theta[k], \dot{\theta}[k], \tau(k\delta t), \mathcal{F}_{\text{tip}}(k\delta t)), \\ \theta[k+1] &= \theta[k] + \dot{\theta}[k]\delta t, \\ \dot{\theta}[k+1] &= \dot{\theta}[k] + \ddot{\theta}[k]\delta t. \end{aligned}$$

- **Output**: The joint trajectory $\theta(k\delta t) = \theta[k]$, $\dot{\theta}(k\delta t) = \dot{\theta}[k]$, $k = 0, \ldots, N$.

The result of the numerical integration converges to the theoretical result as the number of integration steps N goes to infinity. Higher-order numerical integration schemes, such as fourth-order Runge–Kutta, can yield a closer approximation with fewer computations than the simple first-order Euler method.

8.6 Dynamics in the Task Space

In this section we consider how the dynamic equations change under a transformation to coordinates of the end-effector frame (task-space coordinates). To keep things simple we consider a six-degree-of-freedom open chain with joint-space dynamics

$$\tau = M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}), \qquad \theta \in \mathbb{R}^6,\ \tau \in \mathbb{R}^6. \tag{8.82}$$

We also ignore, for the time being, any end-effector forces $\mathcal{F}_{\text{tip}}$. The twist $\mathcal{V} = (\omega, v)$ of the end-effector is related to the joint velocity $\dot{\theta}$ by

$$\mathcal{V} = J(\theta)\dot{\theta}, \tag{8.83}$$

with the understanding that $\mathcal{V}$ and $J(\theta)$ are always expressed in terms of the same reference frame. The time derivative $\dot{\mathcal{V}}$ is then

$$\dot{\mathcal{V}} = \dot{J}(\theta)\dot{\theta} + J(\theta)\ddot{\theta}. \tag{8.84}$$

At configurations θ where $J(\theta)$ is invertible, we have

$$\begin{aligned} \dot{\theta} &= J^{-1}\mathcal{V}, & (8.85)\\ \ddot{\theta} &= J^{-1}\dot{\mathcal{V}} - J^{-1}\dot{J}J^{-1}\mathcal{V}. & (8.86) \end{aligned}$$

Substituting for $\dot{\theta}$ and $\ddot{\theta}$ in Equation (8.82) leads to

$$\tau = M(\theta)\left(J^{-1}\dot{\mathcal{V}} - J^{-1}\dot{J}J^{-1}\mathcal{V}\right) + h(\theta, J^{-1}\mathcal{V}). \tag{8.87}$$

Let $J^{-\mathrm{T}}$ denote $(J^{-1})^{\mathrm{T}} = (J^{\mathrm{T}})^{-1}$. Pre-multiply both sides by $J^{-\mathrm{T}}$ to get

$$\begin{aligned} J^{-\mathrm{T}}\tau &= J^{-\mathrm{T}}MJ^{-1}\dot{\mathcal{V}} - J^{-\mathrm{T}}MJ^{-1}\dot{J}J^{-1}\mathcal{V} \\ &\quad + J^{-\mathrm{T}}h(\theta, J^{-1}\mathcal{V}). \end{aligned} \tag{8.88}$$

Expressing $J^{-\mathrm{T}}\tau$ as the wrench $\mathcal{F}$, the above can be written

$$\mathcal{F} = \Lambda(\theta)\dot{\mathcal{V}} + \eta(\theta, \mathcal{V}), \tag{8.89}$$

where

$$\Lambda(\theta) = J^{-\mathrm{T}} M(\theta) J^{-1}, \tag{8.90}$$
$$\eta(\theta, \mathcal{V}) = J^{-\mathrm{T}} h(\theta, J^{-1}\mathcal{V}) - \Lambda(\theta)\dot{J}J^{-1}\mathcal{V}. \tag{8.91}$$

These are the dynamic equations expressed in end-effector frame coordinates. If an external wrench $\mathcal{F}$ is applied to the end-effector frame then, assuming the actuators provide zero forces and torques, the motion of the end-effector frame is governed by these equations.

Note that $J(\theta)$ must be invertible (i.e., there must be a one-to-one mapping between joint velocities and end-effector twists) in order to derive the task-space dynamics above. Also note the dependence of $\Lambda(\theta)$ and $\eta(\theta, \mathcal{V})$ on θ. In general, we cannot replace the dependence on θ by a dependence on the end-effector configuration X because there may be multiple solutions to the inverse kinematics, and the dynamics depends on the specific joint configuration θ.

8.7 Constrained Dynamics

Now consider the case where the n-joint robot is subject to a set of k holonomic or nonholonomic Pfaffian velocity constraints of the form

$$A(\theta)\dot{\theta} = 0, \qquad A(\theta) \in \mathbb{R}^{k \times n}. \tag{8.92}$$

(See Section 2.4 for an introduction to Pfaffian constraints.) Such constraints can come from loop-closure constraints; for example, the motion of an end-effector rigidly holding a door handle is subject to $k = 5$ constraints due to the hinges of the door. As another example, a robot writing with a pen is subject to a single constraint that keeps the height of the tip of the pen above the paper at zero. In any case, we assume that the constraints do no work on the robot, i.e., the generalized forces τ_{con} due to the constraints satisfy

$$\tau_{\mathrm{con}}^{\mathrm{T}}\dot{\theta} = 0.$$

This assumption means that τ_{con} must be a linear combination of the columns of $A^{\mathrm{T}}(\theta)$, i.e., $\tau_{\mathrm{con}} = A^{\mathrm{T}}(\theta)\lambda$ for some $\lambda \in \mathbb{R}^k$, since these are the generalized forces that do no work when $\dot{\theta}$ is subject to the constraints (8.92):

$$(A^{\mathrm{T}}(\theta)\lambda)^{\mathrm{T}}\dot{\theta} = \lambda^{\mathrm{T}} A(\theta)\dot{\theta} = 0 \qquad \text{for all } \lambda \in \mathbb{R}^k.$$

For the writing-robot example, the assumption that the constraint is workless means that there can be no friction between the pen and the paper.

Adding the constraint forces $A^{\mathrm{T}}(\theta)\lambda$ to the equations of motion, we can write the $n+k$ constrained equations of motion in the $n+k$ unknowns $\{\ddot{\theta}, \lambda\}$ (the forward dynamics) or in the $n+k$ unknowns $\{\tau, \lambda\}$ (the inverse dynamics):

$$\tau = M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}) + A^{\mathrm{T}}(\theta)\lambda, \tag{8.93}$$

$$A(\theta)\dot{\theta} = 0, \tag{8.94}$$

where λ is a set of Lagrange multipliers and $A^{\mathrm{T}}(\theta)\lambda$ are the joint forces and torques that act against the constraints. From these equations, it should be clear that the robot has $n-k$ velocity freedoms and k "force freedoms" – the robot is free to create any generalized force of the form $A^{\mathrm{T}}(\theta)\lambda$. (For the writing robot, there is an equality constraint: the robot can only apply pushing forces into the paper and table, not pulling forces.)

Often it is convenient to reduce the $n+k$ equations in $n+k$ unknowns to n equations in n unknowns, without explicitly calculating the Lagrange multipliers λ. To do this, we can solve for λ in terms of the other quantities and substitute our solution into Equation (8.93). Since the constraints are satisfied at all times, the time rate of change of the constraints satisfies

$$\dot{A}(\theta)\dot{\theta} + A(\theta)\ddot{\theta} = 0. \tag{8.95}$$

Assuming that $M(\theta)$ and $A(\theta)$ are full rank, we can solve Equation (8.93) for $\ddot{\theta}$, substitute into Equation (8.95), and omit the dependences on θ and $\dot{\theta}$ for conciseness, to get

$$\dot{A}\dot{\theta} + AM^{-1}(\tau - h - A^{\mathrm{T}}\lambda) = 0. \tag{8.96}$$

Using $A\ddot{\theta} = -\dot{A}\dot{\theta}$, after some manipulation, we obtain

$$\lambda = (AM^{-1}A^{\mathrm{T}})^{-1}(AM^{-1}(\tau - h) - A\ddot{\theta}). \tag{8.97}$$

Combining Equation (8.97) with Equation (8.93) and manipulating further, we get

$$P\tau = P(M\ddot{\theta} + h) \tag{8.98}$$

where

$$P = I - A^{\mathrm{T}}(AM^{-1}A^{\mathrm{T}})^{-1}AM^{-1} \tag{8.99}$$

and where I is the $n \times n$ identity matrix. The $n \times n$ matrix $P(\theta)$ is of rank $n-k$, and maps the joint generalized forces τ to $P(\theta)\tau$, projecting away the generalized force components that act on the constraints while retaining the generalized forces that do work on the robot. The complementary projection, $I - P(\theta)$, maps τ to $(I - P(\theta))\tau$, the joint forces that act on the constraints and do no work on the robot.

