6. STATICS AND FORCE CONTROL

6.1 Introduction

We have assumed that the task to be performed by the robot requires that the position and velocity of the end-effector be controlled. This is known as position control (even though we are also controlling the velocity). A position controlled robot will ignore any forces resulting from contact between the end-effector and the environment. This is fine for such tasks as material handling or welding. However, there are tasks requiring control of these interaction forces. For example, if a robot is being used for grinding a weld bead, and the height of the weld bead is larger than expected, the grinding forces will become very large and could cause damage to the tool, part, and robot. Another example is assembly where parts must be carefully inserted. In this chapter we will study how to relate joint torques/forces to end-effector forces/torques and also examine various methods for force control.

6.2 Statics

With “statics” we analyse the forces and torques when the arm is stationary. If we revisit the general form of the dynamics equations with $\ddot{q} = \dot{q} = 0$ we find:

$$\begin{aligned}\tau &= M(q)[\ddot{q}] + B(q)[\dot{q}\dot{q}] + C(q)[\dot{q}^2] + G(q) \\ &= G(q)\end{aligned} \tag{6.1}$$

So the motor torques/forces are only required to balance the gravity terms. However, as mentioned in section 5.3, equation (6.1) does not consider contact forces/torques between the end-effector and the robot’s environment.

For now we will assume that G(q)=0 (this means either the robot operates in the horizontal plane or it is perfectly statically balanced). Recalling that work is a force acting through a distance, if we assume no losses occur between the joints and the end-effector, then [1]:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} \bullet \begin{bmatrix} dx \\ dy \\ dz \\ \delta x \\ \delta y \\ \delta z \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_n \end{bmatrix} \bullet \begin{bmatrix} dq_1 \\ dq_2 \\ \vdots \\ dq_n \end{bmatrix} \tag{6.2}$$

where $\bullet$ is the dot product, $[F_x \quad F_y \quad F_z \quad M_x \quad M_y \quad M_z]^T$ is the force/moment vector for the end-effector, $[\tau_1 \quad \tau_2 \quad \cdots \quad \tau_n]^T$ is the vector of joint torques, $[dq_1 \quad dq_2 \quad \cdots \quad dq_n]^T$ is the vector of differential changes in the joint variables and $[dx \quad dy \quad dz \quad \delta x \quad \delta y \quad \delta z]^T$ is the vector of differential changes in the end-effector’s position and orientation in Cartesian space. This can re-written as

$$F \bullet dC = \tau \bullet dq \tag{6.3}$$

or equivalently

$$F^T dC = \tau^T dq \tag{6.4}$$

Recall from section 3.2 that

$$dC = J(q)dq \tag{6.5}$$

Substituting (6.5) into (6.4), and simplifying, gives

$$F^T J(q) dq = \tau^T dq \tag{6.6}$$

$$F^T J(q) = \tau^T \tag{6.7}$$

Transposing both sides gives the final result

$$\tau = J(q)^T F \tag{6.8}$$

If G(q)≠0 then equation (6.8) should be replaced with

$$\tau = J(q)^T F + G(q) \tag{6.9}$$

Note: In (6.2) the robot was assumed to have all revolute joints. In general, τ is an n × 1 vector of joint torques (for each revolute joint) and/or joint forces (for each prismatic joint). For example, $\tau = \begin{bmatrix} F_1 \\ \tau_2 \end{bmatrix}$ for a PR robot.

Example 6.1

Given the robot is the planar 3R manipulator shown in Figure 6.1 and it operates in the horizontal plane. The parameters are: a_1=0.5 m, a_2=0.5 m and a_3=0.1 m.

a) Calculate the joint torques required to push a pin located at the position p_x=0.6 m and p_y=0; and oriented in the $-y_0$ direction with a force of 10 N.

From the pin's orientation we assume that the tool orientation should be $\phi = -90°$. For this orientation and the specified position, the inverse kinematics equations give the solution (elbow up): θ_1=62.0°, θ_2=–105.1° and θ_3=–46.9°.

