

3

Kinematic Analysis

The kinematic constraint equations corresponding to the natural coordinates were explained in detail in Chapter 2, both for planar and three-dimensional multibody systems. They were then compared to other types of coordinates. Attention was also given to the main sources of constraint equations with natural coordinates: rigid body constraints, joint constraints, and the optional definition of relative or joint coordinates.

In this chapter we will make use of those constraint equations to solve what is usually called kinematic problems, namely, the initial position or assembly problem, the finite displacement problem, and the velocity and acceleration analysis. The first two problems require an iterative solution of a system of nonlinear equations. Some special techniques to improve the convergence will be explained. Special attention will be addressed to the important case of *over constrained* multibody systems or, in general, to systems with non-independent constraint equations. The last section of this chapter is devoted to the case of *non-holonomic* joints.

3.1 Initial Position Problem

The initial position problem was defined in Section 1.3. It basically consists of determining the position of all the bodies in the system by knowing the positions of the fixed and the *input* bodies which can also be called *guided* or *driven* elements. Mathematically, the initial position problem is reduced to determining from the known coordinates corresponding to the input elements the vector of dependent coordinates that satisfies the nonlinear system of *constraint equations*. Note that the input can also be specified as the externally guided or driven linear or angular coordinates corresponding to several joints (as many joints as there are degrees of freedom) on which mixed coordinates have been defined. This basic notion is explained by means of the following examples:

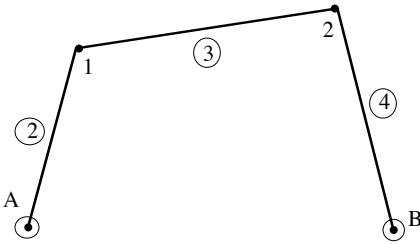


Figure 3.1. Four-bar mechanism modeled with natural coordinates.

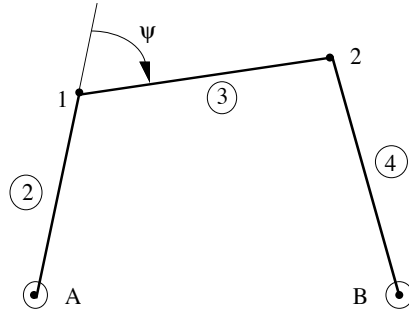


Figure 3.2. Four-bar mechanism modeled with mixed coordinates.

Example 3.1

As a first example, the four-bar mechanism of Figure 3.1 will be considered. This system has four natural coordinates (x_1, y_1, x_2, y_2) .

The constraint equations corresponding to this mechanism are following constant distance conditions:

$$(x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 = 0$$

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 = 0$$

$$(x_2 - x_B)^2 + (y_2 - y_B)^2 - L_4^2 = 0$$

These three equations are not sufficient to determine the four unknown variables of the problem. In fact, it is still necessary to enter the condition that the position of the input element (element 2) is known. If both coordinates of point 1 are known, then only two unknown variables are left. In this case, it is obvious that the first constraint equation which establishes the constant length condition of element 2 no longer makes any sense, because it does not contain any unknown variable. Consequently the problem reduces to the finding of x_2 and y_2 , using the last two nonlinear quadratic constraint conditions.

Example 3.2

Let us consider the four-bar mechanism shown in Figure 3.2, which uses mixed coordinates; that is, the same coordinates as in example 3.1 plus the angle ψ between elements 2 and 3 at joint 1. Let's assume that, instead of the position of the input element, one knows the angle ψ . In this case the constraint equations will be as follows (assuming a suitable value for ψ to be able to use the scalar product):

$$(x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 = 0$$

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 = 0$$

$$(x_2 - x_B)^2 + (y_2 - y_B)^2 - L_4^2 = 0$$

$$(x_1 - x_A)(x_2 - x_1) + (y_1 - y_A)(y_2 - y_1) - L_2 L_3 \cos \psi = 0$$

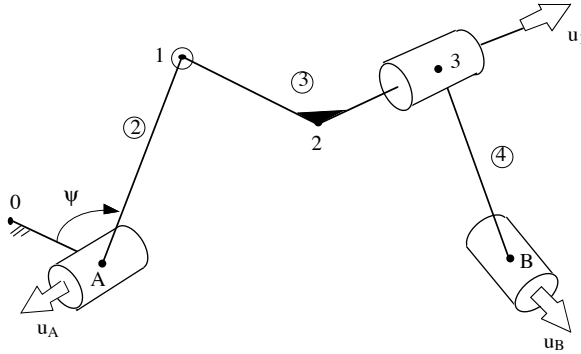


Figure 3.3. RSCR mechanism modeled with natural coordinates.

which is a system with four equations and four unknown variables, assuming that the externally driven angle ψ is known.

Example 3.3

Figure 3.3 depicts a three-dimensional four-bar mechanism *RSCR* (Revolute-Spherical-Cylindrical-Revolute) modeled with natural coordinates. This mechanism has three movable points and one movable unit vector; that is, twelve dependent Cartesian coordinates and one degree of freedom. Also the input angle ψ has been introduced as an additional externally driven coordinate. The constraint equations corresponding to this mechanism are the following:

1. $(x_1 - x_A)(x_0 - x_A) + (y_1 - y_A)(y_0 - y_A) + (z_1 - z_A)(z_0 - z_A) - k_1 \cos \psi = 0$
2. $(x_1 - x_A)^2 + (y_1 - y_A)^2 + (z_1 - z_A)^2 - k_2 = 0$
3. $(x_1 - x_A) u_{Ax} + (y_1 - y_A) u_{Ay} + (z_1 - z_A) u_{Az} - k_3 = 0$
4. $(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 - k_4 = 0$
5. $(x_2 - x_1) u_{1x} + (y_2 - y_1) u_{1y} + (z_2 - z_1) u_{1z} - k_5 = 0$
6. $u_{1x}^2 + u_{1y}^2 + u_{1z}^2 - 1 = 0$
7. $(x_3 - x_B)^2 + (y_3 - y_B)^2 + (z_3 - z_B)^2 - k_6 = 0$
8. $(x_3 - x_B) u_{1x} + (y_3 - y_B) u_{1y} + (z_3 - z_B) u_{1z} - k_7 = 0$
9. $(x_3 - x_B) u_{Bx} + (y_3 - y_B) u_{By} + (z_3 - z_B) u_{Bz} - k_8 = 0$
10. $u_{Bx} u_{1x} + u_{By} u_{1y} + u_{Bz} u_{1z} - k_9 = 0$
11. $(y_3 - y_2) u_{1z} - (z_3 - z_2) u_{1y} = 0$
12. $(z_3 - z_2) u_{1x} - (x_3 - x_2) u_{1z} = 0$
13. $(x_3 - x_2) u_{1y} - (y_3 - y_2) u_{1x} = 0$

This is the system of nonlinear equations that governs the position problem for the *RSCR* mechanism. The first equation corresponds to the input angle definition; equa-

tions 2 and 3 represent rigid body condition for element 2; equations 4 to 6 represent rigid body constraints for element 3; equations 6 to 10 represent the same for element 4, and equations 11 to 13 (only two of them are independent) contribute to define the cylindrical joint. Finally, k_i ($i=1, \dots, 9$) represents constant values.

The above examples clearly indicate that irrespective of the multibody systems being considered, the position problem is always based on solving the constraint equations, which make up the following set of equations:

$$\Phi(\mathbf{q}, t) = 0 \quad (3.1)$$

where \mathbf{q} is the vector of the system dependent coordinates. It will be assumed that there are at least as many equations as there are unknown variables or coordinates. To solve systems of nonlinear equations such as (3.1), it is customary to resort to the well-known Newton-Raphson method which has quadratic convergence in the neighborhood of the solution (the error in each iteration is proportional to the square of the error in the previous iteration) and does not usually cause serious difficulties if one starts with a good initial approximation.

The Newton-Raphson method is based on a linearization of the system (3.1) and consists in replacing this system of equations with the first two terms of its expansion in a Taylor series around a certain approximation \mathbf{q}_i to the desired solution. Once the substitution has been made, the system (3.1) becomes

$$\Phi(\mathbf{q}, t) \equiv \Phi(\mathbf{q}_i) + \Phi_{\mathbf{q}}(\mathbf{q}_i) (\mathbf{q} - \mathbf{q}_i) = 0 \quad (3.2)$$

where the time variable has not been accounted for, so that in this problem has a constant value. Matrix $\Phi_{\mathbf{q}}$ is the Jacobian matrix for constraint equations; that is to say, the matrix of partial derivatives of these equations with respect to the dependent coordinates. This matrix takes the following form:

$$\Phi_{\mathbf{q}} = \begin{bmatrix} \frac{\partial \phi_1}{\partial q_1} & \frac{\partial \phi_1}{\partial q_2} & \dots & \frac{\partial \phi_1}{\partial q_n} \\ \frac{\partial \phi_2}{\partial q_1} & \frac{\partial \phi_2}{\partial q_2} & \dots & \frac{\partial \phi_2}{\partial q_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial \phi_m}{\partial q_1} & \frac{\partial \phi_m}{\partial q_2} & \dots & \frac{\partial \phi_m}{\partial q_n} \end{bmatrix} \quad (3.3)$$

In equation (3.3), m is the number of constraint equations and n the number of dependent coordinates. If the constraint equations are independent, $f=n-m$ is the number of degrees of freedom of the multibody system.

Equation (3.2) represents a system of linear equations constituting an approximation to the nonlinear system (3.1). The vector \mathbf{q} , obtained from the solution of equation (3.2), will be an approximation of the solution in (3.1). By calling this approximate solution \mathbf{q}_{i+1} , a recursive formula is obtained as follows:

$$\Phi(\mathbf{q}_i) + \Phi_{\mathbf{q}}(\mathbf{q}_i) (\mathbf{q}_{i+1} - \mathbf{q}_i) = 0 \quad (3.4)$$

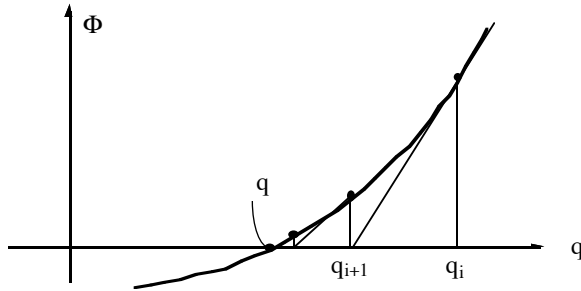


Figure 3.4. Iteration process of the Newton-Raphson method.

which can be used repeatedly until the error in the system of equations (3.1) is insignificant, or until the difference between the results of two successive iterations is smaller than a pre-specified tolerance. Figure 3.4 shows the geometric representation of the Newton-Raphson method for the case of a nonlinear equation with one unknown. The function $\Phi(\mathbf{q})$ is linearized at point \mathbf{q}_i , i.e. substituted by its tangent linear space, which are the first two terms of the Taylor expansion formula. The point where the tangent space intersects the horizontal axis is the approximate solution \mathbf{q}_{i+1} . The function $\Phi(\mathbf{q})$ is again replaced at point \mathbf{q}_{i+1} by the new tangent space and a new approximate solution \mathbf{q}_{i+2} is found. One arrives ultimately within the desired accuracy to the exact solution \mathbf{q} .

Note that the Newton-Raphson iteration will not always converge to a solution. It has been pointed out that if the initial approximation is not close enough to a solution, the algorithm may diverge. There is still another source of difficulties. If the values of the input variables do not correspond to a possible physical solution, the mathematical algorithm will fail irrespective of how the initial approximation has been chosen.

The Jacobian matrix of the constraint equations, defined by means of equation (3.3), plays an extremely important role in all kinematic and dynamic analysis problems. In the equation (3.4), the Jacobian matrix determines the linear equation system used to find the successive approximations for solving the initial position problem. Evaluating and triangularizing this matrix easily and quickly are characteristics of all good multibody system analysis methods. The natural coordinates permit the performance of these operations in the best possible way.

In Section 1.2, it was stated that the initial position problem had multiple solutions, and this is indeed the case. Depending on the vector \mathbf{q}_0 where the iteration begins, some solution will be attained.

Example 3.4

To complete this section on the initial position problem, the equations (3.4) corresponding to the four-bar mechanism studied in Examples 3.1 and 3.2 will be com-

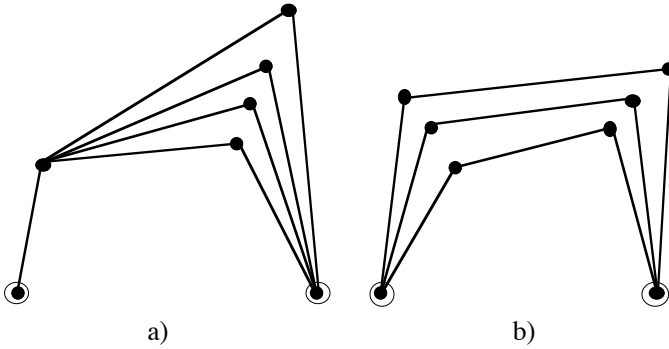


Figure 3.5. Iteration process of the Newton-Raphson method in a four-bar mechanism.

pletely developed. The constraint equations of this case were presented in Example 3.1, and consequently the equation (3.4) takes the following form:

$$2 \begin{bmatrix} (x_1 - x_A)(y_1 - y_A) & 0 & 0 \\ (x_1 - x_2)(y_1 - y_2) & (x_2 - x_1)(y_2 - y_1) \\ 0 & 0 & (x_2 - x_B)(y_2 - y_B) \end{bmatrix}_i \begin{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{pmatrix}_{i+1} - \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{pmatrix}_i \end{pmatrix} = - \begin{pmatrix} (x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 \\ (x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 \\ (x_2 - x_B)^2 + (y_2 - y_B)^2 - L_4^2 \end{pmatrix}_i$$

In this system of equations, at least one of the four unknown coordinates must be known ahead of time in order to be able to solve the problem. If, for example, x_1 is known, then:

$$(x_1)_{i+1} - (x_1)_i = 0$$

and the first column of the Jacobian matrix is multiplied by zero, meaning that it can be eliminated.

In the case of the four-bar mechanism of Figure 3.2, modeled with mixed coordinates and whose constraint equations are presented in Example 3.2, equation (3.4) becomes:

$$\begin{bmatrix} 2(x_1 - x_A) & 2(y_1 - y_A) & 0 & 0 & 0 \\ 2(x_1 - x_2) & 2(y_1 - y_2) & 2(x_2 - x_1) & 2(y_2 - y_1) & 0 \\ 0 & 0 & 2(x_2 - x_B) & 2(y_2 - y_B) & 0 \\ (x_2 - x_1 + x_A - x_1)(y_2 - y_1 + y_A - y_1) & (x_1 - x_A) & (y_1 - y_A) & (L_2 L_3 \sin \psi) \end{bmatrix}_i \cdot \begin{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{pmatrix}_{i+1} - \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{pmatrix}_i \end{pmatrix} = \begin{pmatrix} (x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 \\ (x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 \\ (x_2 - x_B)^2 + (y_2 - y_B)^2 - L_4^2 \\ (x_1 - x_A)(x_2 - x_1) + (y_1 - y_A)(y_2 - y_1) - L_2 L_3 \cos \psi \end{pmatrix}_i$$

Usually, the angle ψ is known; therefore, the last unknown variable ($\psi_{i+1} - \psi_i$) has a zero value. Thus the fifth column of the Jacobian matrix can be eliminated.

One characteristic common to the Jacobians matrices shown in this example (and in all the Jacobians matrices calculated with natural coordinates) is that they are linear functions of the dependent variables. For example, Figures 3.5a and 3.5b include drawings of the initial approximation, the first iterations, and the final solution of the initial position problem in the two four-bar mechanisms of Figures 3.1 and 3.2 computed according to the above expressions.

The Newton-Raphson method, explained in this section, converges rather quickly (quadratic convergence) when it is close to the desired solution. At times, and during the first iterations, it can give very abrupt jumps as a result of not having started from a sufficiently good initial approximation. Figure 3.6 shows what could happen in this case. The approximate solution \mathbf{q}_{i+1} is further away from the true solution \mathbf{q} , than the previous approximation \mathbf{q}_i . It is even possible that the value of function $\Phi(\mathbf{q})$, a function that should be equal to zero, could increase when moving from \mathbf{q}_i to \mathbf{q}_{i+1} .