Using $P(\theta)$, we can solve the inverse dynamics by evaluating the right-hand side of Equation (8.98). The solution $P(\theta)\tau$ is a set of generalized joint forces that achieves the feasible components of the desired joint acceleration $\ddot{\theta}$. To this solution we can add any constraint forces $A^{\mathrm{T}}(\theta)\lambda$ without changing the acceleration of the robot.

We can rearrange Equation (8.98) into the form

$$P_{\ddot{\theta}}\ddot{\theta} = P_{\ddot{\theta}}M^{-1}(\tau - h), \tag{8.100}$$

where

$$P_{\ddot{\theta}} = M^{-1}PM = I - M^{-1}A^{\mathrm{T}}(AM^{-1}A^{\mathrm{T}})^{-1}A. \tag{8.101}$$

The rank-$(n-k)$ projection matrix $P_{\ddot{\theta}}(\theta) \in \mathbb{R}^{n\times n}$ projects away the components of an arbitrary joint acceleration $\ddot{\theta}$ that violate the constraints, leaving only the components $P_{\ddot{\theta}}(\theta)\ddot{\theta}$ that satisfy the constraints. To solve the forward dynamics, we evaluate the right-hand side of Equation (8.100), and the resulting $P_{\ddot{\theta}}\ddot{\theta}$ is the set of joint accelerations.

In Section 11.6 we discuss the related topic of hybrid motion–force control, in which the goal at each instant is to simultaneously achieve a desired acceleration satisfying $A\ddot{\theta} = 0$ (consisting of $n-k$ motion freedoms) and a desired force against the k constraints. In that section we use the task-space dynamics to represent the task-space end-effector motions and forces more naturally.

8.8 Robot Dynamics in the URDF

As described in Section 4.2 and illustrated in the UR5 Universal Robot Description Format file, the inertial properties of link i are described in the URDF by the link elements `mass`, `origin` (the position and orientation of the center-of-mass frame relative to a frame attached at joint i), and `inertia`, which specifies the six elements of the symmetric rotational inertia matrix on or above the diagonal. To fully write the robot's dynamics, for joint i we need in addition the joint element `origin`, specifying the position and orientation of link i's joint frame relative to link $(i-1)$'s joint frame when $\theta_i = 0$, and the element `axis`, which specifies the axis of motion of joint i. We leave to the exercises the translation of these elements into the quantities needed for the Newton–Euler inverse dynamics algorithm.

8.9 Actuation, Gearing, and Friction

Until now we have been assuming the existence of actuators that directly provide commanded forces and torques. In practice there are many types of actuators (e.g., electric, hydraulic, and pneumatic) and mechanical power transformers (e.g., gearheads), and the actuators can be located at the joints themselves or remotely, with mechanical power transmitted by cables or timing belts. Each combination of these has its own characteristics that can play a significant role in the "extended dynamics" mapping the actual control inputs (e.g., the current requested of amplifiers connected to electric motors) to the motion of the robot.

In this section we provide an introduction to some of the issues associated with one particular, and common, configuration: geared DC electric motors located at each joint. This is the configuration used in the Universal Robots UR5, for example.

Figure 8.7 shows the electrical block diagram for a typical n-joint robot driven by DC electric motors. For concreteness, we assume that each joint is revolute. A power supply converts the wall AC voltage to a DC voltage to power the amplifier associated with each motor. A control box takes user input, for example in the form of a desired trajectory, as well as position feedback from encoders located at each joint. Using the desired trajectory, a model of the robot's dynamics, and the measured error in the current robot state relative to the desired robot state, the controller calculates the torque required of each actuator. Since DC electric motors nominally provide a torque proportional to the current through the motor, this torque command is equivalent to a current command. Each motor amplifier then uses a current sensor (shown as external to the amplifier in Figure 8.7, but in reality internal to the amplifier) to continually adjust the voltage across the motor to try to achieve the requested current.[3] The motion of the motor is sensed by the motor encoder, and the position information is sent back to the controller.

The commanded torque is typically updated at around 1000 times per second (1 kHz), and the amplifier's voltage control loop may be updated at a rate ten times that or more.

Figure 8.8 is a conceptual representation of the motor and other components for a single axis. The motor has a single shaft extending from both ends of the motor: one end drives a rotary encoder, which measures the position of the joint, and the other end becomes the input to a gearhead. The gearhead increases the torque while reducing the speed, since most DC electric motors with an appropriate power rating provide torques that are too low to be useful for robotics applications. The purpose of the bearing is to support the gearhead output, freely transmitting torques about the gearhead axis while isolating the gearhead (and motor) from wrench components due to link $i+1$ in the other five directions. The outer cases of the encoder, motor, gearhead, and bearing

[3] The voltage is typically a time-averaged voltage achieved by the duty cycle of a voltage rapidly switching between a maximum positive voltage and a maximum negative voltage.

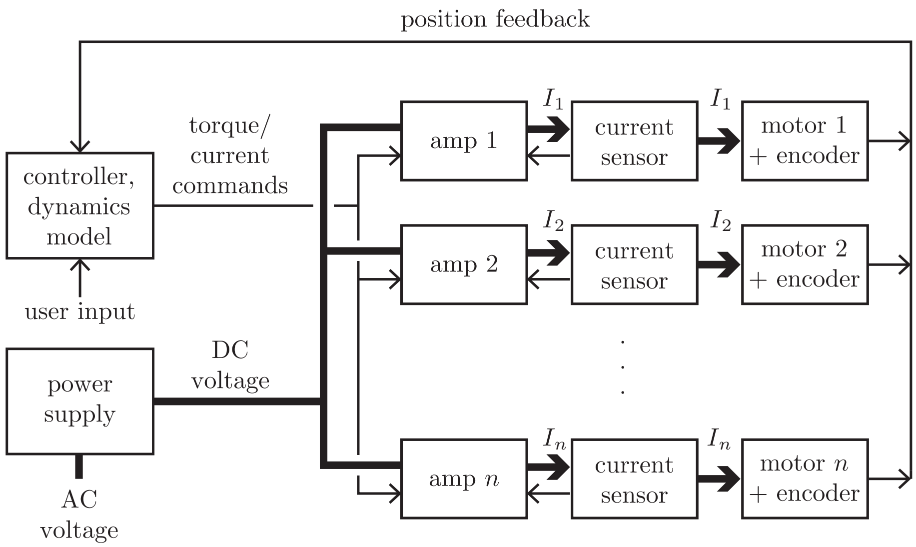


Figure 8.7: A block diagram of a typical n-joint robot. The bold lines correspond to high-power signals while the thin lines correspond to communication signals.

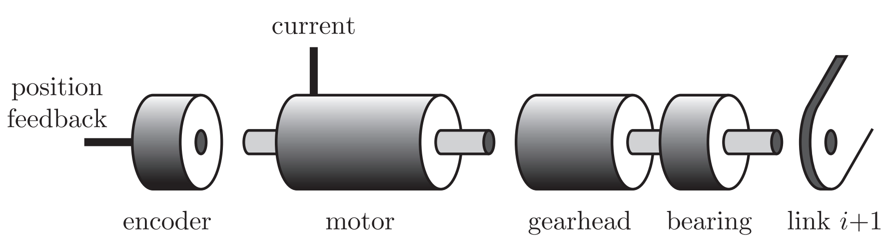


Figure 8.8: The outer cases of the encoder, motor, gearhead, and bearing are fixed in link i, while the gearhead output shaft supported by the bearing is fixed in link $i + 1$.

are all fixed relative to each other and to link i. It is also common for the motor to have some kind of brake, not shown.

8.9.1 DC Motors and Gearing

A DC motor consists of a **stator** and a **rotor** that rotates relative to the stator. DC electric motors create torque by sending current through windings in a magnetic field created by permanent magnets, where the magnets are attached to the stator and the windings are attached to the rotor, or vice versa. A DC motor has multiple windings, some of which are energized and some of which are inactive at any given time. The windings that are energized are chosen as a function of the angle of the rotor relative to the stator. This "commutation" of the windings occurs mechanically using brushes (brushed motors) or electrically using control circuitry (brushless motors). Brushless motors have the advantage of no brush wear and higher continuous torque, since the windings are typically attached to the motor housing where the heat due to the resistance of the windings can be more easily dissipated. In our basic introduction to DC motor modeling, we do not distinguish between brushed and brushless motors.

Figure 8.9 shows a brushed DC motor with an encoder and a gearhead.

The torque τ, measured in newton-meters (N m), created by a DC motor is governed by the equation

$$\tau = k_t I,$$

where I, measured in amps (A), is the current through the windings. The constant k_t, measured in newton-meters per amp (N m/A), is called the **torque constant**. The power dissipated as heat by the windings, measured in watts (W), is governed by

$$P_{\text{heat}} = I^2 R,$$

where R is the resistance of the windings in ohms (Ω). To keep the motor windings from overheating, the continuous current flowing through the motor must be limited. Accordingly, in continuous operation, the motor torque must be kept below a continuous-torque limit τ_{cont} determined by the thermal properties of the motor.