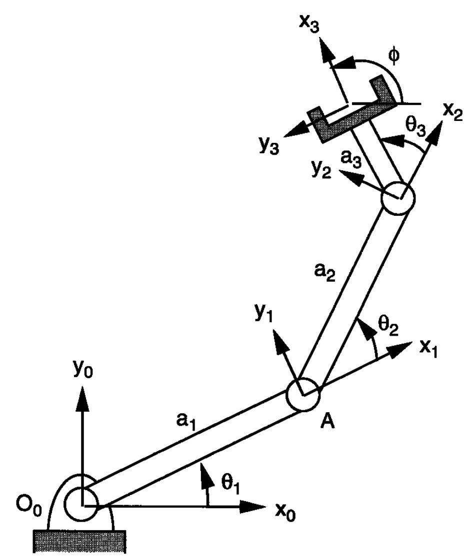


Figure 6.1 Planar 3R manipulator [2].

We can calculate the required joint torques by solving the problem in either world or tool coordinates.

If we use world coordinates then the force/moment vector is:

$$F_{world} = \begin{bmatrix} F_{x0} & F_{y0} & M_{z0} \end{bmatrix}^T = \begin{bmatrix} 0 & -10\text{ N} & 0 \end{bmatrix}^T \tag{6.10}$$

and the manipulator Jacobian we found in example 3.1 is used with equation (6.8) to calculate the joint torques as follows:

$$\tau = J(q)^T F_{world}$$

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} -a_1 S\theta_1 - a_2 S\theta_{12} - a_3 S\theta_{123} & a_1 C\theta_1 + a_2 C\theta_{12} + a_3 C\theta_{123} & 1 \\ -a_2 S\theta_{12} - a_3 S\theta_{123} & a_2 C\theta_{12} + a_3 C\theta_{123} & 1 \\ -a_3 S\theta_{123} & a_3 C\theta_{123} & 1 \end{bmatrix} \begin{bmatrix} F_{x0} \\ F_{y0} \\ M_{z0} \end{bmatrix} \tag{6.11}$$

$$= \begin{bmatrix} 0 & 0.6 & 1 \\ 0.4415 & 0.3652 & 1 \\ 0.1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -10 \\ 0 \end{bmatrix} = \begin{bmatrix} -6.0 \\ -3.652 \\ 0 \end{bmatrix} \text{Nm}$$

If we use tool coordinates, since we have chosen $\phi=-90°$ the force should be in the $+x_3$ direction. We then use the Jacobian for the tool frame we found in example 3.3 with equation (6.8) to calculate the torques:

$$\tau = {}^3J(q)^T F_{tool}$$

$$\begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix} = \begin{bmatrix} a_2S\theta_3 + a_1S\theta_{23} & a_3 + a_2C\theta_3 + a_1C\theta_{23} & 1 \\ a_2S\theta_3 & a_3 + a_2C\theta_3 & 1 \\ 0 & a_3 & 1\end{bmatrix}\begin{bmatrix}F_{x3}\\ F_{y3}\\ M_{z3}\end{bmatrix} \tag{6.12}$$

$$= \begin{bmatrix} -0.6 & 0.0 & 1 \\ -0.3652 & 0.4415 & 1 \\ 0 & 0.1 & 1\end{bmatrix}\begin{bmatrix}10\\ 0\\ 0\end{bmatrix} = \begin{bmatrix}-6.0\\ -3.652\\ 0\end{bmatrix}\text{Nm}$$

Of course the final result for the joint torques using both solution methods is the same. The choice of whether to work in world or tool coordinates depends on the particular task.

b) It is interesting to see how the torques vary as a function of the arm configuration and the direction of the applied force. We will look at two related examples. First, we will move the arm slowly in the x_0 direction while applying a force of 10 N in the y_0 direction (*i.e.* pushing upwards). The position and orientation of the tool will otherwise be fixed at $p_y=0$ and $\phi=0$. The corresponding joint torques are plotted vs. p_x in Figure 6.2. Examining this figure, it is clear that the joint torques increase as the arm is extended and the moment-arm of the applied force is increased. In fact, joint 1 is producing most of the applied force, while the torques at joints 2 and 3 prevent the arm from folding.