This problem is not easy to solve without resorting to much more complicated numerical methods. In general, one should do everything possible to start from good initial approximations. If this cannot be achieved, then one should try to apply a reduction to the coordinates modification given by equation (3.4) and to apply it to the previous approximate solution \mathbf{q}_i . As this often works, a correction factor of 1/2 or 1/3 is recommended. Finally, one should always make sure that the module of $\Phi(\mathbf{q})$ decreases when going from point \mathbf{q}_i to \mathbf{q}_{i+1} .

Some authors have solved the position problem at times by calculating different solutions numerically by means of the so-called *continuation methods* (Tsai and Morgan (1985)). Continuation methods start out from a position where the multibody system complies with all the constraint equations, although the input elements might not be at the desired position and the fixed joints might not be at

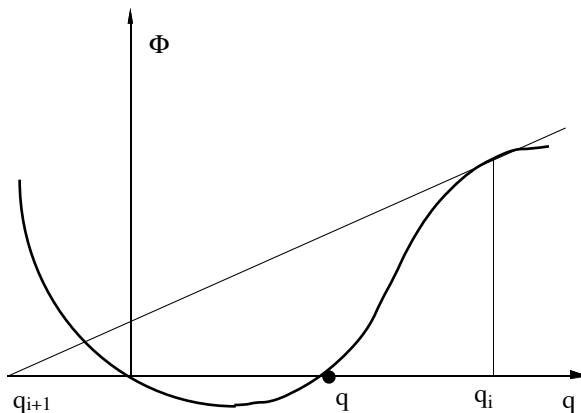


Figure 3.6. Possible divergence in the Newton-Raphson iteration

their final position. With relaxed conditions regarding the input elements and the fixed element, it is not difficult to find a position on the multibody system that satisfies the constraint equations. Then, by means of small finite displacements whose convergence is guaranteed by the Newton-Raphson method, an attempt is made to move the input elements and the fixed joints to their correct position. At times, the bifurcation points (points at which two or more possible movements can occur) provide a way of finding different solutions to the position problem.

3.2 Velocity and Acceleration Analysis

3.2.1 Velocity Analysis

The equations that permit solving the velocity problem originate after one differentiates with respect to time the constraint equations. If these equations are represented symbolically as

$$\Phi(\mathbf{q}, t) = 0 \quad (3.5)$$

by differentiating with respect to time, the following equation is obtained:

$$\Phi_{\mathbf{q}}(\mathbf{q}, t) \dot{\mathbf{q}} = -\Phi_t \equiv \mathbf{b} \quad (3.6)$$

where $\Phi_{\mathbf{q}}$ is the Jacobian matrix defined by means of equation (3.3). Vector $\dot{\mathbf{q}}$ is the vector of dependent velocities (derivative with respect to the time of the vector of dependent coordinates or position variables). Vector $(-\Phi_t = \mathbf{b})$ is the partial derivative of the constraint equations with respect to time. If all the constraints are scleronomous, meaning that there are no rheonomous or time dependent constraints, this derivative will be zero) If the position of the multibody system is known, equation (3.6) allows us to determine the velocities of the multibody system by starting from the velocity of the input elements. Just as in the position problem, the matrix that controls the velocity problem is the Jacobian matrix of the constraint equations. The essential difference between both problems is that where the position problem is nonlinear, the equations governing the velocity problem are linear. This means that the equations do not have to be iterated, and there is only one solution to a properly posed problem. The following example illustrates these concepts:

Example 3.5

As an example of this, the velocity equations of the four-bar mechanism of Figure 3.1 will be determined below by using: a) relative coordinates, b) reference point coordinates, c) natural coordinates, and d) mixed coordinates.

- a) Using *relative* coordinates, the constraint equations are given by (See Section 2.1.1),

$$L_1 \cos \Psi_1 + L_2 \cos (\Psi_1 + \Psi_2) + L_3 \cos (\Psi_1 + \Psi_2 + \Psi_3) - OD = 0$$

$$L_1 \sin \Psi_1 + L_2 \sin (\Psi_1 + \Psi_2) + L_3 \sin (\Psi_1 + \Psi_2 + \Psi_3) = 0$$

Differentiating these equations with respect to time, we obtain:

$$\begin{aligned} & -L_1 \sin \Psi_1 \dot{\Psi}_1 - L_2 \sin (\Psi_1 + \Psi_2) (\dot{\Psi}_1 + \dot{\Psi}_2) - \\ & - L_3 \sin (\Psi_1 + \Psi_2 + \Psi_3) (\dot{\Psi}_1 + \dot{\Psi}_2 + \dot{\Psi}_3) = 0 \\ & L_1 \cos \Psi_1 \dot{\Psi}_1 + L_2 \cos (\Psi_1 + \Psi_2) (\dot{\Psi}_1 + \dot{\Psi}_2) + \\ & + L_3 \cos (\Psi_1 + \Psi_2 + \Psi_3) (\dot{\Psi}_1 + \dot{\Psi}_2 + \dot{\Psi}_3) = 0 \end{aligned}$$

and by rearranging these equations, we arrive at:

$$\begin{bmatrix} -L_1 s_1 - L_2 s_{12} - L_3 s_{123} & -L_2 s_{12} - L_3 s_{123} & -L_3 s_{123} \\ L_1 c_1 + L_2 c_{12} + L_3 c_{123} & L_2 c_{12} + L_3 c_{123} & L_3 c_{123} \end{bmatrix} \begin{pmatrix} \dot{\Psi}_1 \\ \dot{\Psi}_2 \\ \dot{\Psi}_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

where $s_I = \sin \Psi_I$, $s_{I2} = \sin (\Psi_I + \Psi_2)$, and so forth.

If one of the three velocities in the previous equation is known such as the one corresponding to the input coordinate) the corresponding column of the Jacobian matrix can be moved to the right-hand side of the equation. This results in a system of two linear equations with two unknown velocities that can be solved with no difficulties.

b) Using *reference point* coordinates, the constraint equations are represented by (See Section 2.1.2):

$$\begin{aligned} (x_1 - x_0) - L_1/2 \cos \Psi_1 &= 0 \\ (y_1 - y_0) - L_1/2 \sin \Psi_1 &= 0 \\ (x_2 - x_1) - L_1/2 \cos \Psi_1 - L_2/2 \cos \Psi_2 &= 0 \\ (y_2 - y_1) - L_1/2 \sin \Psi_1 - L_2/2 \sin \Psi_2 &= 0 \\ (x_3 - x_2) - L_2/2 \cos \Psi_2 - L_3/2 \cos \Psi_3 &= 0 \\ (y_3 - y_2) - L_2/2 \sin \Psi_2 - L_3/2 \sin \Psi_3 &= 0 \\ (x_3 - x_D) - L_3/2 \cos \Psi_3 &= 0 \\ (y_3 - y_D) - L_3/2 \sin \Psi_3 &= 0 \end{aligned}$$

and the time derivatives are:

$$\begin{aligned} \dot{x}_1 + L_1/2 \dot{\Psi}_1 \sin \Psi_1 &= 0 \\ \dot{y}_1 - L_1/2 \dot{\Psi}_1 \cos \Psi_1 &= 0 \\ \dot{x}_2 - \dot{x}_1 + L_1/2 \dot{\Psi}_1 \sin \Psi_1 + L_2/2 \dot{\Psi}_2 \sin \Psi_2 &= 0 \\ \dot{y}_2 - \dot{y}_1 - L_1/2 \dot{\Psi}_1 \cos \Psi_1 - L_2/2 \dot{\Psi}_2 \cos \Psi_2 &= 0 \\ \dot{x}_3 - \dot{x}_2 + L_2/2 \dot{\Psi}_2 \sin \Psi_2 + L_3/2 \dot{\Psi}_3 \sin \Psi_3 &= 0 \\ \dot{y}_3 - \dot{y}_2 - L_2/2 \dot{\Psi}_2 \cos \Psi_2 - L_3/2 \dot{\Psi}_3 \cos \Psi_3 &= 0 \\ \dot{x}_3 + L_3/2 \dot{\Psi}_3 \sin \Psi_3 &= 0 \\ \dot{y}_3 - L_3/2 \dot{\Psi}_3 \cos \Psi_3 &= 0 \end{aligned}$$

These equations can be expressed in matrix form as follows:

$$\Phi_{\mathbf{q}} \dot{\mathbf{q}} = 0$$

where the matrix $\Phi_{\mathbf{q}}$ is

$$\begin{bmatrix} 1 & 0 & s_1 L_1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -c_1 L_1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & s_1 L_1/2 & 1 & 0 & s_2 L_2/2 & 0 & 0 & 0 \\ 0 & -1 & -c_1 L_1/2 & 0 & 1 & -c_2 L_2/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & s_2 L_2/2 & 1 & 0 & s_3 L_3/2 \\ 0 & 0 & 0 & 0 & -1 & -c_2 L_2/2 & 0 & 1 & -c_3 L_3/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & s_3 L_3/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -c_3 L_3/2 \end{bmatrix}$$

which is a system of eight equations with nine unknown velocities. If the angular velocity $\dot{\psi}_1$ is known for element 2, the third column of the Jacobian matrix will be moved to the right side member. The result will be a system of eight linear equations with eight unknown velocities.

c) With *natural* coordinates the constraint equations (Section 2.1.3) are represented by:

$$(x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 = 0$$

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 = 0$$

$$(x_3 - x_B)^2 + (y_3 - y_B)^2 - L_4^2 = 0$$

whose time derivatives are:

$$(x_1 - x_A) \dot{x}_1 + (y_1 - y_A) \dot{y}_1 = 0$$

$$(x_2 - x_1) (\dot{x}_2 - \dot{x}_1) + (y_2 - y_1) (\dot{y}_2 - \dot{y}_1) = 0$$

$$(x_2 - x_B) \dot{x}_2 + (y_2 - y_B) \dot{y}_2 = 0$$

and in matrix form yields:

$$\begin{bmatrix} (x_1 - x_A) & (y_1 - y_A) & 0 & 0 \\ (x_1 - x_2) & (y_1 - y_2) & (x_2 - x_1) & (y_2 - y_1) \\ 0 & 0 & (x_2 - x_B) & (y_2 - y_B) \end{bmatrix} \begin{Bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$

By knowing one of the four natural velocities and by moving the corresponding column to the right-hand side of this equation, one can find the remaining velocities with the resulting set of three linear equations and three unknown variables.

d) Using *mixed* coordinates, the constraint equations (Section 2.1.4) are:

$$(x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 = 0$$

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 = 0$$

$$(x_3 - x_B)^2 + (y_3 - y_B)^2 - L_4^2 = 0$$

$$(x_1 - x_A) (x_2 - x_1) + (y_1 - y_A) (y_2 - y_1) - L_2 L_3 \cos \Psi = 0$$

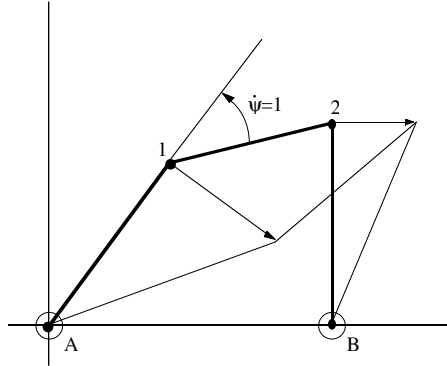


Figure 3.7. Results of a velocity analysis in a four-bar mechanism.

differentiating with respect to time:

$$(x_1-x_A) \dot{x}_1 + (y_1-y_A) \dot{y}_1 = 0$$

$$(x_2-x_1) (\dot{x}_2-\dot{x}_1) + (y_2-y_1) (\dot{y}_2-\dot{y}_1) = 0$$

$$(x_2-x_B) \dot{x}_2 + (y_2-y_B) \dot{y}_2 = 0$$

$$(x_2-x_1) \dot{x}_1 + (x_1-x_A) (\dot{x}_2-\dot{x}_1) + (y_2-y_1) \dot{y}_1 + (y_1-y_A) (\dot{y}_2-\dot{y}_1) + L_2 L_3 \sin\psi \dot{\psi} = 0$$

which can be expressed in matrix form as

$$\begin{bmatrix} (x_1-x_A) & (y_1-y_A) & 0 & 0 & 0 \\ (x_1-x_2) & (y_1-y_2) & (x_2-x_1) & (y_2-y_1) & 0 \\ 0 & 0 & (x_2-x_B) & (y_2-y_B) & 0 \\ (x_2-2x_1+x_A) & (y_2-2y_1+y_A) & (x_1-x_A) & (y_1-y_A) & L_2 L_3 \sin\psi \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

If $\dot{\psi}$ is known, the fifth column will be moved to the right-hand side and will leave four equations with four unknowns. Figure 3.7 shows the result of a velocity analysis in accordance with an input velocity of $\dot{\psi}=1$.

3.2.2 Acceleration Analysis

The finding of the dependent acceleration vector $\ddot{\mathbf{q}}$ becomes apparent by simply differentiating with respect to time the velocity equation (3.6). This yields the following result:

$$\Phi_{\mathbf{q}}(\mathbf{q}, t) \ddot{\mathbf{q}} = -\dot{\Phi}_{\mathbf{t}} - \dot{\Phi}_{\mathbf{q}} \dot{\mathbf{q}} \equiv \mathbf{c} \tag{3.7}$$

If the position vector \mathbf{q} and the velocity vector $\dot{\mathbf{q}}$ are known, by solving the system of linear equations (3.7), one can find the dependent acceleration vector $\ddot{\mathbf{q}}$. Note that the leading matrix of the systems of linear equations (3.6) and (3.7) is exactly the same. As a consequence, if it has been formed and triangularized

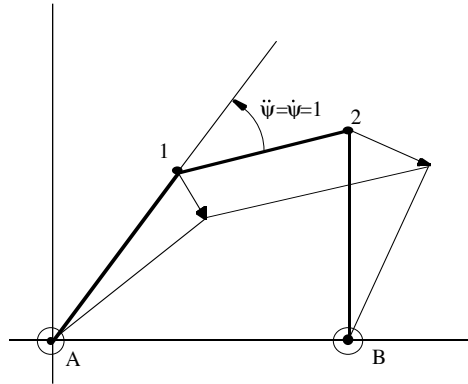


Figure 3.8. Results of an acceleration analysis in a four-bar mechanism.

to solve the velocity problem, the acceleration analysis can be carried out by simply forming the right-hand side and by performing a forward reduction and backward substitution. When there are no rheonomous or time-dependent constraints, the velocity problem is homogeneous; whereas the acceleration problem is always non-homogeneous as long as the velocities are not equal to zero.

Equation (3.7) can be differentiated once again to obtain the *jerk* or *over acceleration* equation:

$$\Phi_q \frac{d}{dt} (\ddot{\mathbf{q}}) = -\ddot{\Phi}_t - 2 \dot{\Phi}_q \dot{\mathbf{q}} - \ddot{\Phi}_q \dot{\mathbf{q}} \tag{3.8}$$

Once again a system of linear equations has been obtained whose leading matrix is the Jacobian matrix of the constraint equations.

Example 3.6

Included below are the acceleration equations for the four-bar mechanism of Example 3.5d modeled with mixed coordinates. These equations are obtained by differentiating the corresponding velocity equations:

$$\begin{bmatrix} (x_1-x_A) & (y_1-y_A) & 0 & 0 & 0 \\ (x_2-x_2) & (y_2-y_2) & (x_2-x_1) & (y_2-y_1) & 0 \\ 0 & 0 & (x_2-x_B) & (y_2-y_B) & 0 \\ (x_2-2x_1+x_A)(y_2-2y_1+y_A) & (x_1-x_A) & (y_1-y_A) & L_2 L_3 \sin\psi \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{y}_1 \\ \ddot{x}_2 \\ \ddot{y}_2 \\ \ddot{\psi} \end{bmatrix} = - \begin{bmatrix} \dot{x}_1 & \dot{y}_1 & 0 & 0 & 0 \\ (\dot{x}_1-\dot{x}_2) & (\dot{y}_1-\dot{y}_2) & (\dot{x}_2-\dot{x}_1) & (\dot{y}_2-\dot{y}_1) & 0 \\ 0 & 0 & \dot{x}_2 & \dot{y}_2 & 0 \\ (\dot{x}_2-2\dot{x}_1) & (\dot{y}_2-2\dot{y}_1) & \dot{x}_1 & \dot{y}_1 & L_2 L_3 \cos\psi \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{\psi} \end{bmatrix}$$

If all the velocities $\dot{\mathbf{q}}$ and input accelerations $\ddot{\boldsymbol{\psi}}$ are known, the remaining accelerations $\ddot{\mathbf{q}}$ can be calculated by means of the four equations with four unknowns resulting from moving the fifth column, multiplied by $\ddot{\boldsymbol{\psi}}$, to the RHS of the acceleration equation. Figure 3.8 graphically shows the result of an acceleration analysis that corresponds to the expression developed in this example.