A simplified model of a DC motor, where all units are in the SI system, can be derived by equating the electrical power consumed by the motor $P_{\text{elec}} = IV$ in watts (W) to the mechanical power $P_{\text{mech}} = \tau w$ (also in W) and other power produced by the motor,

$$IV = \tau w + I^2 R + LI\frac{dI}{dt} + \text{friction and other power-loss terms},$$

where V is the voltage applied to the motor in volts (V), w is the angular speed of the motor in radians per second (1/s), and L is the inductance due to the windings in henries (H). The terms on the right-hand side are the mechanical power produced by the motor, the power lost to heating the windings due to the resistance of the wires, the power consumed or produced by energizing or de-energizing the inductance of the windings (since the energy stored in an inductor is $\frac{1}{2}LI^2$, and power is the time derivative of energy), and the power

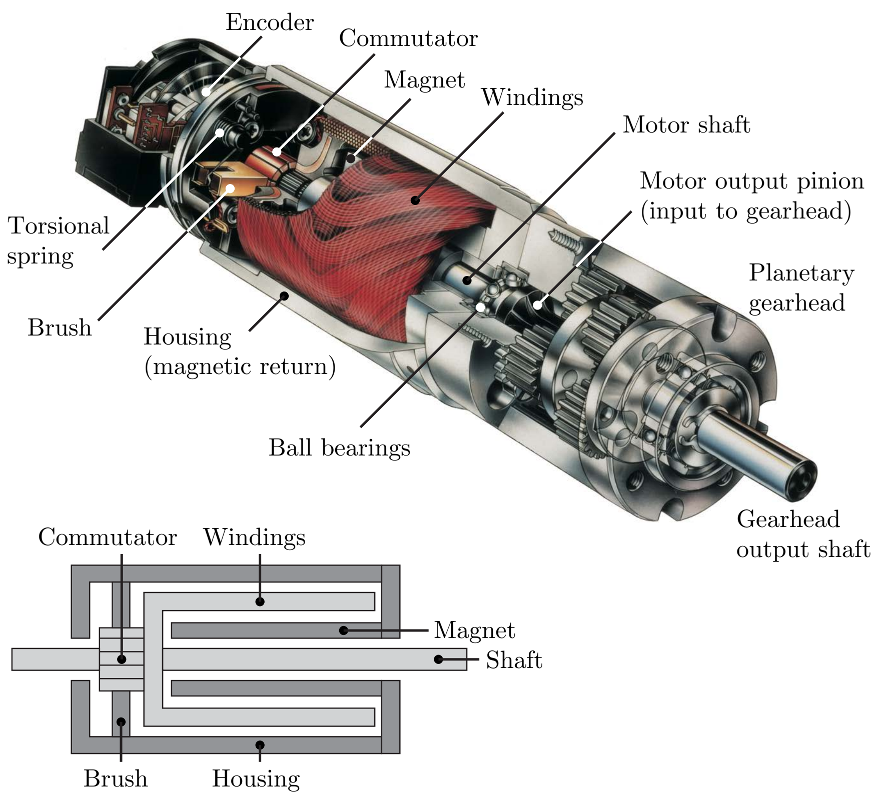


Figure 8.9: (Top) A cutaway view of a Maxon brushed DC motor with an encoder and gearhead. (Cutaway image courtesy of Maxon Precision Motors, Inc., maxonmotorusa.com.) The motor's rotor consists of the windings, commutator ring, and shaft. Each of the several windings connects different segments of the commutator, and as the motor rotates, the two brushes slide over the commutator ring and make contact with different segments, sending current through one or more windings. One end of the motor shaft turns the encoder, and the other end is input to the gearhead. (Bottom) A simplified cross-section of the motor only, showing the stator (brushes, housing, and magnets) in dark gray and the rotor (windings, commutator, and shaft) in light gray.

lost to friction in bearings, etc. Dropping this last term, replacing τw by $k_t I w$, and dividing by I, we get the voltage equation

$$V = k_t w + IR + L\frac{dI}{dt}. \tag{8.102}$$

Often Equation (8.102) is written with the **electrical constant** k_e (with units of V s) instead of the torque constant k_t, but in SI units (V s or N m/A) the numerical values of the two are identical; they represent the same constant property of the motor. So we prefer to use k_t.

The voltage term $k_t w$ in Equation (8.102) is called the **back electromotive force** or **back-emf** for short, and it is what differentiates a motor from being simply a resistor and inductor in series. It also allows a motor, which we usually think of as converting electrical power to mechanical, to be run as a generator, converting mechanical power to electrical. If the motor's electrical inputs are disconnected (so no current can flow) and the shaft is forced to turn by some external torque, you can measure the back-emf voltage $k_t w$ across the motor's inputs.

For simplicity, in the rest of this section we ignore the $L\,dI/dt$ term. This assumption is exactly satisfied when the motor is operating at a constant current. With this assumption, Equation (8.102) can be rearranged to

$$w = \frac{1}{k_t}(V - IR) = \frac{V}{k_t} - \frac{R}{k_t^2}\tau,$$

expressing the speed w as a linear function of τ (with a slope of $-R/k_t^2$) for a constant V. Now assume that the voltage across the motor is limited to the range $[-V_{\max}, +V_{\max}]$ and the current through the motor is limited to $[-I_{\max}, +I_{\max}]$, perhaps by the amplifier or power supply. Then the operating region of the motor in the torque–speed plane is as shown in Figure 8.10. Note that the signs of τ and w are opposite in the second and fourth quadrants of this plane, and therefore the product τw is negative. When the motor operates in these quadrants, it is actually consuming mechanical power, not producing mechanical power. The motor is acting like a damper.

Focusing on the first quadrant ($\tau \geq 0, w \geq 0, \tau w \geq 0$), the boundary of the operating region is called the **speed–torque curve**. The **no-load speed** $w_0 = V_{\max}/k_t$ at one end of the speed–torque curve is the speed at which the motor spins when it is powered by $V_{\max}$ but is providing no torque. In this operating condition, the back-emf $k_t w$ is equal to the applied voltage, so there is no voltage remaining to create current (or torque). The **stall torque** $\tau_{\text{stall}} = k_t V_{\max}/R$ at the other end of the speed–torque curve is achieved when the shaft is blocked from spinning, so there is no back-emf.

Figure 8.10 also indicates the continuous operating region where $|\tau| \leq \tau_{\text{cont}}$. The motor may be operated intermittently outside the continuous operating region, but extended operation outside the continuous operating region raises the possibility that the motor will overheat.

The motor's rated mechanical power is $P_{\text{rated}} = \tau_{\text{cont}} w_{\text{cont}}$, where w_{cont} is

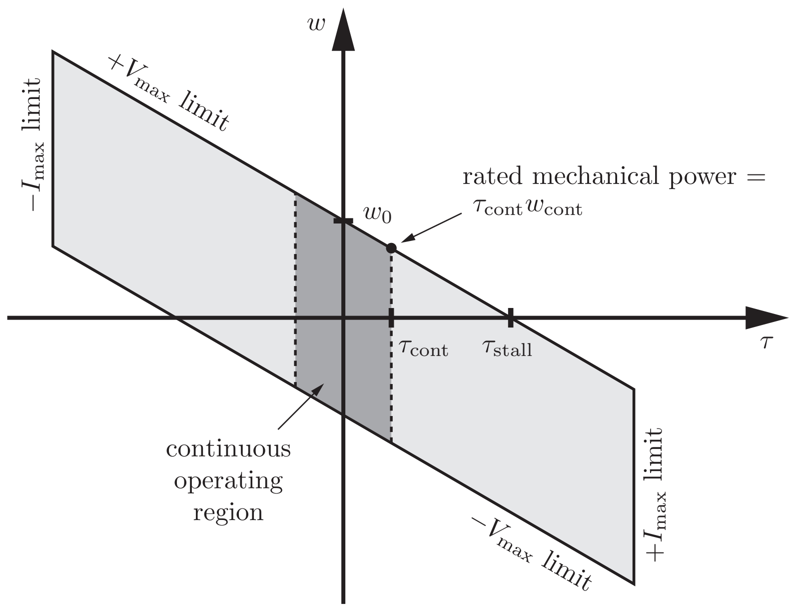


Figure 8.10: The operating region (dark gray) of a current- and voltage-limited DC electric motor, and its continuous operating region (light gray).

the speed on the speed–torque curve corresponding to τ_{cont}. Even if the motor's rated power is sufficient for a particular application, the torque generated by a DC motor is typically too low to be useful. As mentioned earlier, gearing is therefore used to increase the torque while also decreasing the speed. For a gear ratio G, the output speed of the gearhead is

$$w_{\text{gear}} = \frac{w_{\text{motor}}}{G}.$$

For an ideal gearhead, no power is lost in the torque conversion, so $\tau_{\text{motor}} w_{\text{motor}} = \tau_{\text{gear}} w_{\text{gear}}$, which implies that

$$\tau_{\text{gear}} = G\tau_{\text{motor}}.$$

In practice, some mechanical power is lost due to friction and impacts between gear teeth, bearings, etc., so

$$\tau_{\text{gear}} = \eta G \tau_{\text{motor}},$$

where $\eta \leq 1$ is the efficiency of the gearhead.