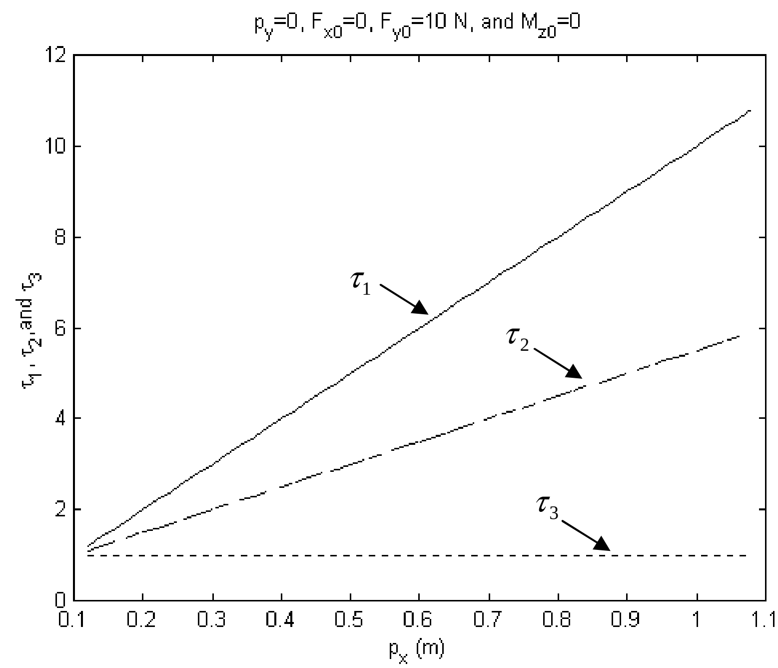


Figure 6.2 Joint torques for the arm pushing upwards as it is slowly extended in the x_0 direction.

Second, we will move the arm as before but now the applied force will be in the x_0 direction (*i.e.* pushing to the right). The corresponding joint torques are plotted vs. p_x in Figure 6.3. Note that τ_1 and τ_3 are always equal to zero. This occurs because the line of action of the force passes through joint 1 and joint 3. Joint 2 is left to do all of the work. **The graph shows that as the arm gets close to its maximum extension τ_2 decreases dramatically.**

Why is this happening?

What other mechanisms have similar behaviour?

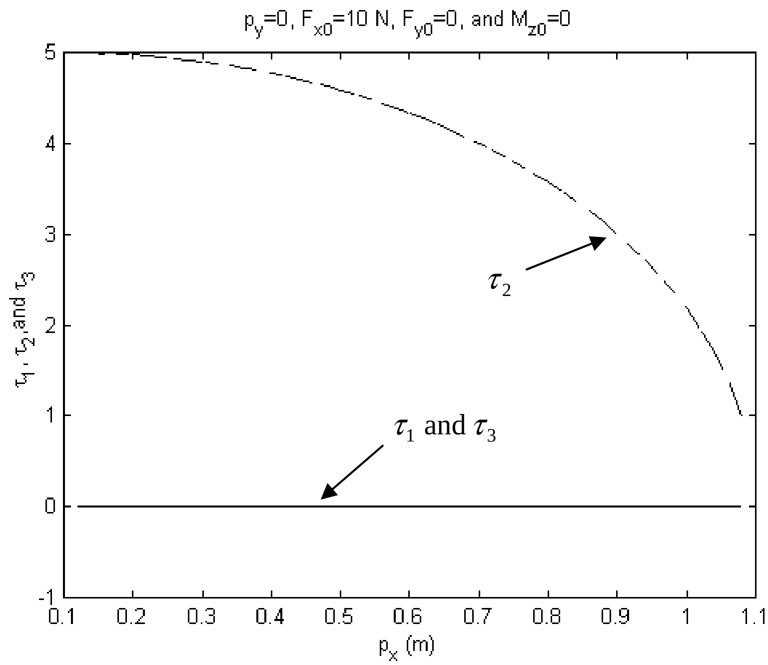


Figure 6.3 Joint torques for the robot arm pushing to the right as it is slowly extended in the x_0 direction.