3.3 Finite Displacement Analysis

The finite displacement analysis is closely related to the initial position problem, and is controlled by the same system of nonlinear equations (the kinematic constraint equations). The velocity and acceleration analyses are used at times in finite displacement analysis to improve the initial approximation with which the iterative process begins, which explains the reason for including it here and not immediately after the initial position problem.

3.3.1 Newton-Raphson Iteration

As explained in Section 1.2, once one knows a position of the multibody system where all the constraint equations are satisfied, the finite displacement problem consists of finding the new position that the system takes when a finite displacement is applied to each one of the input elements or externally driven relative coordinates. Finite displacement is understood to be any movement other than infinitesimal.

The main problem dealt with in this section is of the same nature and consequently controlled by the same equations of the position problem. Therefore, the Newton-Raphson method can be used for solving it. The difference between both problems lies in the fact that *the finite displacement problem usually relies on a good initial approximation* which is obtained from a previous exact position where all the elements satisfy the constraints. It is possible to improve upon the approximation by means of a velocity and acceleration analysis, as will be described in the next section.

These advantages do away with many of the convergence problems encountered in the initial position problem. In addition, the problem of multiple solutions becomes marginal. If the displacement of the input elements is small enough, then of all the possible solutions for the constraint equations, the correct one will be the closest to the starting position. This is precisely the one obtained by the Newton-Raphson iterations. However, there still remains the possibility of driving or trying to drive the multibody system to unfeasible positions, that is, positions that cannot be reached without violating some constraints equations. Trying to move the end-effector of a robot out of its workspace is an example of a finite displacement problem where the Newton-Raphson method will necessarily fail to find a correct solution.

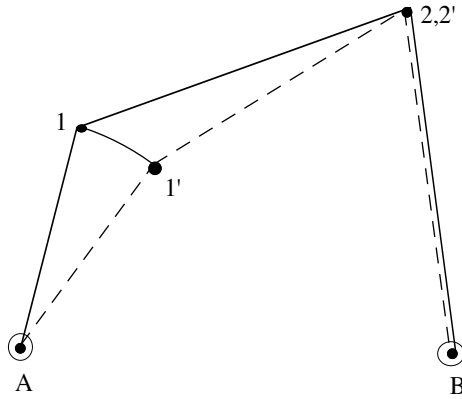


Figure 3.9. Improving the initial approximation.

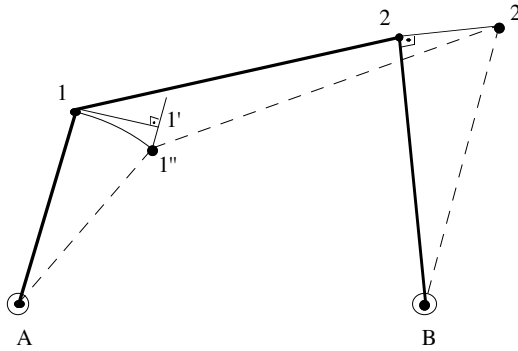


Figure 3.10. Better estimate for an initial approximation.

3.3.2 Improved Initial Approximation

In order to determine and improve the initial or starting approximation, the example of the four-bar mechanism will be used once again. This will clearly describe the method without any loss of generality.

Figure 3.9 shows a four-bar mechanism, in which the input element has been rotated a finite angle. One possible way of generating an initial approximation is by not varying the remaining natural coordinates as in the starting position shown in Figure 3.9. This approximation leads to a severe violation of the constraint equations.

The initial approximation shown in Figure 3.9 can be improved upon by means of *velocity analysis*, as indicated in Figure 3.10. The velocity analysis is carried out by imposing a velocity at the input element so that the endpoint $1'$ of the velocity vector of 1 is the closest point to $1''$ over the perpendicular to $A-1$ ($1'-1''$ is parallel to $A-1$).

Since $1''$ is known, it is not difficult to determine the velocity of the input element such that the end of the velocity vector at point 1 is $1'$. Using this velocity as input, a velocity analysis is performed, and the ends of the velocity vectors are determined for all the basic points of the mechanism (in Figure 3.10, $2'$ is the end of the velocity vector of point 2).

The initial approximation used to start the iterations for the Newton-Raphson method is indicated by the dotted lines in Figure 3.10. It is an improvement over the one in Figure 3.9. Note that the initial approximation is (A- $1''$ - $2'$ -B) and not (A- $1'$ - $2'$ -B). The exact position $1''$ of point 1 is known because it belongs to the input element and this exact position should be used.

It is not essential that point $1'$ be the closest one to $1''$ on the tangent to the trajectory of 1. Another simpler possibility for calculating point $1'$ and the velocity of the input element, is to assume that point 1 changes to position $1''$ in an arbitrary period of time such as 1 second. Next, calculate the angular velocity of the input element by dividing angle $1-1''$ (in radians) by the said amount of time where the quotient is the said angular velocity. The position at the initial approximation of any point P can be calculated by means of the following expression:

$$\mathbf{q} = \mathbf{q}_0 + \dot{\mathbf{q}} \Delta t \quad (3.9)$$

Equation (3.9) is an approximate integral of velocities starting from the previous position. An approximate integration which also causes the accelerations to intervene can be obtained in a similar manner:

$$\mathbf{q} = \mathbf{q}_0 + \dot{\mathbf{q}} \Delta t + \frac{1}{2} \ddot{\mathbf{q}} \Delta t^2 \quad (3.10)$$

This formula suggests that the initial approximation can be constructed starting from a velocity analysis and an acceleration analysis. To calculate the velocity and acceleration of the input element one may proceed as follows:

1. Apply one of the previously studied methods and determine the velocity of the input elements.
2. Knowing the initial and final position of the input elements and their velocity, determine the acceleration to be applied to them applying equation (3.10) to the input elements.

Determination of the initial approximation by means of velocity and acceleration analysis allows the iterations to begin with a *better approximation* to the final solution. The cost of an acceleration analysis is small if a velocity analysis has already been performed. The matrix for both systems of equations is the same, and one only needs to form it and triangularize it once. Based on the experience gained through numerical experiments performed by the authors, the initial approximation constructed with velocities and accelerations does not always give better results than the one determined from velocities only.

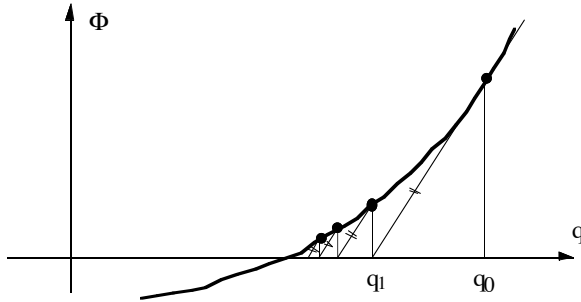


Figure 3.11. Iteration process of the Modified Newton-Raphson method.

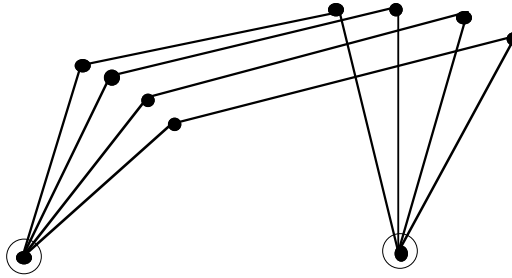


Figure 3.12. Kinematic simulation of a four-bar mechanism.

3.3.3 Modified Newton-Raphson Iteration

The Newton-Raphson method of solving systems of nonlinear equations proceeds as indicated in equation (3.4) and in Figure 3.4. It has already been mentioned that this iterative scheme has second order convergence in the neighborhood of the solution. The most important computational burden in the solution of equation (3.4) is the factorization of the Jacobian matrix.

The idea behind the modified Newton-Raphson method consists of applying the same iterative scheme but with a constant Jacobian matrix (See Figure 3.11),

$$\Phi(\mathbf{q}_i) + \Phi_{\mathbf{q}}(\mathbf{q}_0) (\mathbf{q}_{i+1} - \mathbf{q}_i) = 0 \tag{3.11}$$

The main advantage of the modified Newton-Raphson method is the reduced computational cost of each iteration. More iterations may be necessary to satisfy the convergence criterion, but in general the total CPU time can be reduced. However, if the motion increments (finite displacements of the input elements) are not small, the modified Newton-Raphson method is bound to have more convergence difficulties than the standard Newton-Raphson method. Sometimes, a mixed strategy such as a new Jacobian factorization every few iterations may give the best results.

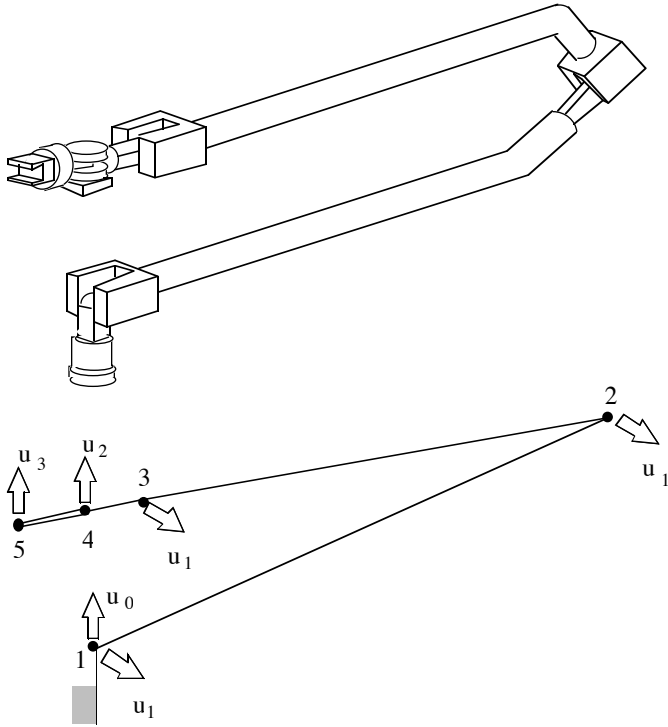


Figure 3.13. Spatial robotic manipulator modeled with natural coordinates.

Table 3.1. CPU time in seconds for 1000 finite displacement analyses of the robot in Figure 3.13.

(SNR) Standard Newton Raphson	(MNR) Modified Newton Raphson	MNR with improved initial approx.
53.1	21.7	15.5

3.3.4 Kinematic Simulation

Kinematic simulation is merely a repetition of the finite displacement problem, with the object of generating a sequence of positions that represent its movement in a specific time period or range of the input variables. This sequence of positions can later be depicted in animated form on the computer screen, as long as a system with sufficiently fast graphics is available.

The only problem with kinematic simulation is to find easy and general means of defining the movement of the input elements for the time interval in which one wants to simulate the motion of the multibody system. The increment

between two consecutive positions depends on the speed with which one wants to visualize the motion afterwards.

Kinematic simulation may simply involve the animated display of the system's motion, or it may also deal with the study of trajectories, possible collisions or geometrical interferences between solid models of the mechanism elements, and even the necessary driving forces and reactions that occur in a specific movement. This last problem is truly dynamic (an inverse dynamic problem) even though it may seem to be just a kinematic problem, since no dynamic differential equations need to be integrated.

All data previously stated for the finite displacement problem and the kinematic simulation is basically valid only when the displacements of the input elements are small. If the input displacements are very large, then it is desirable to split them into a series of smaller ones and to solve them sequentially. As an example Figure 3.12 shows the kinematic simulation of a four-bar mechanism where there are various consecutive positions of the system.

Example 3.7

Figure 3.13 shows a spatial 6R robot modeled with natural coordinates. It has four movable points and three movable vectors, with a total number of 21 dependent coordinates and six degrees of freedom. The kinematic simulation consists of imposing an end-effector translation on an elliptic path contained in a plane perpendicular to the robot initial position plane. One thousand finite increments in the end-effector position have been imposed. The corresponding CPU times on an HP 9000/834 computer (14 Mips and 1.8 DP Linpack Mflops) are shown in Table 3.1 for different conditions: standard Newton-Raphson method, modified Newton-Raphson method, and modified Newton-Raphson method with initial approximation obtained from a velocity analysis. In this case the improvements that result from using modified Newton-Raphson method and the velocity approximation are quite important. These figures could be modified for other computers according to the DP Linpack megaflops ratio.

3.4 Redundant Constraints

It has been seen in the previous sections of this chapter that the nonlinear kinematic constraint equations that govern the initial position or finite displacement problems can be formulated as,

$$\Phi(\mathbf{q}, t) = \mathbf{0} \quad (3.12)$$

In order to solve the previously mentioned kinematic problems departing from equation (3.12), such as initial position and finite displacement problems using Newton-Raphson iterations, or velocity and acceleration analysis, it is necessary to solve linear systems of equations in the form:

$$\Phi_{\mathbf{q}}(\mathbf{q}, t) \mathbf{x} = \mathbf{d} \quad (3.13)$$

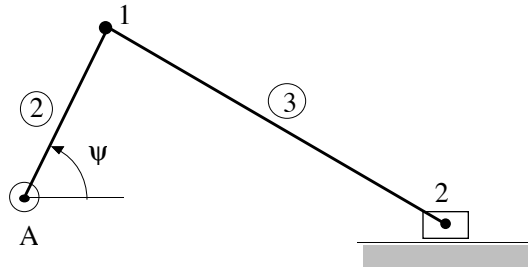


Figure 3.14. Slider-crank mechanism modeled with mixed coordinates.

The vector \mathbf{x} may represent incremental displacements, velocities, or accelerations, depending on the type of problem being analyzed. The corresponding RHS term is \mathbf{d} .

In practice, the multibody software developer and the engineer analyst very often face the problem of an excess of equations (more equations than the strictly necessary number) in (3.12), reflected by the fact that some of these equations are not independent from the remaining ones. This lack of equation independence in system (3.12) may lead to the following troubles in the solution of the linear system of equations (3.13):

- a) a rank deficiency in the Jacobian matrix $\Phi_{\mathbf{q}}$, if an inadequate subset of equations is chosen, and
- b) an over-constrained system of linear equations (more equations than unknowns) which will not have a solution that satisfy all the equations.

This section is addressed to consider the ways on which redundant equations appear in system (3.12), the consequences that this fact has on the system of linear equations (3.13) and the practical solutions or numerical strategies that can be followed to eliminate the resulting difficulties. Some very simple examples will be used to explain these points.

Multibody systems with n dependent coordinates and f degrees of freedom will be considered in the sequel. If the analyst is able to find $m=n-f$ independent constraint equations, no redundant constraints appear in the formulation and the standard formulations of previous sections in this chapter have full validity.

However, if $m>n-f$ consistent constraint equations are found, it is clear that there are $m-(n-f)$ redundant constraint equations. The following examples illustrate two possible origins for this situation.

Example 3.8

Figure 3.14 shows a planar slider-crank mechanism driven by the angle ψ . The constraint equations corresponding to the rigid body condition and the prismatic joint are

$$(x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 = 0 \quad (\text{i})$$

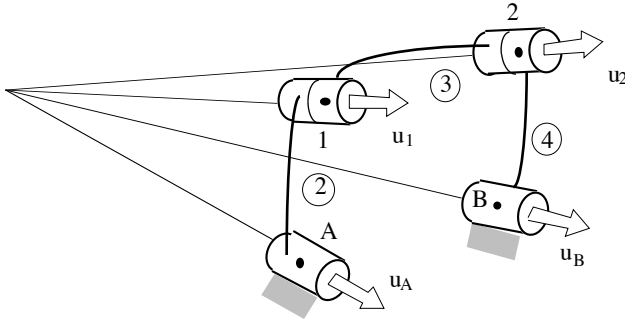


Figure 3.15. Spherical mechanism with natural coordinates.