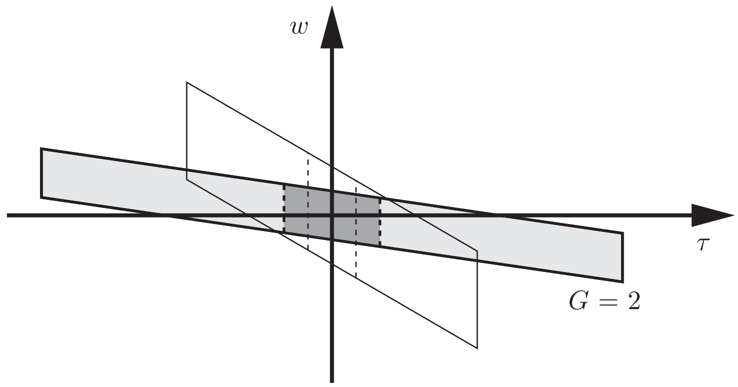


Figure 8.11: The original motor operating region, and the operating region with a gear ratio $G = 2$ showing the increased torque and decreased speed.

Figure 8.11 shows the operating region of the motor from Figure 8.10 when the motor is geared by $G = 2$ (with $\eta = 1$). The maximum torque doubles, while the maximum speed shrinks by a factor of two. Since many DC motors are capable of no-load speeds of 10 000 rpm or more, robot joints often have gear ratios of 100 or more to achieve an appropriate compromise between speed and torque.

8.9.2 Apparent Inertia

The motor's stator is attached to one link and the rotor is attached to another link, possibly through a gearhead. Therefore, when calculating the contribution of a motor to the masses and inertias of the links, the mass and inertia of the stator must be assigned to one link and the mass and inertia of the rotor must be assigned to the other link.

Consider a stationary link 0 with the stator of the joint-1 gearmotor attached to it. The rotational speed of joint 1, the output of the gearhead, is $\dot{\theta}$. Therefore the motor's rotor rotates at $G\dot{\theta}$. The kinetic energy of the rotor is therefore

$$\mathcal{K} = \frac{1}{2}\mathcal{I}_{\text{rotor}}(G\dot{\theta})^2 = \frac{1}{2} \underbrace{G^2\mathcal{I}_{\text{rotor}}}_{\text{apparent inertia}} \dot{\theta}^2,$$

where $\mathcal{I}_{\text{rotor}}$ is the rotor's scalar inertia about the rotation axis and $G^2\mathcal{I}_{\text{rotor}}$ is the **apparent inertia** (often called the **reflected inertia**) of the rotor about the axis. In other words, if you were to grab link 1 and rotate it manually, the inertia contributed by the rotor would feel as if it were a factor G^2 larger than its actual inertia, owing to the gearhead.

While the inertia $\mathcal{I}_{\text{rotor}}$ is typically much less than the inertia $\mathcal{I}_{\text{link}}$ of the rest of the link about the rotation axis, the apparent inertia $G^2\mathcal{I}_{\text{rotor}}$ may be on the order of, or even larger than, $\mathcal{I}_{\text{link}}$.

One consequence as the gear ratio becomes large is that the inertia seen by joint i becomes increasingly dominated by the apparent inertia of the rotor. In other words, the torque required of joint i becomes relatively more dependent on $\ddot{\theta}_i$ than on other joint accelerations, i.e., the robot's mass matrix becomes more diagonal. In the limit when the mass matrix has negligible off-diagonal components (and in the absence of gravity), the dynamics of the robot are decoupled – the dynamics at one joint has no dependence on the configuration or motion of the other joints.

As an example, consider the 2R arm of Figure 8.1 with $L_1 = L_2 = \mathfrak{m}_1 = \mathfrak{m}_2 = 1$. Now assume that each of joint 1 and joint 2 has a motor of mass 1, with a stator of inertia 0.005 and a rotor of inertia 0.00125, and a gear ratio G (with $\eta = 1$). With a gear ratio $G = 10$, the mass matrix is

$$M(\theta) = \begin{bmatrix} 4.13 + 2\cos\theta_2 & 1.01 + \cos\theta_2 \\ 1.01 + \cos\theta_2 & 1.13 \end{bmatrix}.$$

With a gear ratio $G = 100$, the mass matrix is

$$M(\theta) = \begin{bmatrix} 16.5 + 2\cos\theta_2 & 1.13 + \cos\theta_2 \\ 1.13 + \cos\theta_2 & 13.5 \end{bmatrix}.$$

The off-diagonal components are relatively less important for this second robot. The available joint torques of the second robot are ten times that of the first robot so, despite the increases in the mass matrix elements, the second robot is capable of significantly higher accelerations and end-effector payloads. The top speed of each joint of the second robot is ten times less than that of the first robot, however.

If the apparent inertia of the rotor is non-negligible relative to the inertia of the rest of the link, the Newton–Euler inverse dynamics algorithm must be modified to account for it. One approach is to treat the link as consisting of two separate bodies, the geared rotor driving the link and the rest of the link, each with its own center of mass and inertial properties (where the link inertial properties include the inertial properties of the stator of any motor mounted on the link). In the forward iteration, the twist and acceleration of each body is determined while accounting for the gearhead in calculating the rotor's motion. In the backward iteration, the wrench on the link is calculated as the sum of two

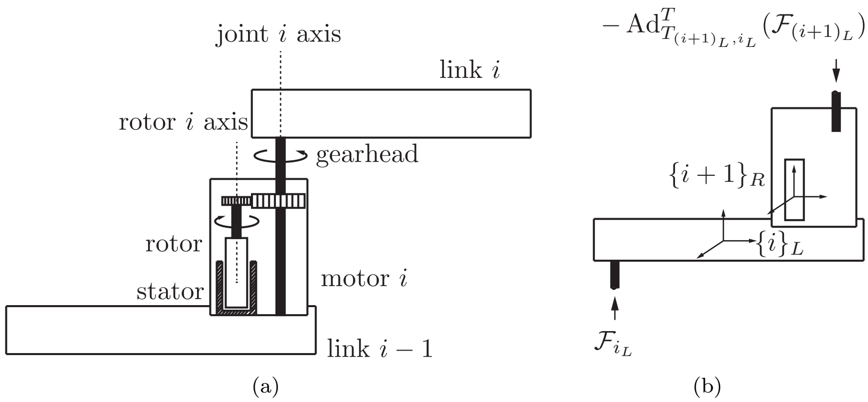


Figure 8.12: (a) Schematic of a geared motor between links $i-1$ and i. (b) The free-body diagram for link i, which is analogous to Figure 8.6.

wrenches: (i) the link wrench as given by Equation (8.53) and (ii) the reaction wrench from the distal rotor. The resultant wrench projected onto the joint axis is then the gear torque τ_{gear}; dividing τ_{gear} by the gear ratio and adding to this the torque resulting from the acceleration of the rotor results in the required motor torque τ_{motor}. The current command to the DC motor is then $I_{\text{com}} = \tau_{\text{motor}}/(\eta k_t)$.

8.9.3 Newton–Euler Inverse Dynamics Algorithm Accounting for Motor Inertias and Gearing

We now reformulate the recursive Newton–Euler inverse dynamics algorithm taking into account the apparent inertias as discussed above. Figure 8.12 illustrates the setup. We assume massless gears and shafts and that the friction between gears as well as the friction between shafts and links is negligible.

Initialization Attach a frame $\{0\}_L$ to the base, frames $\{1\}_L$ to $\{n\}_L$ to the centers of mass of links 1 to n, and frames $\{1\}_R$ to $\{n\}_R$ to the centers of mass of rotors 1 to n. Frame $\{n+1\}_L$ is attached to the end-effector, which is assumed fixed with respect to frame $\{n\}_L$. Define $M_{i_R,(i-1)_L}$ and $M_{i_L,(i-1)_L}$ to be the configuration of $\{i-1\}_L$ in $\{i\}_R$ and in $\{i\}_L$, respectively, when $\theta_i = 0$. Let $\mathcal{A}_i$ be the screw axis of joint i expressed in $\{i\}_L$. Similarly, let $\mathcal{R}_i$ be the screw axis of rotor i expressed in $\{i\}_R$. Let $\mathcal{G}_{i_L}$ be the 6×6 spatial inertia matrix of link i that includes the inertia of the attached stator and $\mathcal{G}_{i_R}$ be the 6×6 spatial inertia matrix of rotor i. The gear ratio of motor i is G_i. The twists $\mathcal{V}_{0_L}$ and $\dot{\mathcal{V}}_{0_L}$ and the wrench $\mathcal{F}_{(n+1)_L}$ are defined in the same way as $\mathcal{V}_0$, $\dot{\mathcal{V}}_0$, and $\mathcal{F}_{n+1}$ in Section 8.3.2.