End of example 6.1.

6.3 Statics Continued: Predicting End-Effector Forces/Torques from Joint Torques/Forces

We have seen how joint torques/forces can be calculated from given end-effector forces/torques. It is also possible to predict the end-effector forces/torques from a given vector of joint torques/forces using the following relationship:

$$\begin{aligned} \tau &= J(q)^T F + G(q) \\ \tau - G(q) &= J(q)^T F \\ \left(J(q)^T\right)^{-1}(\tau - G(q)) &= \left(J(q)^T\right)^{-1} J(q)^T F \\ F &= \left(J(q)^T\right)^{-1}(\tau - G(q)) \quad \text{or} \quad F = \left(J(q)^{-1}\right)^T(\tau - G(q)) \end{aligned} \tag{6.13}$$

Applications of equation (6.13) include: (i) monitoring the forces at the tool to prevent damage to the tool, environment or an object the tool is holding; and (ii) estimating the weight or mass of an object carried by the robot. Note that the robot cannot be at a singularity since the inverse of the Jacobian is required.

Example 6.2

We'll revisit the vertical 2R planar robot we studied in example 5.2. In the earlier example we found that the vector of gravity terms is

$$G(q) = \begin{bmatrix} (\tfrac{1}{2}m_1 + m_2)gl_1C\theta_1 + \tfrac{1}{2}m_2gl_2C\theta_{12} \\ \tfrac{1}{2}m_2gl_2C\theta_{12} \end{bmatrix} \tag{6.14}$$

From example 4.4, the inverse of the manipulator Jacobian for this robot is:

$$J^{-1}(q) = \frac{1}{l_1l_2S\theta_2}\begin{bmatrix} l_2C\theta_{12} & l_2S\theta_{12} \\ -l_1C\theta_1 - l_2C\theta_{12} & -l_1S\theta_1 - l_2S\theta_{12} \end{bmatrix} \tag{6.15}$$

The robot is carrying an object whose mass, m_3, is unknown. We wish to estimate the object mass for the purpose of quality control. We assume the object can be modelled as a concentrated mass located at the origin of the tool frame.

The robot has picked up the object and is holding it stationary at the configuration θ_1=60°, θ_2=-90°. The masses of the links are m_1=100kg and m_2=50kg. The link lengths are l_1=0.8m and l_2=0.8m. The measured joint torques are τ_1=616Nm and τ_2=204Nm.

We can find the forces at the end-effector by substituting G(q) and J(q) from (6.14) and (6.15), and the given information, into (6.13) as follows:

$$\begin{aligned} F &= \left(J(q)^{-1}\right)^T(\tau - G(q)) \\ \begin{bmatrix} F_x \\ F_y \end{bmatrix} &= \frac{1}{l_1l_2S\theta_2}\begin{bmatrix} l_2C\theta_{12} & -l_1C\theta_1 - l_2C\theta_{12} \\ l_2S\theta_{12} & -l_1S\theta_1 - l_2S\theta_{12} \end{bmatrix}\begin{bmatrix} \tau_1 - \left((\tfrac{1}{2}m_1 + m_2)gl_1C\theta_1 + \tfrac{1}{2}m_2gl_2C\theta_{12}\right) \\ \tau_2 - \tfrac{1}{2}m_2gl_2C\theta_{12} \end{bmatrix} \\ &= \frac{1}{-0.640}\begin{bmatrix} 0.693 & -1.093 \\ -0.4 & -0.293 \end{bmatrix}\begin{bmatrix} 616-562 \\ 204-170 \end{bmatrix} \\ &= \begin{bmatrix} 0.0 \\ 49.05 \end{bmatrix}\text{N} \end{aligned} \tag{6.16}$$