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 = 0 \tag{ii}$$

$$y_2 = 0 \tag{iii}$$

It remains to formulate the constraint equation corresponding to the driven angle ψ . If angle ψ is near 90° or (-90°) the cosine equation that can be used is

$$(x_1 - x_A) - L_2 \cos \psi = 0 \tag{iv}$$

However, if angle ψ is near 0° or 180° , this equation is not valid (See Section 2.1.4) and the sine function that shall be used instead is

$$(y_1 - y_A) - L_2 \sin \psi = 0 \tag{v}$$

After these facts we arrive at the following situation. There are four dependent coordinates (four unknown ones (x_1, y_1, x_2, y_2) and one externally driven ψ) and four equations ((i), (ii), (iii), and (iv) or (v) depending on ψ value). If the user takes care of switching between equations (iv) and (v) according to the value of ψ , then the relation $m=n-f$ is always met and there is no problem. However, the user can decide to include always both equations (iv) and (v) with $m>n-f$, leaving to the equation solver the responsibility of disregarding the less appropriate equation in each position. Note that equations (i)-(v) constitute a system of nonlinear redundant but compatible equations: If equations (iv) and (v) are squared and added together, equation (i) is obtained.

Example 3.9

Consider the spherical four-bar mechanisms of Figure 3.15. It is well known that spherical mechanisms (all the revolute joint axes pass on a common point) are exceptions to the Grübler criterion, because they have more degrees of freedom than foreseen by the Grübler formula. In particular, the mechanism of Figure 3.15 has one degree of freedom, but the Grübler formula predicts (-2) . This is due to the particular orientation of joint axes. Note that for arbitrary joint orientations the Grübler result has full sense.

Consider this mechanism in terms of natural coordinates. There are two movable points and two movable unit vectors; hence there are $n=12$ dependent coordinates. The following constraint equations shall be formulated:

– Unit module conditions:

$$1.- \quad u_{1x}^2 + u_{1y}^2 + u_{1z}^2 - 1 = 0$$

$$2.- \quad u_{2x}^2 + u_{2y}^2 + u_{2z}^2 - 1 = 0$$

– Constant distance conditions:

$$3.- \quad (x_1 - x_A)^2 + (y_1 - y_A)^2 + (z_1 - z_A)^2 - L_2^2 = 0$$

$$4.- \quad (x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 - L_3^2 = 0$$

$$5.- \quad (x_2 - x_B)^2 + (y_2 - y_B)^2 + (z_2 - z_B)^2 - L_4^2 = 0$$

– Constant angle conditions:

$$6.- \quad (x_1 - x_A) u_{Ax} + (y_1 - y_A) u_{Ay} + (z_1 - z_A) u_{Az} - k_1 = 0$$

$$7.- \quad (x_2 - x_1) u_{1x} + (y_2 - y_1) u_{1y} + (z_2 - z_1) u_{1z} - k_2 = 0$$

$$8.- \quad (x_2 - x_B) u_{Bx} + (y_2 - y_B) u_{By} + (z_2 - z_B) u_{Bz} - k_3 = 0$$

– Linear combination conditions:

$$9., 10., 11.- \quad \mathbf{u}_1 - k_4 \mathbf{u}_A - k_5 (\mathbf{r}_1 - \mathbf{r}_A) = 0$$

$$12., 13., 14.- \quad \mathbf{u}_2 - k_6 \mathbf{u}_1 - k_7 (\mathbf{r}_2 - \mathbf{r}_1) = 0$$

$$15., 16., 17.- \quad \mathbf{u}_2 - k_8 \mathbf{u}_B - k_9 (\mathbf{r}_2 - \mathbf{r}_B) = 0$$

There are 17 equations. Taking into account that vectors \mathbf{u}_1 and \mathbf{u}_2 have unit length, only two of each three linear combination conditions are necessary. This gives a total number of 14 constraint equations on 12 dependent coordinates.

There is another way to arrive at the same result. Each element with two points and two unit vectors generates six rigid body constraint equations, including two unit module conditions. This gives a total number of 18 equations. If it is taken into account that vectors \mathbf{u}_A and \mathbf{u}_B are constant, that is no unit module conditions for them are necessary and the unit module condition for vectors \mathbf{u}_1 and \mathbf{u}_2 has been considered twice because they belong to two different elements, one arrives again to a total number of constraint equations $m=14$.

Then, this mechanism has 12 dependent coordinates, one degree of freedom, and 14 constraint equations, which gives an excess of three constraint equations, in accordance with the wrong prediction of Grübler criterion (-2 instead of 1). Of these 14 constraint equations, only 11 are independent.

The two previous examples demonstrate without any lack of generality the two ways from which redundant constraint equations arise:

- a) Due to convenience of implementation, as in Example 3.8.
- b) In over constrained multibody systems that are exceptions to the Grübler criterion, as in Example 3.9.

Once a system has been characterized using systems of redundant constraint equations, the search of solutions can follow two different avenues:

1. Systems of equations (3.12) and (3.13) can be preprocessed with the aim of determining and eliminating the dependent equations, to keep only $m=n-f$ independent constraint equations, and then to use the standard formulations of

Sections 3.1-3.3. The main disadvantage of this method is the need to repeat the dependent equation elimination process each time the multibody system changes its configuration or, in the case of Example 3.7, after large changes in the position of the multibody system. Thus this procedure is not suitable for real-time applications (there is no time to repeat the dependent equations elimination process) or even for interactive simulation.

2. The second possibility is to solve system (3.12) directly, with a procedure capable of directly tackling redundant constraints on a strictly standard form. This way will be explained next.

Let us assume that system (3.12) has m nonlinear equations, of which only $(n-f) < m$ are independent. As a consequence, one may be tempted to think that the redundant equations in (3.12) just produce an excess of compatible equations in the linear system (3.13). If this were true no particular difficulties would appear during the solution, because there are a lot of ways and numerical routines to solve linear systems of equations with an excess of compatible equations. However, the problem is a little more complicated than assumed previously.

The redundant but compatible nonlinear equations in system (3.12) can induce an excess of non-compatible linear equations in system (3.13). This does not happen in velocity or acceleration analysis, because in these cases the Jacobian matrix is evaluated in the exact position \mathbf{q} , a position in which all constraint equations (3.12) are satisfied.

However, in the initial position and finite displacement problems, the following Newton-Raphson iteration formula is used:

$$(\Phi_{\mathbf{q}})_i (\mathbf{q}_{i+1} - \mathbf{q}_i) = -(\Phi)_i \quad (3.14)$$

In this expression the Jacobian matrix $\Phi_{\mathbf{q}}$ is evaluated at an intermediate approximate position \mathbf{q}_i at which the constraint equations (3.12) are not fulfilled. This makes the linear system (3.14) over-constrained and non-compatible which does not have an exact solution that satisfies every equation. There are again two ways to circumvent this difficulty:

- a) Sometimes this problem can be solved using Gaussian elimination with column pivoting and row scaling. Then, as long as \mathbf{q}_i is approaching the true solution at which the constraint equations are fulfilled, the algorithm tends to disregard automatically the dependent equations. However, this procedure can not be considered in general sufficiently robust and reliable.
- b) A reliable algorithm to solve the redundant system of linear equations (3.14) is the least-square formulation (Strang (1980)). Let us consider the normal equations corresponding to system (3.14):

$$(\Phi_{\mathbf{q}}^T \Phi_{\mathbf{q}})_i (\mathbf{q}_{i+1} - \mathbf{q}_i) = -(\Phi_{\mathbf{q}}^T)_i (\Phi)_i \quad (3.15)$$

This algorithm converges on a very reliable way to the exact solution of all constraint equations. It has been gathered from numerous simulations that it allows large displacements in the input coordinates with fast and reliable convergence.

It can be argued that the solution of system (3.15) is less efficient than the solution of equation (3.14), mainly because the product $(\Phi_q^T \Phi_q)_i$ needs to be performed prior to the solution. However, practical experience has shown that even for non-redundant systems, equation (3.15) can be more efficient than its counterpart (3.14). In large multibody systems, matrix Φ_q tends to be very sparse, and then the product $(\Phi_q^T \Phi_q)_i$ can be carried out very efficiently. System (3.15), although perhaps less sparse than system (3.14), has the advantage of being symmetric with the possibility of saving storage and using simpler pivoting strategies.

3.5 Subspace of Allowable Motions

In kinematic problems, the motion of the input elements or driven degrees of freedom is already known. From this knowledge the motion of the remaining bodies (or elements) is calculated. In the case of the direct or forward dynamic problem, the motion of the input elements is not known, or at least, it is not known for all of them, and the motion is obtained as a solution to the dynamic differential equations. Before entering the study of the dynamic problems treated in Chapters 4 and 5, we will study in this section the *possible* or *allowable* motions that the multibody system may have in accordance with the constraint equations. The study of these possible motions and the methods of expressing them is a purely kinematic problem that has important implications in the formulation of the differential equations of motion. These allowable motions will be studied next, and it will be distinguished, in order to introduce the subject progressively, between scleronomous and rheonomous constraints.

We will see in this section that the actual velocity vector $\dot{\mathbf{q}}$ of a constrained multibody system is always a vector that belongs to a very particular vector space called the *space of allowable motions*. The term *motions* should actually be *velocities*. The study of this vector space and the ability to find a basis for it constitute very important points for both kinematics and dynamics multibody formulations. Many authors have been explicitly or implicitly referring to it. See for instance: Kamman and Huston (1984), Kim and Vanderploeg (1986), Many et al. (1985), Agrawal (1984), Kane and Levinson (1985), Ider and Amirouche (1988), Huston (1990), and others. However, we find that the concept of the *space of allowable motions* allows for a simpler and more general way to explain, on a unified background, many different ideas and formulations that have been introduced in the last years. This concept is also the key towards the understanding of the improved real time dynamic formulations that will be studied in Chapters 5 and 8.

3.5.1 Scleronomous Systems

Consider a system with m constraint equations that do not depend explicitly on the time variable t , n dependent coordinates, and $f=n-m$ degrees of freedom. The constraint equations only depend on the dependent coordinates vector \mathbf{q} , and can be formulated as

$$\Phi(\mathbf{q}) = 0 \quad (3.16)$$

The velocity and acceleration equations are obtained by differentiating (3.16) with respect to time:

$$\Phi_{\mathbf{q}}(\mathbf{q}) \dot{\mathbf{q}} = 0 \quad (3.17)$$

$$\Phi_{\mathbf{q}}(\mathbf{q}) \ddot{\mathbf{q}} = -\dot{\Phi}_{\mathbf{q}} \dot{\mathbf{q}} \quad (3.18)$$

Equation (3.17) indicates that the velocity vector $\dot{\mathbf{q}}$ of a multibody system, at a specific position, belongs to the *nullspace* of the Jacobian matrix $\Phi_{\mathbf{q}}$ of the constraint equations. The theory of linear systems of equations (Strang (1980)) establishes that if the matrix $\Phi_{\mathbf{q}}$ has m independent rows and n columns ($m+f=n$), (it is of rank m , because the rank is equal to the number of rows), then the nullspace of $\Phi_{\mathbf{q}}$ is the *subspace of the possible or allowable motions* (velocities), in the sense that any possible velocity vector (compatible with the constraint equations) must belong to this subspace. The dimension of the space of allowable motions is the number of degrees of freedom $f=n-m$ of the multibody system.

Example 3.10

Consider again the four-bar mechanism with four dependent coordinates and one degree of freedom of Figure 3.1; thus, it has three constraint equations corresponding to the three constant distance conditions. The Jacobian matrix of the constraint equations for this mechanism (as already shown in Example 3.1) is :

$$\Phi_{\mathbf{q}} = \begin{bmatrix} x_1-x_A & y_1-y_A & 0 & 0 \\ x_1-x_2 & y_1-y_2 & x_2-x_1 & y_2-y_1 \\ 0 & 0 & x_2-x_B & y_2-y_B \end{bmatrix}$$

By symbolically performing a Gaussian elimination of this matrix, it can easily be demonstrated that its nullspace is defined by the following column vector:

$$\mathbf{r} = \begin{pmatrix} (y_1-y_A) [(y_2-y_1)(x_2-x_B) - (x_2-x_1)(y_2-y_B)] \\ - (x_1-x_A) [(y_2-y_1)(x_2-x_B) - (x_2-x_1)(y_2-y_B)] \\ (y_2-y_B) [(y_1-y_2)(x_1-x_A) - (x_1-x_2)(y_1-y_A)] \\ (x_2-x_B) [(y_1-y_2)(x_1-x_A) - (x_1-x_2)(y_1-y_A)] \end{pmatrix}$$

Since in this case the nullspace has dimension 1, the vector \mathbf{r} completely defines this subspace. In other words, any possible velocity vector $\dot{\mathbf{q}}$ shall be equal to the vector \mathbf{r} multiplied by a specific factor.

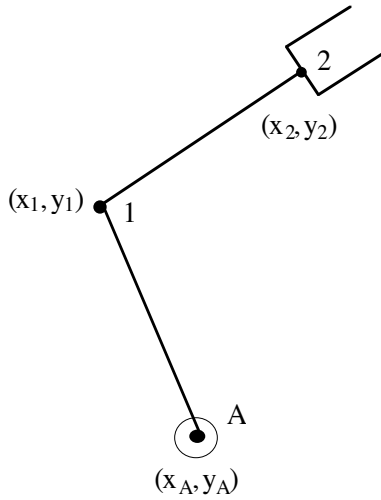


Figure 3.16. Planar robotic manipulator modeled with natural coordinates.

Example 3.11

Figure 3.16 shows a planar robot arm with four dependent natural coordinates and two degrees of freedom. There must be two independent constraint equations, which in this case are the corresponding constant distance equations. Their Jacobian matrix is similar to that of Example 3.10, except for the last row which is in this case eliminated,

$$\Phi_{\mathbf{q}} = \begin{bmatrix} x_1 - x_A & y_1 - y_A & 0 & 0 \\ x_1 - x_2 & y_1 - y_2 & x_2 - x_1 & y_2 - y_1 \end{bmatrix}$$

The velocity equation will be

$$\begin{bmatrix} x_1 - x_A & y_1 - y_A & 0 & 0 \\ x_1 - x_2 & y_1 - y_2 & x_2 - x_1 & y_2 - y_1 \end{bmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

In order to find a basis of the nullspace of this matrix, one may find two linearly independent vectors that belong to the said subspace.

This can be done as follows: by making $\dot{x}_2 = 1$, $\dot{y}_2 = 0$ the following vector is obtained:

$$\mathbf{r}^1 = \begin{pmatrix} \frac{-(x_2-x_1)(x_1-x_A)}{(y_1-y_2)(x_1-x_A)-(x_1-x_2)(y_1-y_A)} \\ \frac{(x_2-x_1)(y_1-y_A)}{(y_1-y_2)(x_1-x_A)-(x_1-x_2)(y_1-y_A)} \\ 1 \\ 0 \end{pmatrix}$$

Similarly, making $\dot{x}_2 = 0, \dot{y}_2 = 1$, the following vector is obtained:

$$\mathbf{r}^2 = \begin{pmatrix} \frac{(y_2-y_1)(y_1-y_A)}{(y_1-y_2)(x_1-x_A)-(x_1-x_2)(y_1-y_A)} \\ \frac{-(y_2-y_1)(x_1-x_A)}{(y_1-y_2)(x_1-x_A)-(x_1-x_2)(y_1-y_A)} \\ 0 \\ 1 \end{pmatrix}$$

It is evident that vectors \mathbf{r}^1 and \mathbf{r}^2 are independent in that one of them can never be obtained by multiplying the other by a constant. Therefore, a basis of the subspace of allowable motions can be formed. Any possible velocity vector of the mechanism in Figure 3.16 can be expressed as a linear combination of \mathbf{r}^1 and \mathbf{r}^2 as follows:

$$\dot{\mathbf{q}} = \mathbf{r}^1 \dot{z}_1 + \mathbf{r}^2 \dot{z}_2$$

where \dot{z}_1 and \dot{z}_2 are the coefficients of the linear combination, namely, *the independent velocities* of the mechanism.