Forward iterations Given $\theta, \dot{\theta}, \ddot{\theta}$, for $i = 1$ to n do

$$\begin{aligned}
T_{i_R,(i-1)_L} &= e^{-[\mathcal{R}_i]G_i\theta_i} M_{i_R,(i-1)_L}, && (8.103)\\
T_{i_L,(i-1)_L} &= e^{-[\mathcal{A}_i]\theta_i} M_{i_L,(i-1)_L}, && (8.104)\\
\mathcal{V}_{i_R} &= \mathrm{Ad}_{T_{i_R,(i-1)_L}}(\mathcal{V}_{(i-1)_L}) + \mathcal{R}_i G_i \dot{\theta}_i, && (8.105)\\
\mathcal{V}_{i_L} &= \mathrm{Ad}_{T_{i_L,(i-1)_L}}(\mathcal{V}_{(i-1)_L}) + \mathcal{A}_i \dot{\theta}_i, && (8.106)\\
\dot{\mathcal{V}}_{i_R} &= \mathrm{Ad}_{T_{i_R,(i-1)_L}}(\dot{\mathcal{V}}_{(i-1)_L}) + \mathrm{ad}_{\mathcal{V}_{i_R}}(\mathcal{R}_i)G_i\dot{\theta}_i + \mathcal{R}_i G_i \ddot{\theta}_i, && (8.107)\\
\dot{\mathcal{V}}_{i_L} &= \mathrm{Ad}_{T_{i_L,(i-1)_L}}(\dot{\mathcal{V}}_{(i-1)_L}) + \mathrm{ad}_{\mathcal{V}_{i_L}}(\mathcal{A}_i)\dot{\theta}_i + \mathcal{A}_i \ddot{\theta}_i. && (8.108)
\end{aligned}$$

Backward iterations For $i = n$ to 1 do

$$\begin{aligned}
\mathcal{F}_{i_L} &= \mathrm{Ad}^{\mathrm{T}}_{T_{(i+1)_L,i_L}}(\mathcal{F}_{(i+1)_L}) + \mathcal{G}_{i_L}\dot{\mathcal{V}}_{i_L} - \mathrm{ad}^{\mathrm{T}}_{\mathcal{V}_{i_L}}(\mathcal{G}_{i_L}\mathcal{V}_{i_L})\\
&\quad + \mathrm{Ad}^{\mathrm{T}}_{T_{(i+1)_R,i_L}}(\mathcal{G}_{(i+1)_R}\dot{\mathcal{V}}_{(i+1)_R} - \mathrm{ad}^{\mathrm{T}}_{\mathcal{V}_{(i+1)_R}}(\mathcal{G}_{(i+1)_R}\mathcal{V}_{(i+1)_R})), && (8.109)\\
\tau_{i,\mathrm{gear}} &= \mathcal{A}_i^{\mathrm{T}}\mathcal{F}_{i_L}, && (8.110)\\
\tau_{i,\mathrm{motor}} &= \frac{\tau_{i,\mathrm{gear}}}{G_i} + \mathcal{R}_i^{\mathrm{T}}(\mathcal{G}_{i_R}\dot{\mathcal{V}}_{i_R} - \mathrm{ad}^{\mathrm{T}}_{\mathcal{V}_{i_R}}(\mathcal{G}_{i_R}\mathcal{V}_{i_R})). && (8.111)
\end{aligned}$$

In the backward iteration stage, the quantity $\mathcal{F}_{(n+1)_L}$ occurring in the first step of the backward iteration is taken to be the external wrench applied to the end-effector (expressed in the $\{n+1\}_L$ frame), with $\mathcal{G}_{(n+1)_R}$ set to zero; $\mathcal{F}_{i_L}$ denotes the wrench applied to link i via the motor i gearhead (expressed in the $\{i_L\}$ frame); $\tau_{i,\mathrm{gear}}$ is the torque generated at the motor i gearhead; and $\tau_{i,\mathrm{motor}}$ is the torque at rotor i.

Note that if there is no gearing then no modification to the original Newton–Euler inverse dynamics algorithm is necessary; the stator is attached to one link and the rotor is attached to another link. Robots constructed with a motor at each axis and no gearheads are sometimes called **direct-drive robots**. Direct-drive robots have low friction, but they see limited use because typically the motors must be large and heavy to generate appropriate torques.

No modification is needed to the Lagrangian approach to the dynamics to handle geared motors, provided that we can correctly represent the kinetic energy of the faster-spinning rotors.

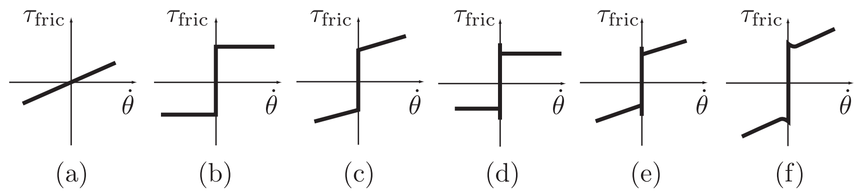


Figure 8.13: Examples of velocity-dependent friction models. (a) Viscous friction, $\tau_{\text{fric}} = b_{\text{viscous}}\dot{\theta}$. (b) Coulomb friction, $\tau_{\text{fric}} = b_{\text{static}}\,\text{sgn}(\dot{\theta})$; τ_{fric} can take any value in $[-b_{\text{static}}, b_{\text{static}}]$ at zero velocity. (c) Static plus viscous friction, $\tau_{\text{fric}} = b_{\text{static}}\,\text{sgn}(\dot{\theta}) + b_{\text{viscous}}\dot{\theta}$. (d) Static and kinetic friction, requiring $\tau_{\text{fric}} \geq |b_{\text{static}}|$ to initiate motion and then $\tau_{\text{fric}} = b_{\text{kinetic}}\,\text{sgn}(\dot{\theta})$ during motion, where $b_{\text{static}} > b_{\text{kinetic}}$. (e) Static, kinetic, and viscous friction. (f) A friction law exhibiting the Stribeck effect – at low velocities, the friction decreases as the velocity increases.

8.9.4 Friction

The Lagrangian and Newton–Euler dynamics do not account for friction at the joints, but the friction forces and torques in gearheads and bearings may be significant. Friction is a complex phenomenon that is the subject of considerable current research; any friction model is a gross attempt to capture the average behavior of the micromechanics of contact.

Friction models often include a **static friction** term and a velocity-dependent **viscous friction** term. The presence of a static friction term means that a nonzero torque is required to cause the joint to begin to move. The viscous friction term indicates that the amount of friction torque increases with increasing velocity of the joint. See Figure 8.13 for some examples of velocity-dependent friction models.

Other factors may contribute to the friction at a joint, including the loading of the joint bearings, the time the joint has been at rest, the temperature, etc. The friction in a gearhead often increases as the gear ratio G increases.

8.9.5 Joint and Link Flexibility

In practice, a robot's joints and links are likely to exhibit some flexibility. For example, the flexspline element of a harmonic drive gearhead achieves essentially zero backlash by being somewhat flexible. A model of a joint with harmonic drive gearing, then, could include a relatively stiff torsional spring between the motor's rotor and the link to which the gearhead is attached.

Similarly, links themselves are not infinitely stiff. Their finite stiffness is exhibited as vibrations along the link.

Flexible joints and links introduce extra states to the dynamics of the robot, significantly complicating the dynamics and control. While many robots are designed to be stiff in order to minimize these complexities, in some cases this is impractical owing to the extra link mass required to create the stiffness.

8.10 Summary

- Given a set of generalized coordinates θ and generalized forces τ, the Euler–Lagrange equations can be written

$$\tau = \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta},$$

where $\mathcal{L}(\theta, \dot{\theta}) = \mathcal{K}(\theta, \dot{\theta}) - \mathcal{P}(\theta)$, $\mathcal{K}$ is the kinetic energy of the robot, and $\mathcal{P}$ is the potential energy of the robot.