The estimated object mass in then:

$$m_3 = F_y/g = 49.05/9.81 = 5.0\text{ kg} \tag{6.17}$$

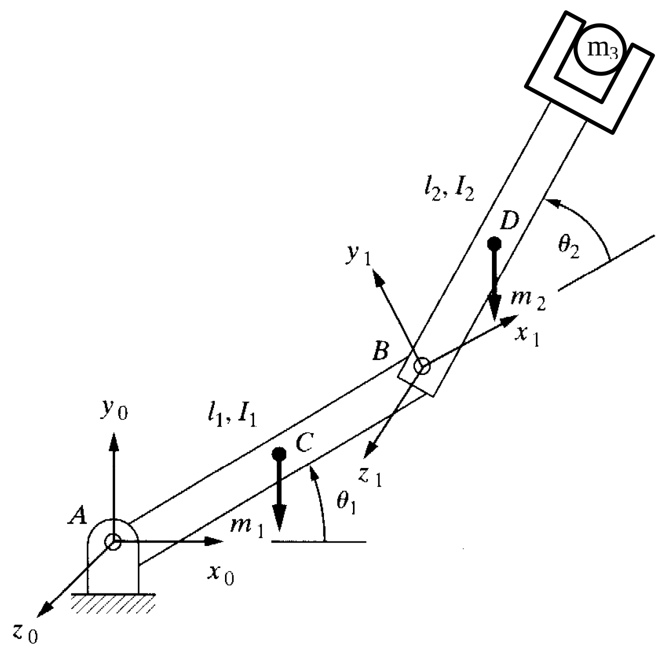


Figure 6.4 2R planar robot oriented in the vertical plane. The mass of the object (m_3) is unknown.

6.4 Methods for Force Control

In this section the term “force control” refers to the control of the forces/moments applied by the end-effector.

6.4.1 Using the Statics Equation and the Joint Torques/Forces

If we only require force control when the arm is stationary (or moving slowly) and don’t require precise force control then equation (6.9) may be used. The force control will be imprecise because gearbox friction makes it difficult to control the joint torque precisely. If DC motors are used the motor torque is proportional to the current so motor current control may be used as an open-loop joint torque control method. For improved performance, joint torque sensors and closed-loop control may be used. There are implementation difficulties with both methods. The robot’s controller may not allow access to the motor currents, and the design of the robot may make installation of joint torque sensors infeasible.

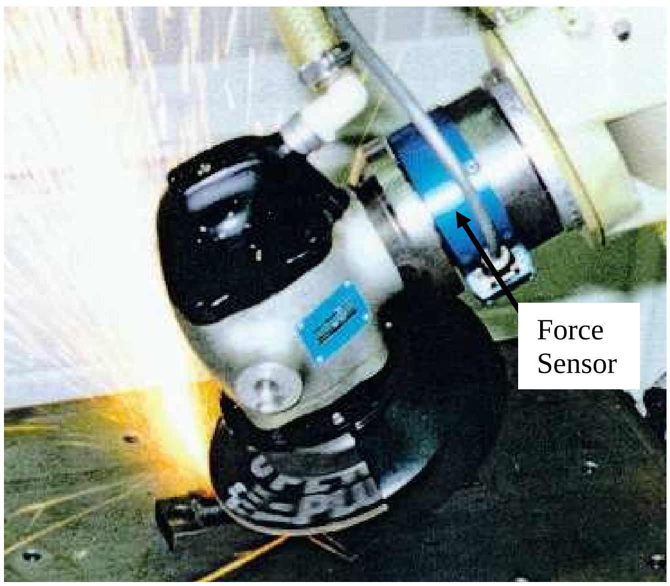


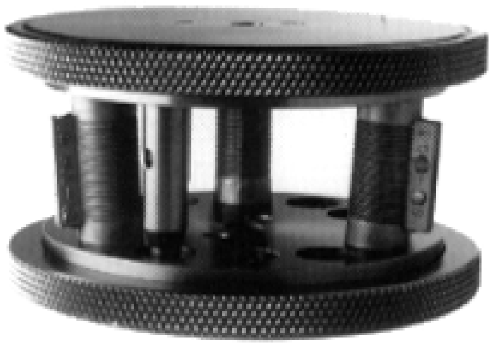
Figure 6.5 Active force controlled robotic grinding [3].