An attempt will be made further on to generalize all that stated in the previous examples. The vector $\dot{\mathbf{q}}$ characterizes the velocity of the system with n dependent coordinates. To represent the velocity of the multibody system with a lower number of variables, one should construct a basis for the nullspace, so that the velocity of the system can be represented by means of a new vector $\dot{\mathbf{z}}$, whose components are those of the vector $\dot{\mathbf{q}}$ on the chosen nullspace basis. Vector $\dot{\mathbf{z}}$ will have only $f=n-m$ components which will be independent.

Let $\mathbf{r}^i (i=1, 2, \dots, f)$ be a set of f linearly independent vectors that constitute a basis of the nullspace of Φ_q . Any dependent velocity vector $\dot{\mathbf{q}}$ can be expressed as a linear combination of this basis as follows:

$$\dot{\mathbf{q}} = \mathbf{r}^1 \dot{z}_1 + \mathbf{r}^2 \dot{z}_2 + \dots + \mathbf{r}^f \dot{z}_f \tag{3.19}$$

Introducing an $(n \times f)$ matrix \mathbf{R} , whose columns are the vectors \mathbf{r}^i , this expression can be written as

$$\dot{\mathbf{q}} = \mathbf{R} \dot{\mathbf{z}} \tag{3.20}$$

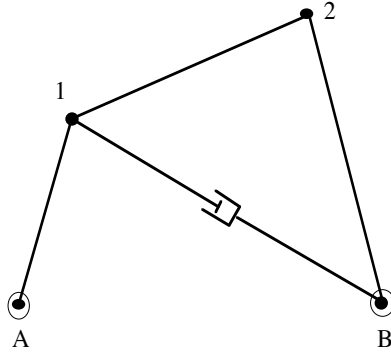


Figure 3.17. Hydraulically driven four-bar mechanism.

Matrix \mathbf{R} thus defined plays a very important role in some of the most efficient formulations for dynamic analysis. Since vectors \mathbf{r}^i are the components of a basis of the nullspace of the Jacobian matrix, it can be verified that,

$$\Phi_{\mathbf{q}}(\mathbf{q}) \mathbf{r}^i = \mathbf{0} \quad (i = 1, 2, \dots, f) \quad (3.21)$$

and consequently

$$\Phi_{\mathbf{q}}(\mathbf{q}) \mathbf{R} = \mathbf{0} \quad (3.22)$$

The previous expression is a reminder that the matrix \mathbf{R} depends on the position vector \mathbf{q} , and therefore there is a different matrix \mathbf{R} for each of the positions of the multibody system.

The results obtained for the velocities can also be extended to the accelerations. We must search for a way of expressing the dependent accelerations $\ddot{\mathbf{q}}$ in terms of f independent accelerations $\ddot{\mathbf{z}}$. Differentiating (3.20) with respect to time we arrive at the following expression:

$$\ddot{\mathbf{q}} = \mathbf{R} \ddot{\mathbf{z}} + \dot{\mathbf{R}} \dot{\mathbf{z}} \quad (3.23)$$

Note that now both matrices \mathbf{R} and $\dot{\mathbf{R}}$ are needed for the acceleration transformation. The calculation method for matrix $\dot{\mathbf{R}}$ depends on the method adopted to form \mathbf{R} and will be seen later on in this chapter.

Even though in the simple examples presented in this section the matrix \mathbf{R} has been calculated symbolically. In practice, this matrix needs to be calculated numerically. However, both the concepts and applications concerning the matrix \mathbf{R} remain of general validity.

3.5.2 Rheonomous Systems

Rheonomous systems are characterized by the fact that some of the constraint equations depend on the time variable. This general case will be used to general-

ize the concepts and ideas introduced previously. For rheonomous systems the analytical expression for the constraint equations become:

$$\Phi(\mathbf{q}, t) = \mathbf{0} \quad (3.24)$$

The velocity and acceleration equations are obtained by differentiating this equation with respect to time once and twice, respectively:

$$\Phi_{\mathbf{q}}(\mathbf{q}, t) \dot{\mathbf{q}} = -\dot{\Phi}_t \equiv \mathbf{b} \quad (3.25)$$

$$\Phi_{\mathbf{q}}(\mathbf{q}, t) \ddot{\mathbf{q}} = -\ddot{\Phi}_t - \dot{\Phi}_{\mathbf{q}} \dot{\mathbf{q}} \equiv \mathbf{c} \quad (3.26)$$

where the dot indicates total derivative; and the sub index t , the partial derivative with respect to time. Equations (3.25) and (3.26) serve as definitions for the right-hand side vectors \mathbf{b} and \mathbf{c} , which will be extensively used in the dynamic formulations of Chapter 5.

If all the degrees of freedom of the multibody system are controlled kinematically, that is, if the motion of all the input elements is known as a function of time, equations (3.25) and (3.26) constitute two systems of m equations with m unknowns controlled by rank m matrices. The solution of these equation systems is perfectly determined, and there should be no problem in finding this solution.

Example 3.12

Such is the case with all the mechanism's degrees of freedom controlled kinematically in the hydraulically driven four-bar mechanism shown in Figure 3.17, whose constraint equations are:

$$(x_1 - x_A)^2 + (y_1 - y_A)^2 - L_2^2 = 0$$

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 - L_3^2 = 0$$

$$(x_2 - x_B)^2 + (y_2 - y_B)^2 - L_4^2 = 0$$

$$(x_1 - x_B)^2 + (y_1 - y_B)^2 - f(t)^2 = 0$$

It may be seen that the last constraint is time dependent, thus rheonomous.

It is noted that the general case, where some of the input elements can be controlled kinematically (their motions prescribed by means of rheonomous constraint equations), and others have their motion kinematically undetermined, constitutes a dynamic problem determined by the differential equations of motion. From here on, it will be assumed that the equation (3.25) has a total number of m independent constraint equations, corresponding to the constraints of rigid body, joints, and degrees of freedom kinematically controlled by means of rheonomous equations. If there are n dependent coordinates, there will be $(n-m)$ free or kinematically undetermined degrees of freedom.

We will introduce now a large family of methods in which the independent velocities $\dot{\mathbf{z}}$ can be defined as the projection of the dependent velocities $\dot{\mathbf{q}}$ on the rows of a constant (not time or position dependent) matrix \mathbf{B}

$$\dot{\mathbf{z}} = \mathbf{B} \dot{\mathbf{q}} \quad (3.27)$$

Equation (3.26) can be augmented by equation (3.27) to yield

$$\begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix} \dot{\mathbf{q}} = \begin{bmatrix} \mathbf{b} \\ \dot{\mathbf{z}} \end{bmatrix} \quad (3.28)$$

Let us assume at this point that matrix \mathbf{B} , in addition to being constant, also fulfills the condition of having $f=n-m$ rows that are linearly independent from one another and also linearly independent of the m rows of Φ_q .

With these assumptions, the matrix in equation (3.28) can be inverted, and finding the vector $\dot{\mathbf{q}}$ involves the solution of the following equation:

$$\dot{\mathbf{q}} = \begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{b} \\ \dot{\mathbf{z}} \end{bmatrix} \equiv \mathbf{S} \mathbf{b} + \mathbf{R} \dot{\mathbf{z}} \quad (3.29)$$

where \mathbf{S} is a matrix constituted by the m first columns of the inverse matrix of equation (3.29), and \mathbf{R} is the matrix constituted by the $f=n-m$ last columns of the said inverse matrix. It can be verified that

$$\begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix} \begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix}^{-1} = \begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix} [\mathbf{S} \mathbf{R}] = \begin{bmatrix} \Phi_q \mathbf{S} & \Phi_q \mathbf{R} \\ \mathbf{B} \mathbf{S} & \mathbf{B} \mathbf{R} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (3.30)$$

which demonstrates that the columns of matrix \mathbf{R} pertain to and generate the nullspace of Φ_q .

Regarding the linear equation system (3.25) which is undetermined as long as a value is not given to the input velocities, equation (3.29) indicates that the general solution of the system is obtained as the sum of a *particular solution* of the complete equation (term \mathbf{Sb}) in addition to the *general solution* of the homogeneous equation (term $\mathbf{R}\dot{\mathbf{z}}$).

The result of equation (3.29) may be compared with the terminology commonly used in Kane's method (Kane and Levinson (1985)). The columns of matrix \mathbf{R} are the *partial velocities* with respect to the generalized coordinates \mathbf{z} , and the term \mathbf{Sb} constitutes the *partial velocities with respect to time*. However, the approach presented herein includes a more general algebraic method to compute these partial velocities for all kind of multibody systems: open or closed chains, overconstrained, singular positions, and so forth.

The acceleration equation can be obtained in a similar manner. Augmenting equation (3.26) with the derivative with respect to time of equation (3.27), we obtain:

$$\begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix} \ddot{\mathbf{q}} = \begin{bmatrix} \mathbf{c} \\ \ddot{\mathbf{z}} \end{bmatrix} \quad (3.31)$$

and the inversion of this matrix:

$$\ddot{\mathbf{q}} = \begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{c} \\ \ddot{\mathbf{z}} \end{bmatrix} = \mathbf{S} \mathbf{c} + \mathbf{R} \ddot{\mathbf{z}} \quad (3.32)$$

This expression, analogous to expression (3.29), indicates that matrix \mathbf{R} can be calculated by triangularizing the leading matrix of systems (3.28) or (3.31),

and performing f successive forward and backward substitutions with the f last columns of unit matrix \mathbf{I} as the RHS terms.

Some of the dynamic formulations that will be seen in Chapter 5 require the calculation of the term $(\mathbf{S}\mathbf{c})$ in expression (3.32). This term generalizes the role of the term $(\mathbf{R}\dot{\mathbf{z}})$ in equation (3.23). In order to determine $(\mathbf{S}\mathbf{c})$, one possibility is to calculate the matrix \mathbf{S} by the same method used to calculate \mathbf{R} and then multiply by the known vector \mathbf{c} . This way is valid but not very efficient. It is not necessary to calculate matrix \mathbf{S} , but just to calculate the product $(\mathbf{S}\mathbf{c})$. From expression (3.32), it is concluded that the product $(\mathbf{S}\mathbf{c})$ is $\ddot{\mathbf{q}}$ when $\dot{\mathbf{z}}$ is zero. By making $\dot{\mathbf{z}}$ equal to zero in expression (3.31) and finding $\ddot{\mathbf{q}}$, one can arrive at the desired term. Since the leading matrix of system (3.31) has been previously triangularized when finding matrix \mathbf{R} , the calculation of the term being considered requires very little additional effort.

During the preceding development one has been able to see that the inclusion of rheonomous links in the analysis can be carried out very simply and efficiently. The methods currently used to determine a basis of the subspace of allowable motions (matrix \mathbf{R}) are divided into two large groups, the *projection* methods and the *orthogonalization* methods, which will be studied in the following sections.

3.5.3 Calculation of Matrix \mathbf{R} : Projection Methods

Projection methods are based on defining the independent velocities $\dot{\mathbf{z}}$ as the projection of the dependent velocities $\dot{\mathbf{q}}$ on the rows of a known constant matrix \mathbf{B} :

$$\dot{\mathbf{z}} = \mathbf{B} \dot{\mathbf{q}} \quad (3.33)$$

Following the mathematical formulation of Section 3.5.2 for the general case of rheonomous systems, the following expressions have been obtained:

$$\begin{bmatrix} \Phi_{\mathbf{q}} \\ \mathbf{B} \end{bmatrix} \dot{\mathbf{q}} = \begin{bmatrix} \mathbf{b} \\ \dot{\mathbf{z}} \end{bmatrix} \quad (3.34)$$

$$\dot{\mathbf{q}} = \begin{bmatrix} \Phi_{\mathbf{q}} \\ \mathbf{B} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{b} \\ \dot{\mathbf{z}} \end{bmatrix} = \mathbf{S} \mathbf{b} + \mathbf{R} \dot{\mathbf{z}} \quad (3.35)$$

$$\begin{bmatrix} \Phi_{\mathbf{q}} \\ \mathbf{B} \end{bmatrix} \ddot{\mathbf{q}} = \begin{bmatrix} \mathbf{c} \\ \ddot{\mathbf{z}} \end{bmatrix} \quad (3.36)$$

$$\ddot{\mathbf{q}} = \begin{bmatrix} \Phi_{\mathbf{q}} \\ \mathbf{B} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{c} \\ \ddot{\mathbf{z}} \end{bmatrix} = \mathbf{S} \mathbf{c} + \mathbf{R} \ddot{\mathbf{z}} \quad (3.37)$$

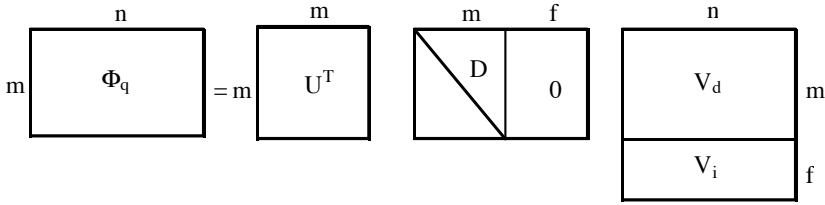


Figure 3.18. Singular Value Decomposition of the matrix Φ_q .

It is clear that these expressions completely define the transformation between dependent and independent variables.

This only leaves matrix \mathbf{B} to be determined. Once this matrix is calculated, it remains constant during a large range of motion of the multibody system. The condition with which matrix \mathbf{B} must comply in order for the inverse matrix in expressions (3.35) and (3.37) to exist, its $n-m$ rows must be independent from one another and independent from the m rows of matrix Φ_q . We can identify and describe in this context three methods that have been proposed in the literature to construct the matrix \mathbf{B} . These will be reviewed below.

1. *Method based on the Singular Value decomposition.* Singular Value decomposition (SV) is a generalization of the eigenvalue and eigenvector concept applicable to rectangular matrices. The SV decomposes a rectangular matrix such as Φ_q as indicated in the sketch of Figure 3.18, or shortly:

$$\Phi_q = U^T \mathbf{D} \mathbf{V} \quad (3.38)$$

where matrix \mathbf{U} is orthogonal (its inverse is equal to its transpose and its rows are mutually orthogonal) of size $(m \times m)$. Matrix \mathbf{D} is composed of a diagonal matrix of size $(m \times m)$ that contains the singular values and a zero matrix given by $f=n-m$ last columns. Matrix \mathbf{V} is orthogonal of size $(n \times n)$ and can be decomposed into two sub-matrices \mathbf{V}_d and \mathbf{V}_i of sizes $(m \times n)$ and $(f \times n)$ respectively, according to the partition in \mathbf{D} . The most important property of the SV decomposition that pertains to the problem at hand is that the rows of the matrix \mathbf{V}_i constitute an *orthogonal basis* of the nullspace of matrix Φ_q . In other words, it is verified that

$$\Phi_q \mathbf{V}_i^T = \mathbf{0} \quad (3.39)$$

In view of this expression, Singh and Likins (1985) proposed constructing the matrix \mathbf{R} directly from the SV decomposition. The problem is that the SVD is essentially an iterative process in some ways similar to the calculation of all the eigenvalues and eigenvectors of a matrix. This process consumes a great deal of computer time, and it is absolutely impractical to carry out at each position \mathbf{q} of the system. Other authors (Mani, Haug, and Atkinson (1985)) have proposed us-

ing the SV decomposition to calculate the matrix \mathbf{B} . This operation only needs to be performed once or at most a few times throughout the entire range of the motion of the multibody system. Bear in mind that many matrices \mathbf{R} , corresponding to different positions \mathbf{q} of the multibody system, can be calculated from only one matrix \mathbf{B} . Matrix \mathbf{B} continues to be valid as long as its rows are independent from those of $\Phi_{\mathbf{q}}(\mathbf{q})$.