- The equations of motion of a robot can be written in the following equivalent forms:

$$\begin{aligned} \tau &= M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}) \\ &= M(\theta)\ddot{\theta} + c(\theta, \dot{\theta}) + g(\theta) \\ &= M(\theta)\ddot{\theta} + \dot{\theta}^{\mathrm{T}}\Gamma(\theta)\dot{\theta} + g(\theta) \\ &= M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta), \end{aligned}$$

where $M(\theta)$ is the $n \times n$ symmetric positive-definite mass matrix, $h(\theta, \dot{\theta})$ is the sum of the generalized forces due to the gravity and quadratic velocity terms, $c(\theta, \dot{\theta})$ are quadratic velocity forces, $g(\theta)$ are gravitational forces, $\Gamma(\theta)$ is an $n \times n \times n$ matrix of Christoffel symbols of the first kind obtained from partial derivatives of $M(\theta)$ with respect to θ, and $C(\theta, \dot{\theta})$ is the $n \times n$ Coriolis matrix whose (i, j)th entry is given by

$$c_{ij}(\theta, \dot{\theta}) = \sum_{k=1}^{n} \Gamma_{ijk}(\theta)\dot{\theta}_k.$$

If the end-effector of the robot is applying a wrench $\mathcal{F}_{\mathrm{tip}}$ to the environment, the term $J^{\mathrm{T}}(\theta)\mathcal{F}_{\mathrm{tip}}$ should be added to the right-hand side of the robot's dynamic equations.

- The symmetric positive-definite rotational inertia matrix of a rigid body is

$$\mathcal{I}_b = \begin{bmatrix} \mathcal{I}_{xx} & \mathcal{I}_{xy} & \mathcal{I}_{xz} \\ \mathcal{I}_{xy} & \mathcal{I}_{yy} & \mathcal{I}_{yz} \\ \mathcal{I}_{xz} & \mathcal{I}_{yz} & \mathcal{I}_{zz} \end{bmatrix},$$

where

$$\begin{array}{ll} \mathcal{I}_{xx} = \int_{\mathcal{B}}(y^2+z^2)\rho(x,y,z)dV, & \mathcal{I}_{yy} = \int_{\mathcal{B}}(x^2+z^2)\rho(x,y,z)dV, \\ \mathcal{I}_{zz} = \int_{\mathcal{B}}(x^2+y^2)\rho(x,y,z)dV, & \mathcal{I}_{xy} = -\int_{\mathcal{B}} xy\rho(x,y,z)dV, \\ \mathcal{I}_{xz} = -\int_{\mathcal{B}} xz\rho(x,y,z)dV, & \mathcal{I}_{yz} = -\int_{\mathcal{B}} yz\rho(x,y,z)dV, \end{array}$$

$\mathcal{B}$ is the volume of the body, dV is a differential volume element, and $\rho(x,y,z)$ is the density function.

- If $\mathcal{I}_b$ is defined in a frame {b} at the center of mass, with axes aligned with the principal axes of inertia, then $\mathcal{I}_b$ is diagonal.

- If {b} is at the center of mass but its axes are not aligned with the principal axes of inertia, there always exists a rotated frame {c} defined by the rotation matrix R_{bc} such that $\mathcal{I}_c = R_{bc}^{\mathrm{T}}\mathcal{I}_b R_{bc}$ is diagonal.

- If $\mathcal{I}_b$ is defined in a frame {b} at the center of mass then $\mathcal{I}_q$, the inertia in a frame {q} aligned with {b} but displaced from the origin of {b} by $q \in \mathbb{R}^3$ in {b} coordinates, is

$$\mathcal{I}_q = \mathcal{I}_b + \mathfrak{m}(q^{\mathrm{T}}qI - qq^{\mathrm{T}}).$$

- The spatial inertia matrix $\mathcal{G}_b$ expressed in a frame {b} at the center of mass is defined as the 6×6 matrix

$$\mathcal{G}_b = \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & \mathfrak{m}I \end{bmatrix}.$$

  In a frame {a} at a configuration T_{ba} relative to {b}, the spatial inertia matrix is

$$\mathcal{G}_a = [\mathrm{Ad}_{T_{ba}}]^{\mathrm{T}}\mathcal{G}_b[\mathrm{Ad}_{T_{ba}}].$$

- The Lie bracket of two twists $\mathcal{V}_1$ and $\mathcal{V}_2$ is

$$\mathrm{ad}_{\mathcal{V}_1}(\mathcal{V}_2) = [\mathrm{ad}_{\mathcal{V}_1}]\mathcal{V}_2,$$

  where

$$[\mathrm{ad}_{\mathcal{V}}] = \begin{bmatrix} [\omega] & 0 \\ [v] & [\omega] \end{bmatrix} \in \mathbb{R}^{6\times 6}.$$

- The twist–wrench formulation of the rigid-body dynamics of a single rigid body is

$$\mathcal{F}_b = \mathcal{G}_b\dot{\mathcal{V}}_b - [\mathrm{ad}_{\mathcal{V}_b}]^{\mathrm{T}}\mathcal{G}_b\mathcal{V}_b.$$

  The equations have the same form if $\mathcal{F}$, $\mathcal{V}$, and $\mathcal{G}$ are all expressed in the same frame, regardless of the frame.

- The kinetic energy of a rigid body is $\frac{1}{2}\mathcal{V}_b^{\mathrm{T}}\mathcal{G}_b\mathcal{V}_b$, and the kinetic energy of an open-chain robot is $\frac{1}{2}\dot{\theta}^{\mathrm{T}}M(\theta)\dot{\theta}$.

- The forward–backward Newton–Euler inverse dynamics algorithm is the following:

  Initialization: Attach a frame $\{0\}$ to the base, frames $\{1\}$ to $\{n\}$ to the centers of mass of links $\{1\}$ to $\{n\}$, and a frame $\{n+1\}$ at the end-effector, fixed in the frame $\{n\}$. Define $M_{i,i-1}$ to be the configuration of $\{i-1\}$ in $\{i\}$ when $\theta_i = 0$. Let $\mathcal{A}_i$ be the screw axis of joint i expressed in $\{i\}$, and $\mathcal{G}_i$ be the 6×6 spatial inertia matrix of link i. Define $\mathcal{V}_0$ to be the twist of the base frame $\{0\}$ expressed in base-frame coordinates. (This quantity is typically zero.) Let $\mathfrak{g} \in \mathbb{R}^3$ be the gravity vector expressed in base-frame-$\{0\}$ coordinates, and define $\dot{\mathcal{V}}_0 = (0, -\mathfrak{g})$. (Gravity is treated as an acceleration of the base in the opposite direction.) Define $\mathcal{F}_{n+1} = \mathcal{F}_{\mathrm{tip}} = (m_{\mathrm{tip}}, f_{\mathrm{tip}})$ to be the wrench applied to the environment by the end-effector expressed in the end-effector frame $\{n+1\}$.

  Forward iterations: Given $\theta, \dot{\theta}, \ddot{\theta}$, for $i = 1$ to n do

$$\begin{aligned} T_{i,i-1} &= e^{-[\mathcal{A}_i]\theta_i} M_{i,i-1}, \\ \mathcal{V}_i &= \mathrm{Ad}_{T_{i,i-1}}(\mathcal{V}_{i-1}) + \mathcal{A}_i\dot{\theta}_i, \\ \dot{\mathcal{V}}_i &= \mathrm{Ad}_{T_{i,i-1}}(\dot{\mathcal{V}}_{i-1}) + \mathrm{ad}_{\mathcal{V}_i}(\mathcal{A}_i)\dot{\theta}_i + \mathcal{A}_i\ddot{\theta}_i. \end{aligned}$$

  Backward iterations: For $i = n$ to 1 do

$$\begin{aligned} \mathcal{F}_i &= \mathrm{Ad}^{\mathrm{T}}_{T_{i+1,i}}(\mathcal{F}_{i+1}) + \mathcal{G}_i\dot{\mathcal{V}}_i - \mathrm{ad}^{\mathrm{T}}_{\mathcal{V}_i}(\mathcal{G}_i\mathcal{V}_i), \\ \tau_i &= \mathcal{F}_i^{\mathrm{T}}\mathcal{A}_i. \end{aligned}$$

- Let $J_{ib}(\theta)$ be the Jacobian relating $\dot{\theta}$ to the body twist $\mathcal{V}_i$ in link i's center-of-mass frame $\{i\}$. Then the mass matrix $M(\theta)$ of the manipulator can be expressed as

$$M(\theta) = \sum_{i=1}^{n} J_{ib}^{\mathrm{T}}(\theta)\mathcal{G}_i J_{ib}(\theta).$$

- The forward dynamics problem involves solving

$$M(\theta)\ddot{\theta} = \tau(t) - h(\theta, \dot{\theta}) - J^{\mathrm{T}}(\theta)\mathcal{F}_{\mathrm{tip}}$$

  for $\ddot{\theta}$, using any efficient solver of equations of the form $Ax = b$.