6.4.2 Active Force Control using a Force Sensor

The control methods described in section 6.4.1 are known as active force control methods since the control system includes some form of powered actuators, in this case the robot’s motors. Another active force control method requires the use of a force sensor mounted between the end of the robot arm and the end-effector. The measured forces are then used to modify the position and velocity setpoints for the robot’s position controller. For example, when grinding a weldbead the feedrate of the grinder can be adjusted to maintain a constant grinding force, resulting in higher quality and productivity. A photograph of active force controlled robotic grinding is shown in Figure 6.5. The advantage of this force sensor-based approach is that access to motor currents is not required. The main disadvantage is the high cost of the force sensor (≈ \$10,000).

6.4.3 Passive Force Control

When the force control does not require large or rapid changes to be made to the position or velocity of the end-effector, an approach known as “passive force control” may be used. Rather than using powered actuators, springs and dampers are used to provide passive compliance. (Compliance is the inverse of stiffness). The most successful device for this purpose is known an RCC (standing for Remote Centre of Compliance device). An RCC uses springs or elastomer pads to provide the desired amount of compliance. A common application of an RCC is to control the forces/moments during assembly insertion. The use of this passive force control, in combination with chamfers on the parts to be assembled, allows the insertion to be successful even when the parts are misaligned (*i.e.* avoids jamming). The use of an RCC for assembly insertion is illustrated in Figure 6.6. Using an RCC also reduces the assembly costs since the tolerances of the parts, fixtures and the robot can be less.



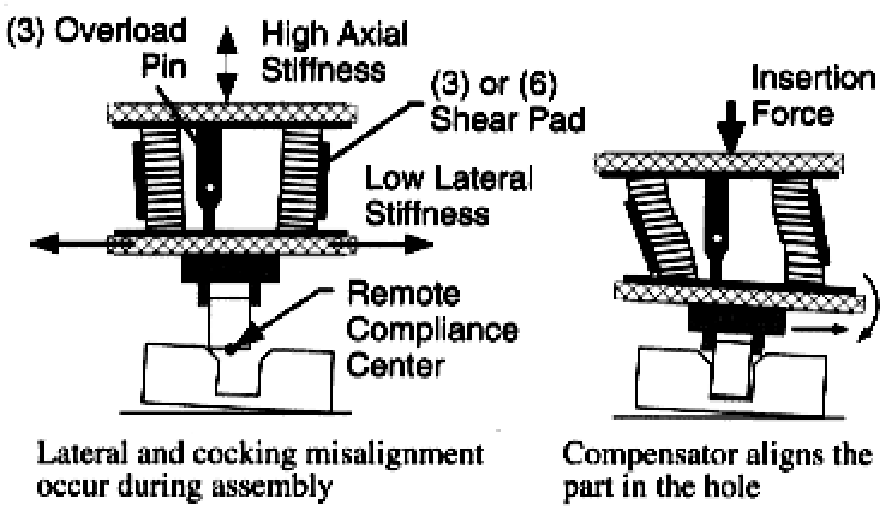


Figure 6.6 Top: An RCC from ATI Industrial Automation [4].
Bottom: Using an RCC to compensate for misalignments during assembly insertion (note that both rotation and translation are possible) [4].

References

1. J.J. Craig, "Introduction to Robotics", Addison Wesley Longman, 1989.
2. L.-W. Tsai, "Robot Analysis", John Wiley & Sons, 1999.
3. http://robotics.mcmaster.ca/Research/Deburring.htm
4. http://www.ati-ia.com