Equation (3.39) indicates that the rows of matrix \mathbf{V}_i are orthogonal to the rows of $\Phi_{\mathbf{q}}$ at the position \mathbf{q} , for which the SV decomposition has been performed. This means that matrix \mathbf{V}_i complies with the conditions required for matrix \mathbf{B} , so long as no large changes are produced in the positions \mathbf{q} and, thus in matrix $\Phi_{\mathbf{q}}$, that the linear independence condition between the rows of the said matrix and those of matrix \mathbf{B} is lost. The following example helps to clarify this point.

Example 3.13

Demonstrate that after performing the singular value decomposition of $\Phi_{\mathbf{q}}$, under the condition that $\mathbf{R} = \mathbf{V}_i^T$, the following relation $\mathbf{B} = \mathbf{V}_i$ is satisfied.

After the singular value decomposition $\Phi_{\mathbf{q}}$ is orthogonal to \mathbf{V}_i^T , we can take $\mathbf{R} = \mathbf{V}_i^T$, and also

$$\dot{\mathbf{q}} = \mathbf{R} \dot{\mathbf{z}}$$

and

$$\dot{\mathbf{z}} = \mathbf{B} \dot{\mathbf{q}} = \mathbf{B} \mathbf{R} \dot{\mathbf{z}}$$

Therefore $\mathbf{B} \mathbf{R} = \mathbf{I}$. Since $\mathbf{R} = \mathbf{V}_i^T$ and \mathbf{V}_i is an orthogonal matrix, the following relationship: $\mathbf{B} = \mathbf{V}_i$ immediately follows.

This method of calculating the matrix \mathbf{R} has a very simple geometric interpretation that can be seen in Figure 3.19. Let's assume that bar OA is fixed at O by means of a spherical pair. If the rotation around the axis OA is not considered, this mechanism has two degrees of freedom and three natural coordinates, the Cartesian coordinates of point A. Thus, it will have one constraint equation which will be the constant distance condition between points O and A. In this case $m=1$, $n=3$ and $f=2$.

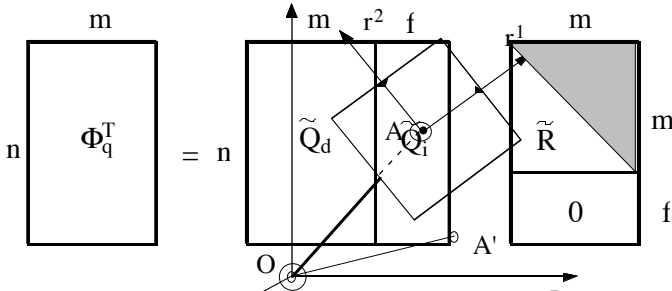


Figure 3.20. QR Decomposition of the matrix Φ_q .

Figure 3.19. Graphical representation of the columns of the matrix \mathbf{B} .

The subspace of the possible movements has a dimension 2, and is formed by the plane perpendicular to OA through A , since in fact all the possible velocity vectors of point A are contained in the said plane. Vectors \mathbf{r}^1 and \mathbf{r}^2 constitute an orthogonal basis of the subspace. It has been possible to calculate them by means of the SV decomposition of matrix Φ_q at the position OA .

Now it can be assumed that the mechanism moves and changes to a new position OA' . Matrix \mathbf{B} continues being defined by vectors \mathbf{r}^1 and \mathbf{r}^2 , calculated at the position OA . The independent velocities \dot{z}_1 and \dot{z}_2 are the projections of the velocity of A' on the axes \mathbf{r}^1 and \mathbf{r}^2 , respectively. Note that the velocity of A' is no longer contained in the tangent plane at A but can be easily determined from its projections on \mathbf{r}^1 and \mathbf{r}^2 , and by the condition that it is perpendicular to OA' (constraint condition for velocities).

2. *Method based on the QR decomposition.* This method of constructing matrix \mathbf{B} is similar to the previous one, but it uses the QR instead of the SV decomposition. The main advantage is that QR decomposition is a direct, not iterative, process which requires considerably fewer arithmetical operations as indicated by Kim and Vanderploeg (1986).

The QR method decomposes the matrix Φ_q^T as indicated in the sketch of figure 3.20, or briefly,

$$\Phi_q^T = \tilde{\mathbf{Q}} \tilde{\mathbf{R}} \tag{3.40}$$

where $\tilde{\mathbf{Q}}$ is an orthogonal ($n \times n$) matrix, and $\tilde{\mathbf{R}}$ is a rectangular ($n \times m$) matrix formed by an upper triangular matrix ($m \times m$) and a zero matrix of order ($f \times m$). Note that a tilde has been used to distinguish the result of the QR decomposition from the matrix \mathbf{Q} that symbolizes the forcing vector in dynamic analysis (Chapter 4) and the matrix \mathbf{R} (basis of the nullspace of the Jacobian matrix). The application of this decomposition to the problem at hand is straightforward when considering that the f last columns of $\tilde{\mathbf{Q}}$ which define the sub-matrix $\tilde{\mathbf{Q}}$ consti-

tute an orthogonal basis of the nullspace of the matrix Φ_q . This matrix can be written as

$$\mathbf{B} = \tilde{\mathbf{Q}}_i^T \tag{3.41}$$

and likewise verified that

$$\Phi_q(\mathbf{q}) \tilde{\mathbf{Q}}_i = 0 \tag{3.42}$$

This matrix \mathbf{B} is used in exactly the same way as that calculated by means of the SV decomposition. The QR decomposition is carried out at a determined position \mathbf{q} of the multibody system. The matrix \mathbf{B} is formed, and the matrix \mathbf{R} is calculated for the successive positions of the multibody system using equation (3.35), without recalculating the matrix \mathbf{B} . This recalculation will have to be done when its rows become a linear combination of those of Φ_q .

The geometric interpretation of the method based on QR decomposition is similar to that of the SV decomposition. Returning to Figure 3.19, matrix \mathbf{B} formed by vectors \mathbf{r}^1 and \mathbf{r}^2 gives inadequate results when the bar moves to a position perpendicular to OA and therefore, parallel to the plane defined by \mathbf{r}^1 and \mathbf{r}^2 . At this position the derivative of the constant distance condition in matrix Φ_q is a linear combination of \mathbf{r}^1 and \mathbf{r}^2 and the matrix of equations (3.34) and (3.36) cannot be inverted.

Both the QR and SV decomposition can be carried out by means of standard Fortran or C subroutines, contained in the readily available IMSL, Harwell, NAG, and other mathematical libraries.

3. Method based on Gaussian triangularization. This method, described by Serna et al. (1982), is based on the triangularization of matrix Φ_q by means of the Gauss method with total pivoting. This triangularization implies decomposition of the Jacobian matrix in sub-matrices, as shown below:

$$\Phi_q \equiv \begin{bmatrix} \Phi_q^d & \Phi_q^i \end{bmatrix} \tag{3.43}$$

where matrix Φ_q^d is a square matrix ($m \times m$) that contains the columns of Φ_q in which the pivots have appeared. Matrix Φ_q^i contains the columns in which the pivots have not appeared and has the size ($m \times f$). In the theory of linear equation systems, the variables associated with columns Φ_q^i are called *independent variables*, and those associated with columns Φ_q^d are called *dependent variables*. The reason for this nomenclature is that to solve a system of m equations with n unknowns, with a matrix such as that in equation (3.43), it is necessary to assign a value to the independent variables and then, with matrix Φ_q^d reduced to triangular form, calculate the dependent variables with the corresponding forward and backward substitutions.

Once matrix Φ_q is triangularized as shown in equation (3.43), matrix \mathbf{B} is a Boolean matrix constructed as follows:

$$\mathbf{B} \equiv \begin{bmatrix} \mathbf{0} & | & \mathbf{I} \end{bmatrix} \tag{3.44}$$

whereupon the matrix from which the inverse of matrix \mathbf{R} is calculated is

$$\begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix} = \begin{bmatrix} \Phi_q^d & \Phi_q^i \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (3.45)$$

The rows of matrix \mathbf{B} defined in this way are formed by ones and zeros. Since matrix Φ_q^d is triangularizable, it is guaranteed that the rows of matrix \mathbf{B} are independent from those of Φ_q . Note that the triangularization of matrix (3.45) is simpler than with the SV or QR decomposition. In the part corresponding to matrix \mathbf{B} , no additional work is necessary, since the zeros have already been obtained. With this method, matrix \mathbf{R} is calculated more easily and with fewer arithmetical operations.

Other repercussions from choosing matrix \mathbf{B} , in accordance with equation (3.44), will be analyzed below. Particularizing equation (3.27) for this case,

$$\dot{\mathbf{z}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \dot{\mathbf{q}} \quad (3.46)$$

This expression indicates that the independent velocities $\dot{\mathbf{z}}$ are chosen as a *subset* or *extraction* of the dependent velocities $\dot{\mathbf{q}}$. In other words, f elements of $\dot{\mathbf{q}}$ have been chosen to form vector $\dot{\mathbf{z}}$.

Example 3.14

Let us consider a planar mechanism with five bars and two degrees of freedom, as shown in Figure 3.21. The independent coordinates are a subset of the dependent coordinates. Let's assume that the independent velocities are

$$\dot{z}_1 \equiv \dot{y}_2$$

$$\dot{z}_2 \equiv \dot{x}_3$$

According to expression (3.35), matrix \mathbf{R} can be formed thusly. The first column of matrix \mathbf{R} (vector \mathbf{r}^1) is the velocity vector of the mechanism, when the independent velocities have the following value:

$$\dot{y}_2 = 1 \quad \dot{x}_3 = 0$$

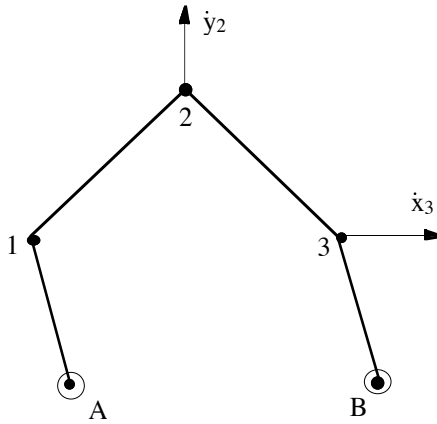


Figure 3.21. Five bar mechanism.

The second column of matrix \mathbf{R} (vector \mathbf{r}^2) is the velocity vector of the mechanism, when the independent velocities are:

$$\dot{y}_2 = 0 \quad \dot{x}_3 = 1$$

Thus in this case and with this type of matrix \mathbf{B} , matrix \mathbf{R} is especially easy to construct. It will suffice to alternatively give value 1 to each of the independent velocities, keeping all the others at the value 0. Therefore, the partition of matrix \mathbf{R} , which corresponds to the independent velocities, becomes the unit matrix \mathbf{I} .

This geometrical interpretation and significance of the independent velocities contrasts with that carried out for the SVD, starting from Figure 3.19.

3.5.4. Orthogonalization Methods

One can use the orthogonalization methods to try to obtain a matrix \mathbf{R} , whose columns are in some way orthogonal to one another at all times. The method that will be explained here was developed by Liang and Lance (1985) and is based on the previously considered matrix:

$$\mathbf{P} \equiv \begin{bmatrix} \Phi_q \\ \mathbf{B} \end{bmatrix} \tag{3.47}$$

where the matrix \mathbf{B} can be constructed by means of any of the methods shown in previous sections, but preferably by means of the third method. This produces a Boolean matrix \mathbf{B} in accordance with the partition of Φ_q , and determined by the Gaussian triangularization method with total pivoting.

The first step in this method consists of orthogonalizing the n rows of matrix \mathbf{P} by means of the Gram-Schmidt orthogonalization method. This yields an $(n \times n)$ matrix as follows:

$$\mathbf{V} \equiv \begin{bmatrix} \mathbf{V}^d \\ \mathbf{V}^i \end{bmatrix} \quad (3.48)$$

where the rows of \mathbf{V}^d are the rows of Φ_q with each of them orthogonalized in relation to the previous ones, and where the rows of \mathbf{V}^i are the rows of matrix \mathbf{B} with each one of them orthogonalized in relation to the rows of \mathbf{V}^i and to the previous rows of \mathbf{V}^d .

According to the standard Gram-Schmidt orthogonalization process, all the orthonormal vectors \mathbf{v}^i are calculated by means of the general expressions:

$$\mathbf{v}^i = \alpha_i \left(\mathbf{p}^i - \sum_1^{i-1} (\mathbf{v}^j \cdot \mathbf{p}^i) \mathbf{v}^j \right) \quad (3.49)$$

and

$$\alpha_i = 1 / \left\| \mathbf{p}^i - \sum_1^{i-1} (\mathbf{v}^j \cdot \mathbf{p}^i) \mathbf{v}^j \right\| \quad (3.50)$$

The f last vectors obtained in this way correspond to the rows of matrix \mathbf{B} , orthogonalized with respect to those of Φ_q and in relation to the previous rows of \mathbf{B} . These f rows form matrix \mathbf{V}^i (See equation (3.48)). This matrix is orthogonal to Φ_q . In addition, the rows of \mathbf{V}^i are independent and mutually orthogonal. By the first condition, the rows of \mathbf{V}^i pertain to the Φ_q nullspace, and by the second condition, they constitute a basis. Thus, matrix \mathbf{V}^{iT} can be taken as matrix \mathbf{R} .

The final step is the calculation of the term $(\mathbf{R}\dot{\mathbf{z}})$ or $(\mathbf{S}\mathbf{c})$ is calculated. Liang and Lance (1985) calculate matrix $\dot{\mathbf{R}}$ explicitly, differentiating equations (3.49) and (3.50) with respect to time:

$$\begin{aligned} \dot{\mathbf{v}}^i &= \dot{\alpha}_i \left(\mathbf{p}^i - \sum_1^{i-1} (\mathbf{v}^j \cdot \mathbf{p}^i) \mathbf{v}^j \right) + \\ &+ \alpha_i \left(\dot{\mathbf{p}}^i - \sum_1^{i-1} (\dot{\mathbf{v}}^j \cdot \mathbf{p}^i + \mathbf{v}^j \cdot \dot{\mathbf{p}}^i) \mathbf{v}^j - \sum_1^{i-1} (\mathbf{v}^j \cdot \mathbf{p}^i) \dot{\mathbf{v}}^j \right) \end{aligned} \quad (3.51)$$

and

$$\dot{\alpha}_i = - \left(\mathbf{p}^i \cdot \mathbf{p}^i - \sum_1^{i-1} (\mathbf{v}^j \cdot \mathbf{p}^i)^2 \right)^{-\frac{3}{2}} \left(\dot{\mathbf{p}}^i \cdot \mathbf{p}^i - \sum_1^{i-1} (\mathbf{v}^j \cdot \mathbf{p}^i) (\dot{\mathbf{v}}^j \cdot \mathbf{p}^i + \mathbf{v}^j \cdot \dot{\mathbf{p}}^i) \right) \quad (3.52)$$

In these expressions, the derivatives of the \mathbf{P} rows are obtained from the derivatives of the rows of Φ_q . The derivatives of the \mathbf{B} rows are zero.

The calculation of matrix \mathbf{R} by means of equations (3.49) and (3.50) and of matrix $\dot{\mathbf{R}}$ by means of equations (3.51) and (3.52) requires an enormous computational effort, which is far greater than that required for the previously explained projection methods. Another important feature of this method is that the matrix \mathbf{R} obtained depends on the order in which rows Φ_q and \mathbf{B} are considered, since the Gram-Schmidt orthogonalization depends on this order. The effect of this order is not yet known.

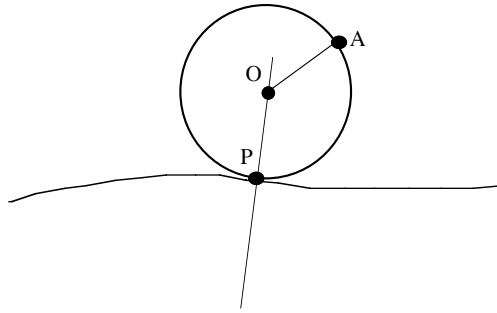


Figure 3.22. Disc modeled with two basic points, in contact with a surface.

3.6 Multibody Systems with Non-Holonomic Joints

Non-holonomic pairs or joints are those whose constraint equation (joint constraint equations) do not depend on the dependent coordinates only but also on the dependent velocities by means of non-integrable equations. The rolling of a disc or wheel on a surface is a typical example of the non-holonomic constraint. It is the only one that will be discussed here. A distinction will be made between the planar case and the three-dimensional one.