- The robot's dynamics $M(\theta)\ddot{\theta} + h(\theta, \dot{\theta})$ can be expressed in the task space as
$$\mathcal{F} = \Lambda(\theta)\dot{\mathcal{V}} + \eta(\theta, \mathcal{V}),$$
where $\mathcal{F}$ is the wrench applied to the end-effector, $\mathcal{V}$ is the twist of the end-effector, and $\mathcal{F}$, $\mathcal{V}$, and the Jacobian $J(\theta)$ are all defined in the same frame. The task-space mass matrix $\Lambda(\theta)$ and gravity and quadratic velocity forces $\eta(\theta, \mathcal{V})$ are
$$\begin{aligned} \Lambda(\theta) &= J^{-T} M(\theta) J^{-1}, \\ \eta(\theta, \mathcal{V}) &= J^{-T} h(\theta, J^{-1}\mathcal{V}) - \Lambda(\theta)\dot{J}J^{-1}\mathcal{V}. \end{aligned}$$

- Define two $n \times n$ projection matrices of rank $n - k$
$$\begin{aligned} P(\theta) &= I - A^{\mathrm{T}}(AM^{-1}A^{\mathrm{T}})^{-1}AM^{-1}, \\ P_{\ddot{\theta}}(\theta) &= M^{-1}PM = I - M^{-1}A^{\mathrm{T}}(AM^{-1}A^{\mathrm{T}})^{-1}A, \end{aligned}$$
corresponding to the k Pfaffian constraints, $A(\theta)\dot{\theta} = 0$, $A \in \mathbb{R}^{k \times n}$, acting on the robot. Then the $n + k$ constrained equations of motion
$$\begin{aligned} \tau &= M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}) + A^{\mathrm{T}}(\theta)\lambda, \\ A(\theta)\dot{\theta} &= 0 \end{aligned}$$
can be reduced to the following equivalent forms by eliminating the Lagrange multipliers λ:
$$\begin{aligned} P\tau &= P(M\ddot{\theta} + h), \\ P_{\ddot{\theta}}\ddot{\theta} &= P_{\ddot{\theta}}M^{-1}(\tau - h). \end{aligned}$$
The matrix P projects away the joint force–torque components that act on the constraints without doing work on the robot, and the matrix $P_{\ddot{\theta}}$ projects away acceleration components that do not satisfy the constraints.

- An ideal gearhead (one that is 100% efficient) with a gear ratio G multiplies the torque at the output of a motor by a factor G and divides the speed by the factor G, leaving the mechanical power unchanged. The inertia of the motor's rotor about its axis of rotation, as it appears at the output of the gearhead, is $G^2\mathcal{I}_{\text{rotor}}$.

8.11 Software

Software functions associated with this chapter are listed below.

`adV = ad(V)`
Computes $[\mathrm{ad}_\mathcal{V}]$.

`taulist = InverseDynamics(thetalist,dthetalist,ddthetalist,g,Ftip,Mlist,Glist,Slist)`
Uses Newton–Euler inverse dynamics to compute the n-vector τ of the required joint forces–torques given θ, $\dot\theta$, $\ddot\theta$, $\mathfrak{g}$, $\mathcal{F}_{\text{tip}}$, a list of transforms $M_{i-1,i}$ specifying the configuration of the center-of-mass frame of link $\{i\}$ relative to $\{i-1\}$ when the robot is at its home position, a list of link spatial inertia matrices $\mathcal{G}_i$, and a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame.

`M = MassMatrix(thetalist,Mlist,Glist,Slist)`
Computes the mass matrix $M(\theta)$ given the joint configuration θ, a list of transforms $M_{i-1,i}$, a list of link spatial inertia matrices $\mathcal{G}_i$, and a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame.

`c = VelQuadraticForces(thetalist,dthetalist,Mlist,Glist,Slist)`
Computes $c(\theta,\dot\theta)$ given the joint configuration θ, the joint velocities $\dot\theta$, a list of transforms $M_{i-1,i}$, a list of link spatial inertia matrices $\mathcal{G}_i$, and a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame.

`grav = GravityForces(thetalist,g,Mlist,Glist,Slist)`
Computes $g(\theta)$ given the joint configuration θ, the gravity vector $\mathfrak{g}$, a list of transforms $M_{i-1,i}$, a list of link spatial inertia matrices $\mathcal{G}_i$, and a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame.

`JTFtip = EndEffectorForces(thetalist,Ftip,Mlist,Glist,Slist)`
Computes $J^{\mathrm{T}}(\theta)\mathcal{F}_{\text{tip}}$ given the joint configuration θ, the wrench $\mathcal{F}_{\text{tip}}$ applied by the end-effector, a list of transforms $M_{i-1,i}$, a list of link spatial inertia matrices $\mathcal{G}_i$, and a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame.

`ddthetalist = ForwardDynamics(thetalist,dthetalist,taulist,g,Ftip,Mlist,Glist,Slist)`
Computes $\ddot\theta$ given the joint configuration θ, the joint velocities $\dot\theta$, the joint forces-torques τ, the gravity vector $\mathfrak{g}$, the wrench $\mathcal{F}_{\text{tip}}$ applied by the end-effector, a list of transforms $M_{i-1,i}$, a list of link spatial inertia matrices $\mathcal{G}_i$, and a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame.

`[thetalistNext,dthetalistNext] = EulerStep(thetalist,dthetalist,`

`ddthetalist,dt)`
Computes a first-order Euler approximation to $\{\theta(t+\delta t), \dot\theta(t+\delta t)\}$ given the joint configuration $\theta(t)$, the joint velocities $\dot\theta(t)$, the joint accelerations $\ddot\theta(t)$, and a timestep δt.

`taumat = InverseDynamicsTrajectory(thetamat,dthetamat,ddthetamat,`
`g,Ftipmat,Mlist,Glist,Slist)`
The variable `thetamat` is an $N \times n$ matrix of robot joint variables θ, where the ith row corresponds to the n-vector of joint variables $\theta(t)$ at time $t = (i-1)\delta t$, where δt is the timestep. The variables `dthetamat`, `ddthetamat`, and `Ftipmat` similarly represent $\dot\theta$, $\ddot\theta$, and $\mathcal{F}_{\text{tip}}$ as a function of time. Other inputs include the gravity vector $\mathfrak{g}$, a list of transforms $M_{i-1,i}$, a list of link spatial inertia matrices $\mathcal{G}_i$, and a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame. This function computes an $N \times n$ matrix `taumat` representing the joint forces-torques $\tau(t)$ required to generate the trajectory specified by $\theta(t)$ and $\mathcal{F}_{\text{tip}}(t)$. Note that it is not necessary to specify δt. The velocities $\dot\theta(t)$ and accelerations $\ddot\theta(t)$ should be consistent with $\theta(t)$.

`[thetamat,dthetamat] = ForwardDynamicsTrajectory(thetalist,`
`dthetalist,taumat,g,Ftipmat,Mlist,Glist,Slist,dt,intRes)`
This function numerically integrates the robot's equations of motion using Euler integration. The outputs are $N \times n$ matrices `thetamat` and `dthetamat`, where the ith rows correspond respectively to the n-vectors $\theta((i-1)\delta t)$ and $\dot\theta((i-1)\delta t)$. The inputs are the initial state $\theta(0)$, $\dot\theta(0)$, an $N \times n$ matrix of joint forces or torques $\tau(t)$, the gravity vector $\mathfrak{g}$, an $N \times n$ matrix of end-effector wrenches $\mathcal{F}_{\text{tip}}(t)$, a list of transforms $M_{i-1,i}$, a list of link spatial inertia matrices $\mathcal{G}_i$, a list of joint screw axes $\mathcal{S}_i$ expressed in the base frame, the timestep δt, and the number of integration steps to take during each timestep (a positive integer).

8.12 Notes and References

An accessible general reference on rigid-body dynamics that covers both the Newton–Euler and Lagrangian formulations is [51]. A more classical reference that covers a wide range of topics in dynamics is [193].

A recursive inverse dynamics algorithm for open chains using the classical screw-theoretic machinery of twists and wrenches was first formulated by Featherstone (the collection of twists, wrenches, and the corresponding analogues of accelerations, momentum, and inertias, are collectively referred to as spatial vector notation); this formulation, as well as more efficient extensions based on articulated body inertias, are described in [46, 47].

The recursive inverse dynamics algorithm presented in this chapter was first described in [133] and makes use of standard operators from the theory of Lie groups and Lie algebras. One of the important practical advantages of this approach is that analytic formulas can be derived for taking first- and higher-order derivatives of the dynamics. This has important consequences for dynamics-based motion optimization: the availability of analytic gradients can greatly improve the convergence and robustness for motion optimization algorithms. These and other related issues are explored in, e.g., [88].

