Other topics in kinematics

Forward kinematics

The forward kinematics problem for a serial-chain manipulator is to find the position and orientation of the end-effector relative to the base given the positions of all of the joints and the values of all of the geometric link parameters. Often, a frame fixed in the end-effector is referred to as the *tool frame,* and while fixed in the final link N, it in general has a constant offset in both position and orientation from frame N. Likewise, a *station frame* is often located in the base to establish the location of the task to be performed. This frame generally has a constant offset in its pose relative to frame 0, which is also fixed in the base. A more general expression of the forward kinematics problem is to find the relative position and orientation of any two designated members given the geometric structure of the robotic mechanism and the values of a number of joint positions equal to the number of degrees of freedom of the mechanism. The forward kinematics problem is critical for developing manipulator coordination algorithms because joint positions are typically measured by sensors mounted on the joints and it is necessary to calculate the positions of the joint axes relative to the fixed frame. In practice, the forward kinematics problem is solved by calculating the transformation between a coordinate frame fixed in the end-effector and another coordinate frame fixed in the base, i. e., between the tool and station frames. This is straightforward for a serial chain since the transformation describing the position of the end-effector relative to the base is obtained by simply concatenating transformations between frames fixed in adjacent links of the chain. The convention for the geometric representation of a manipulator presented in Sect. 2.4 reduces this to finding an equivalent 4x4 homogeneous transformation matrix that relates the spatial displacement of the end-effector coordinate frame to the base frame. For the example serial-chain manipulator shown in Fig. 2.3 and neglecting the addition of tool and station frames, the transformation is

$$^{0}T_{6} = {}^{0}T_{1}\,{}^{1}T_{2}\,{}^{2}T_{3}\,{}^{3}T_{4}\,{}^{4}T_{5}\,{}^{5}T_{6}$$

Table 2.8 contains the elements of 0T6 that are calculated using Table 2.7 and (2.44). Once again, homogeneous transformations provide a compact notation, but are computationally inefficient for solving the forward kinematics problem. A reduction in computation can be achieved by separating the position and orientation portions of the transformation to eliminate all multiplications by the 0 and 1 elements of the matrices. In Chap. 3, calculations are made using the spatial vector notation briefly introduced here in Sect. 2.2.6 and explained in detail in Sect. 3.2. This approach does not employ homogeneous transformations, but rather separates out the rotation matrices and positions to achieve computation efficiency. Table 3.1 provides the detailed formulas, with the product of spatial transforms particularly relevant to the forward kinematics problem. Kinematic trees are the general structure of robotic mechanisms that do not contain closed loops, and the forward kinematics of tree structures are addressed in Chap. 3. The forward kinematics problem for closed chains is much more complicated because of the additional constraints present. Solution methods for closed chains are included in Chap. 18.

Forward Instantaneous Kinematics

The forward *instantaneous* kinematics problem for a serial-chain manipulator is: given the positions of all members of the chain and the rates of motion about all the joints, find the total velocity of the end effector. Here the rate of motion about the joint is the angular velocity of rotation about a revolute joint or the translational velocity of sliding along a prismatic joint. The total velocity of a member is the velocity of the origin of the coordinate frame fixed to it combined with its angular velocity. That is, the total velocity has six independent components and therefore, completely represents the velocity field of the member. It is important to note that this problem definition includes an assumption that the pose of the mechanism is completely known. In most situations, this means that either the forward or inverse position kinematics problem must be solved before the forward instantaneous kinematics problem can be addressed. The same is true of the inverse *instantaneous* kinematics problem discussed in the following section. The forward instantaneous kinematics problem is important when doing acceleration analysis for the purpose of studying dynamics. The total velocities of the members are needed for the computation of Coriolis and centripetal acceleration components.

Inverse Instantaneous Kinematics

The important problem from the point of view of robotic coordination is the inverse *instantaneous* kinematics problem. The inverse instantaneous kinematics problem for a serial chain manipulator is: given the positions of all members of the chain and the total velocity of the end-effector, find the rates of motion of all joints. When controlling a movement of an industrial robot that operates in the point-to-point mode, it is not only necessary to compute the final joint positions needed to assume the desired final hand position. It is also necessary to generate a smooth trajectory for motion between the initial and final positions. There are, of course, an infinite number of possible trajectories for this purpose. However, the most straightforward and successful approach employs algorithms based on the solution of the inverse instantaneous kinematics problem. This technique originated in the work of Whitney [2.55] and of Pieper [2.34].
Using the product of exponentials formula for the forward kinematics map, it is possible to develop a geometric algorithm to solve the inverse kinematics problem. This method was originally presented by Paden [85] and built on the unpublished work of Kahan [46].