3.6.1 Wheel Element in the Planar Case: First Method.

Figure 3.22 shows a disc in contact with a surface. In principle, there can be two types of movement between the surface of the wheel and that of the track: *rolling* and rolling plus sliding (simply referred to as *sliding*).

The fact that the movement is one type or the other depends on the dynamic conditions of the problem (coefficient of friction and contact force), which will not be studied here. In this section, only those constraint equations corresponding to the rolling and sliding conditions will be studied.

There are at least two ways of establishing the constraint equations of the system shown in Figure 3.22. One way is to directly establish the non-holonomic constraint equations in terms of dependent velocities. The second way is to substitute the non-holonomic joint with one or more equivalent holonomic joints for that position of the multibody system. Both ways will be seen further on. In either case, it will be assumed that the wheel is modeled by means of two basic points, one at the center and the other at the end of a radius. The movement of both points completely defines the movement of the wheel.

The constraint equations of non-holonomic joints are directly generated in terms of the velocities. Therefore, *there are no constraint equations for the position problem*. To solve the position problem with wheels and tracks, it is necessary to dispense with the wheels and find the position of their centers. It is known that they are located on a curve, as shown in Figure 3.23, which is the

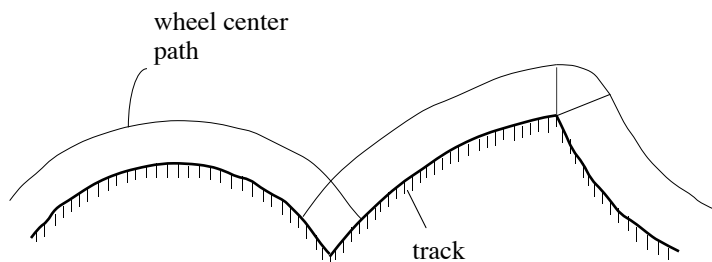


Figure 3.23. Wheel center path.

geometric place of the possible positions of the wheel center. Once the position of the center of the wheel is known, the wheel can be placed at the desired angular position.

Rolling. In the case that there is a rolling motion between the wheel and the track, the joint kinematic constraint equation establishes that the velocity of the wheel point in contact with the track shall be zero. If the track moves, the condition is that the velocity of the two points in contact is the same. From here on, it will be assumed that the track is stationary and all that stated previously will be applicable to the relative movement between the rolling wheel and the track.

Depending on the velocities of the basic points O and A , the condition that the velocity of point P be zero is equivalent to the conditions that the velocities of O and A be perpendicular to PO and PA respectively. When the coordinates of P , O , and A are known, these conditions are easily established by means of the scalar product of vectors:

$$(x_O - x_P) \dot{x}_O + (y_O - y_P) \dot{y}_O = 0 \quad (3.53)$$

$$(x_A - x_P) \dot{x}_A + (y_A - y_P) \dot{y}_A = 0 \quad (3.54)$$

Point P can be determined by means of the normal line traced from point O to the track.

Sliding. In the case that there is a sliding motion, the constraint condition establishes that point P does not have any velocity in the direction normal to the surfaces in contact. This condition is equivalent to equation (3.53), which establishes that the velocity of point O is perpendicular to PO . Note that in this case the equation (3.54) is no longer valid. It should be noted that when there is a rolling movement, two degrees of freedom are restricted. If there is sliding, only one degree of freedom is restricted. This is in accordance with the number of equations that should be satisfied.

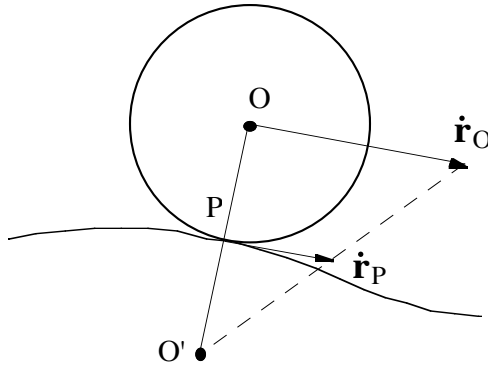


Figure 3.24. Description of the center of curvature of the wheel.

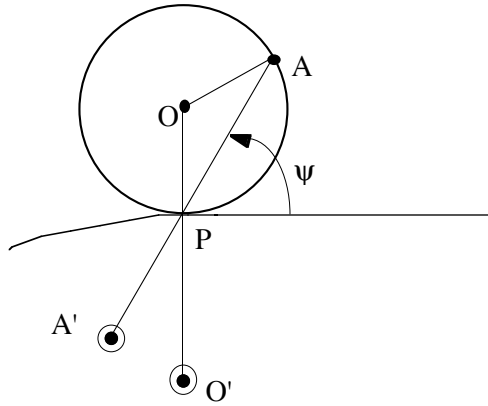


Figure 3.25. Substituting a non-holonomic constraint by a holonomic ones.

To derive equations (3.53) and (3.54) and to find the equations corresponding to the accelerations, one should bear in mind that in order for the equations to be valid at all times, point P must not belong to either the wheel or to the rolling track, but should be the mathematical point that always coincides with the pole (center of velocities). If equations (3.53) and (3.54) are differentiated with respect to the time, one obtains

$$(x_O - x_P) \ddot{x}_O + (\dot{x}_O - \dot{x}_P) \dot{x}_O + (y_O - y_P) \ddot{y}_O + (\dot{y}_O - \dot{y}_P) \dot{y}_O = 0 \quad (3.55)$$

$$(x_A - x_P) \ddot{x}_A + (\dot{x}_A - \dot{x}_P) \dot{x}_A + (y_A - y_P) \ddot{y}_A + (\dot{y}_A - \dot{y}_P) \dot{y}_A = 0 \quad (3.56)$$

In these equations, (x_P, y_P) are the components of the pole velocity which must be calculated, since equations (3.55) and (3.56) should not have unknowns other than the natural accelerations.

To calculate the pole velocity $\dot{\mathbf{r}}_P$, one discovers the only solution is to consider the geometry of the trajectory. If O' is the center of curvature of the wheel

center O trajectory (See Figure 3.24), points P and O are always aligned with O'. The ends of the corresponding velocity vectors are aligned also. From this it may be deduced that:

$$\dot{\mathbf{r}}_P = (O'P)/(O'O) \cdot \dot{\mathbf{r}}_O \quad (3.57)$$

By decomposing this expression into its Cartesian coordinates and substituting them in equations (3.55) and (3.56), the constraint equations corresponding to the accelerations can be obtained.

3.6.2 Wheel Element in the Planar Case: Second Method.

Another way of entering the rolling and sliding conditions is by substituting the non-holonomic joint for one or more equivalent holonomic joints.

In Figure 3.25, the said substitution is being carried out and is based on knowledge of the centers of curvature of the trajectories of the basic points O and A. The center of curvature of the trajectory of O is the center of curvature of the track at the point of contact P. The center of curvature of the trajectory of A can be calculated using the Euler-Savary formula:

$$\left(\frac{1}{PA} + \frac{1}{PA'} \right) \sin \Psi = \frac{1}{PO} + \frac{1}{PO'} \quad (3.58)$$

Rolling. In the case of a rolling motion, the non-holonomic joint is replaced by the articulated quadrilateral A'AOO', since the distance between the point and the center of curvature of its trajectory is constant and has first and second constant derivatives. Therefore, this equivalence is instantaneously valid for velocities and accelerations. Note that at another time, the equivalent four-bar mechanism will be different. For the position of Figure 3.25, the constraint equations are

$$(x_A - x_{A'})^2 + (y_A - y_{A'})^2 - L_{AA'}^2 = 0 \quad (3.59)$$

$$(x_O - x_{O'})^2 + (y_O - y_{O'})^2 - L_{OO'}^2 = 0 \quad (3.60)$$

Sliding. In the case of a sliding movement, only the constant distance condition between points O and O' should be imposed. The equations corresponding to velocities and accelerations are obtained by differentiating equations (3.59) and (3.60) and considering A' and O' as fixed points.

3.6.3 Wheel Element in the Three-Dimensional Case.

Consider the three-dimensional wheel element shown in Figure 3.26, which is formed by two basic points O and A and a unit vector \mathbf{u} perpendicular to the plane of the wheel and associated to point O.

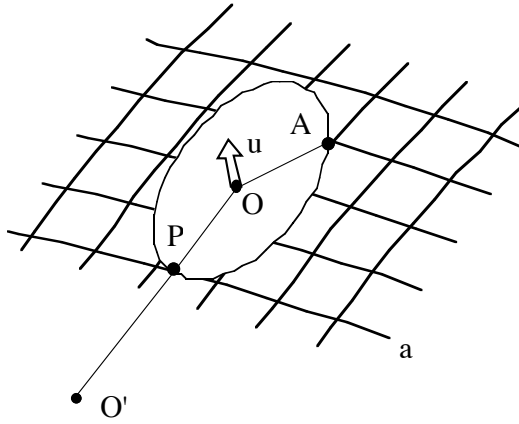


Figure 3.26. Modeling of a three dimensional wheel with natural coordinates.

Rolling. The constraint equations corresponding to this non-holonomic joint are defined by the condition that the velocity of material point P (pertaining to the wheel) is zero:

$$\dot{\mathbf{r}}_P = \mathbf{0} \tag{3.61}$$

The vector equation (3.61) represents three algebraic equations. Therefore three degrees of freedom are restricted by this joint when there is a rolling movement, thus only permitting the three rotations about point P. It is necessary now to consider equation (3.61), in accordance with the natural velocities and coordinates. The velocities of points O and A can be expressed in terms of the angular velocity vector $\boldsymbol{\omega}$:

$$\dot{\mathbf{r}}_O = \boldsymbol{\omega} \wedge (\mathbf{r}_O - \mathbf{r}_P) \tag{3.62}$$

$$\dot{\mathbf{r}}_A = \boldsymbol{\omega} \wedge (\mathbf{r}_A - \mathbf{r}_P) \tag{3.63}$$

which can be expanded to yield

$$\dot{x}_O = \omega_y (z_O - z_P) - \omega_z (y_O - y_P) \tag{3.64}$$

$$\dot{y}_O = \omega_z (x_O - x_P) - \omega_x (z_O - z_P) \tag{3.65}$$

$$\dot{z}_O = \omega_x (y_O - y_P) - \omega_y (x_O - x_P) \tag{3.66}$$

$$\dot{x}_A = \omega_y (z_A - z_P) - \omega_z (y_A - y_P) \tag{3.67}$$

$$\dot{y}_A = \omega_z (x_A - x_P) - \omega_x (z_A - z_P) \tag{3.68}$$

$$\dot{z}_A = \omega_x (y_A - y_P) - \omega_y (x_A - x_P) \tag{3.69}$$

Three conveniently selected equations among the previous ones, which are two corresponding to one point and one to the other, permit determining the vec-

tor $\boldsymbol{\omega}$. For example, by selecting equations (3.64), (3.65), and (3.69), the following matrix equation can be written:

$$\begin{bmatrix} O & (z_O - z_P) & -(y_O - y_P) \\ -(z_O - z_P) & O & (x_O - x_P) \\ (y_A - y_P) & -(x_A - x_P) & O \end{bmatrix} \begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix} = \begin{Bmatrix} \dot{x}_O \\ \dot{y}_O \\ \dot{z}_A \end{Bmatrix} \quad (3.70)$$

If the matrix of this system is not singular, one can find the angular velocity vector $\boldsymbol{\omega}$ and substitute its value in the remaining equations (3.66), (3.67), and (3.68), which are those not used to determine the angular velocity vector. Consequently

$$\begin{aligned} & \begin{Bmatrix} \dot{z}_O \\ \dot{x}_A \\ \dot{y}_A \end{Bmatrix} = \\ & = \begin{bmatrix} (y_O - y_P) & -(x_O - x_P) & 0 \\ 0 & (z_A - z_P) & -(y_A - y_P) \\ -(z_A - z_P) & 0 & -(x_A - x_P) \end{bmatrix} \begin{bmatrix} 0 & (z_O - z_P) & -(y_O - y_P) \\ -(z_O - z_P) & O & -(x_O - x_P) \\ (y_A - y_P) & -(x_A - x_P) & 0 \end{bmatrix}^{-1} \begin{Bmatrix} \dot{x}_O \\ \dot{y}_O \\ \dot{z}_A \end{Bmatrix} \end{aligned} \quad (3.71)$$

This equation can be considered as the constraint equation in velocities for the rolling joint. In practice, these equations must be numerically evaluated.

In the case of accelerations, it is necessary to differentiate equations (3.64)-(3.69), considering that P is the mathematical point of contact:

$$\ddot{x}_O = \dot{\omega}_y (z_O - z_P) - \dot{\omega}_z (y_O - y_P) + \omega_y (\dot{z}_O - \dot{z}_P) - \omega_z (\dot{y}_O - \dot{y}_P) \quad (3.72)$$

$$\ddot{y}_O = \dot{\omega}_z (x_O - x_P) - \dot{\omega}_x (z_O - z_P) + \omega_z (\dot{x}_O - \dot{x}_P) - \omega_x (\dot{z}_O - \dot{z}_P) \quad (3.73)$$

$$\ddot{z}_O = \dot{\omega}_x (y_O - y_P) - \dot{\omega}_y (x_O - x_P) + \omega_x (\dot{y}_O - \dot{y}_P) - \omega_y (\dot{x}_O - \dot{x}_P) \quad (3.74)$$

$$\ddot{x}_A = \dot{\omega}_y (z_A - z_P) - \dot{\omega}_z (y_A - y_P) + \omega_y (\dot{z}_A - \dot{z}_P) - \omega_z (\dot{y}_A - \dot{y}_P) \quad (3.75)$$

$$\ddot{y}_A = \dot{\omega}_z (x_A - x_P) - \dot{\omega}_x (z_A - z_P) + \omega_z (\dot{x}_A - \dot{x}_P) - \omega_x (\dot{z}_A - \dot{z}_P) \quad (3.76)$$

$$\ddot{z}_A = \dot{\omega}_x (y_A - y_P) - \dot{\omega}_y (x_A - x_P) + \omega_x (\dot{y}_A - \dot{y}_P) - \omega_y (\dot{x}_A - \dot{x}_P) \quad (3.77)$$

From equations (3.72), (3.73), and (3.77), the angular acceleration vector $\dot{\boldsymbol{\omega}}$ can be found. It will be a function of the coordinates, the velocities, and vector $\boldsymbol{\omega}$. By substituting in the three remaining equations, one obtains three ratios between the natural accelerations of points A and O. Vector $\boldsymbol{\omega}$, which can be replaced by the equation (3.70), intervenes in these ratios. In the resulting equation, the only unknown term is the pole velocity $\dot{\mathbf{r}}_p$. Point O' is the center of curvature of the intersection of the disc plane with the contact surface (line (a) in Figure 3.26). Point P is a mathematical point that instantaneously moves along the line of intersection. Of the three components of vector $\boldsymbol{\omega}$ at point P, only one perpendicular to the disc plane produces a velocity at this point. This velocity will be:

$$\dot{\mathbf{r}}_P = (\dot{\mathbf{r}}_O - (\dot{\mathbf{r}}_O \cdot \mathbf{u}) \mathbf{u}) \frac{PO'}{PO} \quad (3.78)$$

All the ratios are defined. The only thing left to do is to formulate the constraint equations for the three-dimensional wheel. These equations are more complicated than in the planar case, and they should be numerically formulated.

Sliding. In the case that there is sliding, only the component of the velocity of material point P in a direction normal to the surfaces in contact should be cancelled. In this case, the joint permits five degrees of freedom. One way of entering this constraint equation is by expressing the equations in a local system of coordinates located at the point of contact and only cancelling the normal component. Another possibility is to project the previous equations by means of their scalar product for a unit vector in the direction normal to the surfaces at the point of contact. As this value does not depend on the natural coordinates, it does not increase the degree of the polynomial equations obtained.

For the three-dimensional wheel, the problem is more complicated than in the planar case and should be solved by considering the center of the wheel, the unit vector normal to the wheel on which it is located, and the contact surface.