The task-space formulation was first initiated by Khatib [74], who referred to it as the operational space formulation. Note that the task-space formulation involves taking time derivatives of the forward kinematics Jacobian, i.e., $\dot{J}(\theta)$. Using either the body or space Jacobian, $\dot{J}(\theta)$ can in fact be evaluated analytically; this is explored in one of the exercises at the end this chapter.

A brief history of the evolution of robot dynamics algorithms, as well as pointers to references on the more general subject of multibody system dynamics (of which robot dynamics can be considered a subfield) can be found in [48].

8.13 Exercises

Exercise 8.1 Derive the formulas given in Figure 8.5 for:
(a) a rectangular parallelpiped;
(b) a circular cylinder;
(c) an ellipsoid.

Exercise 8.2 Consider a cast iron dumbbell consisting of a cylinder connecting two solid spheres at either end of the cylinder. The density of the dumbbell is 7500 kg/m^3. The cylinder has a diameter of 4 cm and a length of 20 cm. Each sphere has a diameter of 20 cm.
(a) Find the approximate rotational inertia matrix $\mathcal{I}_b$ in a frame {b} at the center of mass with axes aligned with the principal axes of inertia of the dumbbell.
(b) Write down the spatial inertia matrix $\mathcal{G}_b$.

Exercise 8.3 Rigid-body dynamics in an arbitrary frame.
(a) Show that Equation (8.42) is a generalization of Steiner's theorem.
(b) Derive Equation (8.43).

Exercise 8.4 The 2R open-chain robot of Figure 8.14, referred to as a ro-

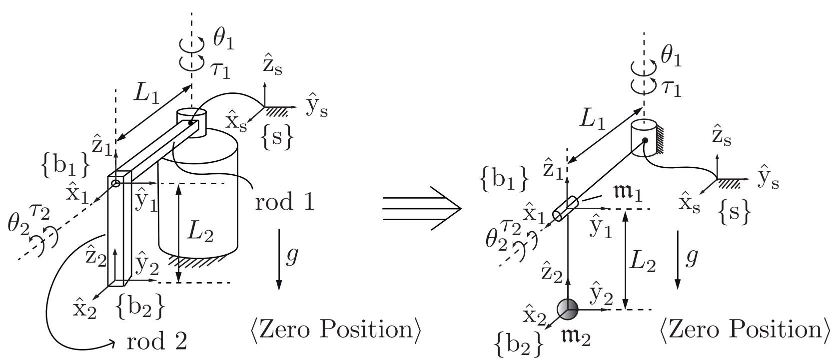


Figure 8.14: 2R rotational inverted pendulum. (Left) Its construction; (right) the model.

tational inverted pendulum or Furuta pendulum, is shown in its zero position. Assuming that the mass of each link is concentrated at the tip and neglecting its thickness, the robot can be modeled as shown in the right-hand figure. Assume that $\mathfrak{m}_1 = \mathfrak{m}_2 = 2$, $L_1 = L_2 = 1$, $g = 10$, and the link inertias $\mathcal{I}_1$ and $\mathcal{I}_2$ (expressed in their respective link frames $\{b_1\}$ and $\{b_2\}$) are

$$\mathcal{I}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}, \qquad \mathcal{I}_2 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

(a) Derive the dynamic equations and determine the input torques τ_1 and τ_2 when $\theta_1 = \theta_2 = \pi/4$ and the joint velocities and accelerations are all zero.
(b) Draw the torque ellipsoid for the mass matrix $M(\theta)$ when $\theta_1 = \theta_2 = \pi/4$.

Exercise 8.5 Prove the following Lie bracket identity (called the Jacobi identity) for arbitrary twists $\mathcal{V}_1$, $\mathcal{V}_2$, $\mathcal{V}_3$:

$$\mathrm{ad}_{\mathcal{V}_1}(\mathrm{ad}_{\mathcal{V}_2}(\mathcal{V}_3)) + \mathrm{ad}_{\mathcal{V}_3}(\mathrm{ad}_{\mathcal{V}_1}(\mathcal{V}_2)) + \mathrm{ad}_{\mathcal{V}_2}(\mathrm{ad}_{\mathcal{V}_3}(\mathcal{V}_1)) = 0.$$

Exercise 8.6 The evaluation of $\dot{J}(\theta)$, the time derivative of the forward kinematics Jacobian, is needed in the calculation of the frame accelerations $\dot{\mathcal{V}}_i$ in the Newton–Euler inverse dynamics algorithm and also in the formulation of the task-space dynamics. Letting $J_i(\theta)$ denote the ith column of $J(\theta)$, we have

$$\frac{d}{dt} J_i(\theta) = \sum_{j=1}^{n} \frac{\partial J_i}{\partial \theta_j} \dot{\theta}_j.$$

(a) Suppose that $J(\theta)$ is the space Jacobian. Show that

$$\frac{\partial J_i}{\partial \theta_j} = \begin{cases} \mathrm{ad}_{J_i}(J_j) & \text{for } i > j \\ 0 & \text{for } i \leq j. \end{cases}$$

(b) Now suppose that $J(\theta)$ is the body Jacobian. Show that

$$\frac{\partial J_i}{\partial \theta_j} = \begin{cases} \mathrm{ad}_{J_i}(J_j) & \text{for } i < j \\ 0 & \text{for } i \geq j. \end{cases}$$

Exercise 8.7 Show that the time derivative of the mass matrix $\dot{M}(\theta)$ can be written explicitly as

$$\dot{M} = \mathcal{A}^{\mathrm{T}}\mathcal{L}^{\mathrm{T}}\Gamma^{\mathrm{T}}[\mathrm{ad}_{\mathcal{A}\dot{\theta}}]^{\mathrm{T}}\mathcal{L}^{\mathrm{T}}\mathcal{G}\mathcal{L}\mathcal{A} + \mathcal{A}^{\mathrm{T}}\mathcal{L}^{\mathrm{T}}\mathcal{G}\mathcal{L}[\mathrm{ad}_{\mathcal{A}\dot{\theta}}]\Gamma\mathcal{A},$$

with the matrices as defined in the closed-form dynamics formulation.

Exercise 8.8 Explain intuitively the shapes of the end-effector force ellipsoids in Figure 8.4 on the basis of the point masses and the Jacobians.

Exercise 8.9 Consider a motor with rotor inertia $\mathcal{I}_{\mathrm{rotor}}$ connected through a gearhead of gear ratio G to a load with scalar inertia $\mathcal{I}_{\mathrm{link}}$ about the rotation axis. The load and motor are said to be **inertia matched** if, for any given torque τ_m at the motor, the acceleration of the load is maximized. The acceleration of the load can be written

$$\ddot{\theta} = \frac{G\tau_m}{\mathcal{I}_{\mathrm{link}} + G^2\mathcal{I}_{\mathrm{rotor}}}.$$

Solve for the inertia-matching gear ratio $\sqrt{\mathcal{I}_{\mathrm{link}}/\mathcal{I}_{\mathrm{rotor}}}$ by solving $d\ddot{\theta}/dG = 0$.

Exercise 8.10 Give the steps that rearrange Equation (8.98) to get Equation (8.100). Remember that $P(\theta)$ is not full rank and cannot be inverted.

Exercise 8.11 Program a function to calculate $h(\theta,\dot{\theta}) = c(\theta,\dot{\theta}) + g(\theta)$ efficiently using Newton–Euler inverse dynamics.

Exercise 8.12 Give the equations that would convert the joint and link descriptions in a robot's URDF file to the data `Mlist`, `Glist`, and `Slist`, suitable for using with the Newton–Euler algorithm `InverseDynamicsTrajectory`.

Exercise 8.13 The efficient evaluation of $M(\theta)$.

(a) Develop conceptually a computationally efficient algorithm for determining the mass matrix $M(\theta)$ using Equation (8.57).
(b) Implement this algorithm.

Exercise 8.14 The function `InverseDynamicsTrajectory` requires the user to enter not only a time sequence of joint variables `thetamat` but also a time sequence of joint velocities `dthetamat` and accelerations `ddthetamat`. Instead, the function could use numerical differencing to find approximately the joint velocities and accelerations at each timestep, using only `thetamat`. Write an alternative `InverseDynamicsTrajectory` function that does not require the user to enter `dthetamat` and `ddthetamat`. Verify that it yields similar results.

Exercise 8.15 Dynamics of the UR5 robot.
(a) Write the spatial inertia matrices $\mathcal{G}_i$ of the six links of the UR5, given the center-of-mass frames and mass and inertial properties defined in the URDF in Section 4.2.
(b) Simulate the UR5 falling under gravity with acceleration $g = 9.81$ m/s^2 in the $-\hat{\mathrm{z}}_{\mathrm{s}}$-direction. The robot starts at its zero configuration and zero joint torques are applied. Simulate the motion for three seconds, with at least 100 integration steps per second. (Ignore the effects of friction and the geared rotors.)