Likewise, in the case of the three-dimensional wheel, it is possible to substitute the rolling or sliding joint for one or more equivalent holonomic joints, remembering the differential geometry of the surfaces in contact and the three-dimensional generalization of the Euler-Savary formula. This approach is more complicated than in the planar case, and will not be developed here.

References

- Agrawal, O.P., "Dynamic Analysis of Multibody Systems Using Tangent Coordinates", Pergamon Press, *The theory of Machines and Mechanisms* (Proc. 7th World Congress), pp. 533-536, (1987).
- Agrawal, O.P. and Saigal, S., "Dynamic Analysis of Multibody Systems Using Tangent Coordinates", *Computers & Structures*, Vol. 31, pp. 349-355, (1989).
- García de Jalón, J., Avello, A., and Jiménez, J.M., "Basis for the Nullspace of the Jacobian Matrix in Constrained Multibody Systems", Pergamon Press, *The Theory of Machines and Mechanisms* (Proc. 7th World Congress), pp. 503-504, (1987).
- Huston, R.L., *Multibody Dynamics.*, Butterworth-Heinemann.,(1990).
- Ider, S.K. and Amirouche, F.M.L., "Coordinate Reduction in the Dynamics of Constrained Multibody Systems– A New Approach", *ASME Journal of Applied Mechanics*, Vol. 55, pp. 899-904, (1988).
- Kamman, J.W. and Huston, R.L., "Dynamics of Constrained Multibody Systems", *ASME Journal of Applied Mechanics*, Vol. 51, pp. 899-903, (1984).
- Kane, T.R. and Levinson, D.A., *Dynamics: Theory and applications*, McGraw-Hill Book Company, (1985).

- Kim, S.S. and Vanderploeg, M.J., "QR Decomposition for State Space Representation of Constrained Mechanical Dynamic Systems", *ASME Journal on Mechanisms, Transmissions and Automation in Design*, Vol. 108, pp. 176-182, (1986).
- Liang, C.G. and Lance, G.M., "A Differentiable Null Space Method for Constrained Dynamic Analysis", ASME paper No. 85-DET-86, (1985).
- Mani, N.K., Haug, E.J., and Atkinson, K.E., "Application of Singular Value Decomposition for Analysis of Mechanical System Dynamics", *ASME Journal on Mechanisms, Transmissions and Automation in Design*, Vol. 107, pp. 82-87, (1985).
- Serna, M.A., Avilés, R., and García de Jalón, J., "Dynamic Analysis of Plane Mechanisms with Lower-Pairs in Basic Coordinates", *Mechanism and Machine Theory*, Vol. 17, pp. 397-403, (1982).
- Singh, R.P. and Likins, P.W., "Singular Value Decomposition for Constrained Dynamic Systems", *ASME Journal of Applied Mechanics*, Vol. 52, pp. 943-948, (1985).
- Strang, G., *Linear Algebra and its Applications*, Academic Press, New York, (1976).
- Tsai, L.-W. and Morgan, A.P., "Solving the Kinematics of the Most General Six- and Five-Degree-of-Freedom Manipulators by Continuation Methods", *ASME Journal of Mechanisms, Transmissions and Automation in Design*, Vol. 107, pp. 189-200, (1985).

Problems

- 3/1 Starting from the constraint equations resulting from Problem 2/1, write the velocity equations of the mechanism shown in the Figure P3/1 when modeled with:
a) Relative coordinates. b) Reference point coordinates. c) Natural coordinates. d) Mixed coordinates, with relative coordinates in all the pairs.
- 3/2 Write the velocity and acceleration equations for the mechanism shown in the Figure.

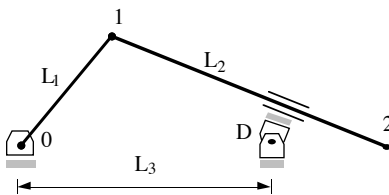


Figure P3/1.

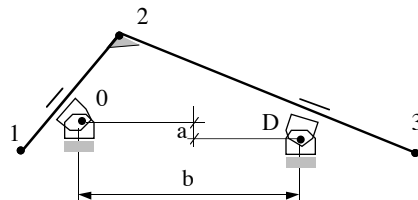


Figure P3/2.

- 3/3 Assuming that there is rolling with no slipping between the disk and the rod and using the set of mixed coordinates chosen in Problem 2/2 (natural and relative), write the Newton-Raphson iteration equations for the finite displacement problem. Write the driving constraint equations for the cases when the rotation of the disk and the relative rotation in joint 1 are the externally driven variables.

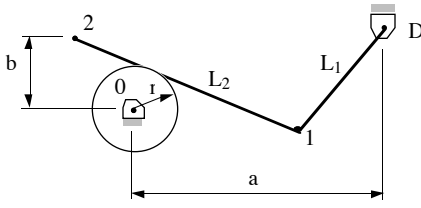


Figure P3/3.

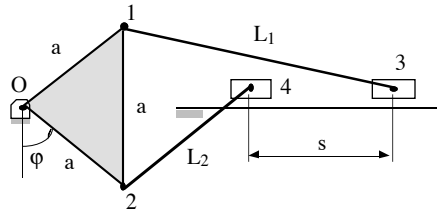


Figure P3/4.

3/4 Write the velocity equations for the mechanism of the figure.

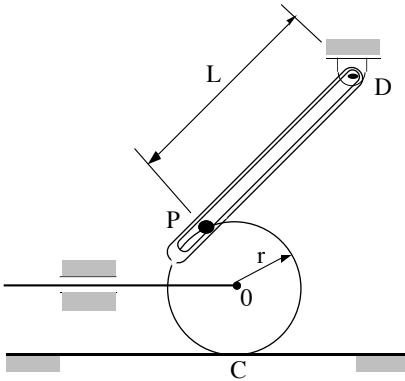


Figure P3/5.

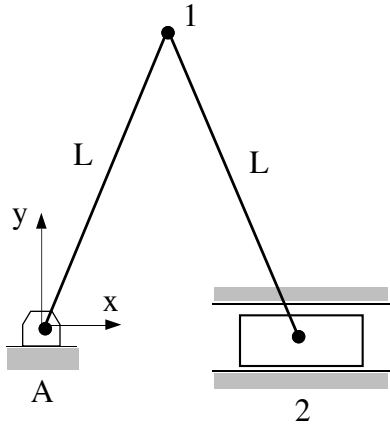


Figure P3/6

- 3/5 The wheel on the figure rolls without slipping. Use mixed coordinates (natural and relative) and find the velocity equations. It is suggested that the contact between the wheel and the ground be modeled by means of a rack and pinion type of kinematic joint.
- 3/6 The figure shows a slider-crank mechanism with the two elements of equal length. Write the velocity equations of this mechanism when the driven velocity is:
 a) x -velocity of point 1. b) x -velocity of point 2.
 What will happen to the rank of the Jacobian matrix when both bars are in the vertical position?
- 3/7 For the slider-crank mechanism of Problem 3/6, find analytically the matrix \mathbf{R} (null-space of the Jacobian matrix without driving constraints) both for the general position and for the case when both bars are placed vertically (singular position).
- 3/8 Find the velocity equations of the quick return mechanism of the figure when using mixed coordinates, so that the input angle φ is directly related to the output distance s .

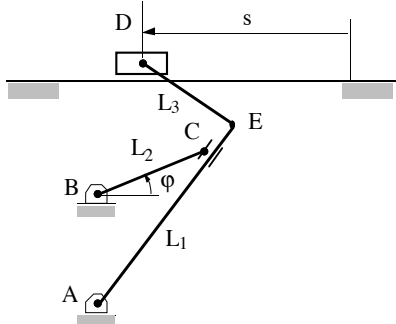


Figure P3/8.

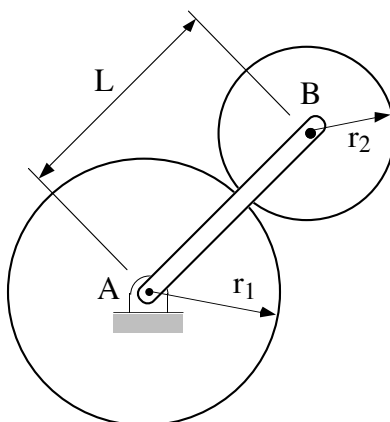


Figure P3/9.

3/9 The centers of the two gears shown in the figure are connected by means of a rod with point A being fixed. Considering mixed coordinates and using the constraint equations of Problem 2/6, find the equations that relate the angular velocities (relative or absolute) of the three elements.

3/10 Consider the mechanism in the figure and find the constraint equations that relate the velocities and accelerations of the angles φ_1 and φ_2 with the parameter s , that measures the relative position between elements 3 and 4 and its derivatives with respect to time.

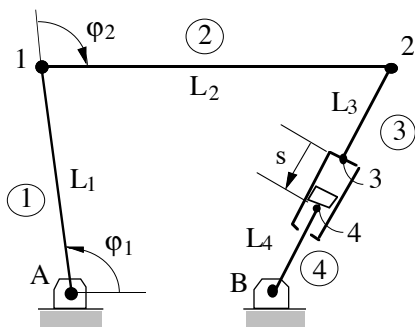


Figure P3/10.

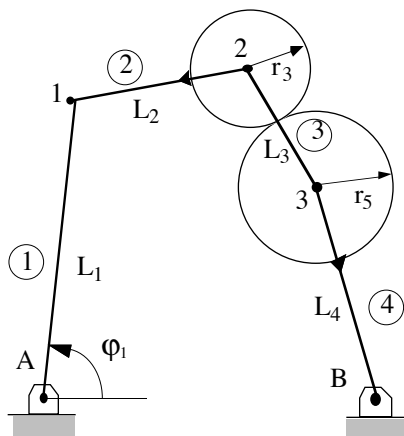


Figure P3/11.

3/11 Consider the mechanism shown (See Problem 2/8 for the constraint equations) to be modeled with natural coordinates. Rods 2 and 4 are attached to the gears with radius

r_3 and r_5 whose centers are connected by means of rod 3. Write the velocity equations for the complete system in terms of the input angle φ_1 .

- 3/12 Determine in the mechanism shown the constraint equations that relate the velocities and accelerations of coordinate s with the angle φ .

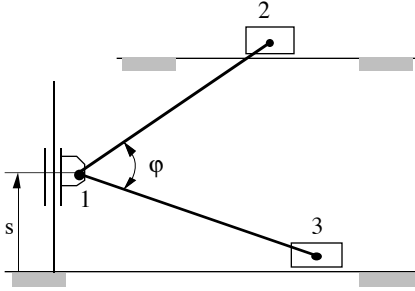


Figure P3/12.

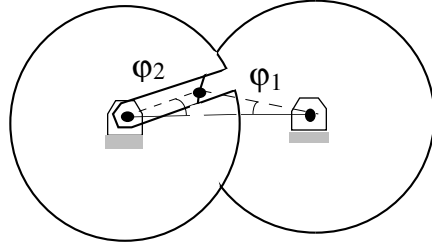


Figure P3/13.

- 3/13 Consider the Geneva wheel of the figure. Using natural coordinates, find the equations that relate the velocities of input angle φ_1 with the output angle φ_2 .

- 3/14 Find for the clam-shell bucket of the figure the equations that relate the speed of the control cable v with the angle θ .

- 3/15 The figure shows the frame $A12B$ that can rotate about the fixed axis AB by the action of the string attached to point 2 that goes through a pulley located at C (See Problem 2/16). Find the equations that allow one to relate the angular velocity ψ with the cable speed \dot{s} at the pulley.

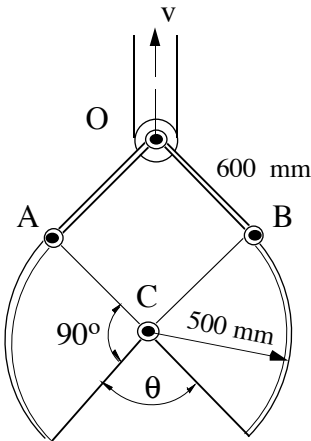


Figure P3/14.

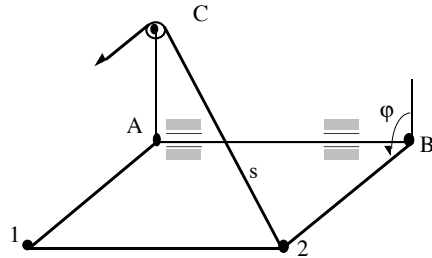


Figure P3/15.

3/16 The ends of a slender rod of length $\sqrt{2}$ move on the sides of a cube with sides of unit length. Find the equation that relates the speed of both ends of the rod.

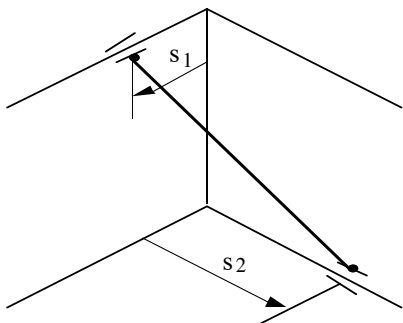


Figure P3/16.

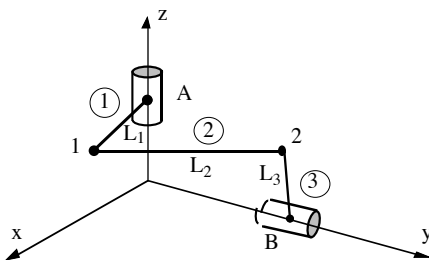


Figure P3/17.

3/17 Use natural coordinates and the result of Problem 2/18 to find the Jacobian matrix of the *RSSR* mechanism shown in the figure. How many arithmetic operations are necessary to evaluate this Jacobian matrix?

3/18 The mechanism shown has a revolute joint, a spherical joint, and a composite joint RC. Find the constraint equations that relate the time derivatives of angle ϕ and distance s .

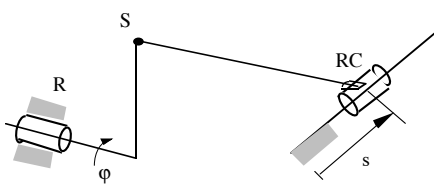


Figure P3/18.

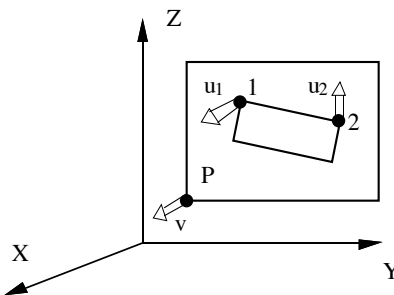


Figure P3/19.

3/19 For the 3-D planar joint shown in the figure, find the constraint equations for positions, velocities, and accelerations. The plane is defined by point P and its normal unit vector \mathbf{v} , and the body is defined by points 1 and 2, and unit vectors \mathbf{u}_1 and \mathbf{u}_2 . Discuss what difficulties may arise with other possible body positions and/or configurations.

3/20 Find the velocity constraint equations of the gyroscope shown, considering the angles of relative motion (See Problem 2/20).

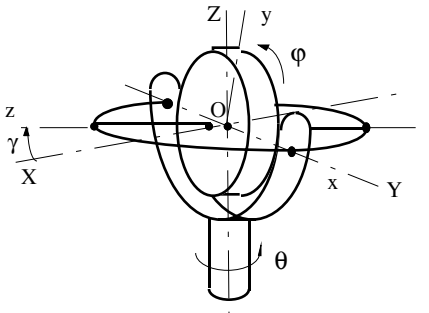


Figure P3/20.

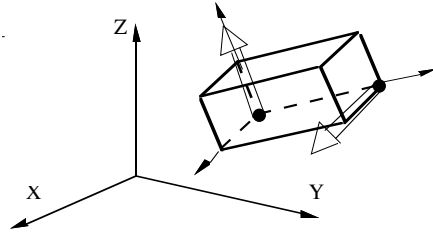


Figure P3/21.

- 3/21 Find the six kinematic constraint equations (and their derivatives) that allow one to guide kinematically the three translational displacements and the three roll, pitch and yaw rotations (in the local reference frame) of the rigid body of the figure.
- 3/22 A six degree of freedom spatial manipulator is depicted in Figure 3.13 (See also Problem 2/21). Use natural coordinates to find the finite displacement Newton-Raphson iteration equations.