Tasks minimum description

In current robot controllers, the desired trajectory of the end-effector is described by a sequence of frames. However, in many industrial applications, it is not necessary to completely specify the location of the end-effector frame and the task could be described by a reduced number of coordinates. For example:

- when the manipulated object is *symmetric*: for a spherical object, it is not necessary to specify the orientation; likewise, the rotation of a cylindrical object about its axis can be left free;

- releasing an object into a *container*: if the end-effector is ahead above the container, only an approach distance has to be specified; the task is thus described by a translational component;
- *transferring* objects from one point to another with arbitrary orientation; the task can be described by three translational components;
- placing a cylindrical object on a conveyor: the only orientation constraint is that the principal axis of the cylinder is horizontal; if the end-effector is already above the conveyor, the task could be described by two continents (one vertical translation and one rotation).

Static wrench transmission

A general force system can be shown to be equivalent to a single force together with a moment acting about its line of action. This is called a wrench. There is a deep isometry between the geometries of systems of wrench axes and that of systems of instantaneous screw axes [2.59]. Static wrench analysis of a manipulator establishes the relationship between wrenches applied to the end-effector and forces/torques applied to the joints. This is essential for controlling a manipulator's interactions with its environment. Examples include tasks involving fixed or quasi-fixed workpieces such as inserting a component in place with a specified force and tightening a nut to a prescribed torque. More information can be found in Chaps. 9 and 37. Through the principle of virtual work, the relationship between wrenches applied to the end-effector and forces/torques applied to the joints can be shown to be

$$\tau = J^T f$$

where τ is the n-dimensional vector of applied joint forces/torques for an n-degree-of-freedom manipulator and f is the spatial force vector in which n and f are the component vectors of torques and forces, respectively, applied to the end-effector, both expressed in the coordinate frame relative to which the Jacobian is also expressed. Thus, in the same way the Jacobian maps the joint rates to the spatial velocity of the end-effector, its transpose maps the wrenches applied to the end-effector to the equivalent joint forces/torques. As in the velocity case, when the Jacobian is not square, the inverse relationship is not uniquely defined.

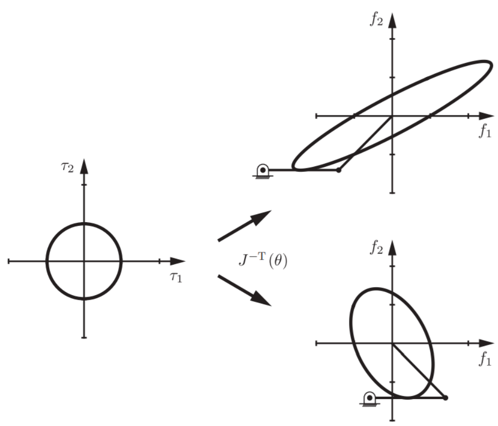


Figure 5.5: Force ellipsoids for two different postures of the 2R planar open chain.

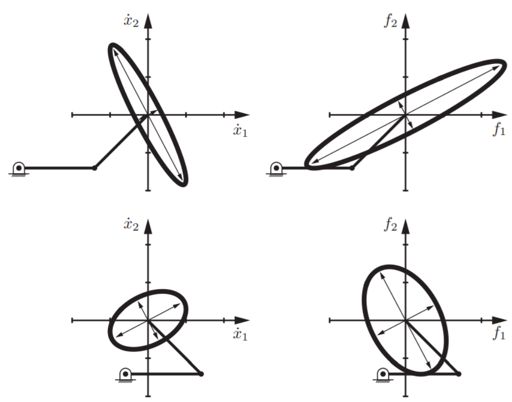


Figure 5.6: Left-hand column: Manipulability ellipsoids at two different arm configurations. Right-hand column: The force ellipsoids for the same two arm configurations.

Inverse kinematics

The inverse kinematics problem for a serial-chain manipulator is to find the values of the joint positions given the position and orientation of the end-effector relative to the base and the values of all of the geometric link parameters. Once again, this is a simplified statement applying only to serial chains. A more general statement is: given the relative positions and orientations of two members of a mechanism, find the values of all of the joint positions. This amounts to finding all of the joint positions given the homogeneous transformation between the two members of interest. In the common case of a six-degree-of-freedom serial chain manipulator, the known transformation is $^{0}T_{6}$.It is clear that the inverse kinematics problem for serial-chain manipulators requires the solution of sets of *nonlinear equations*. In the case of a six-degree of-freedom manipulator, three of these equations relate to the position vector within the homogeneous transformation, and the other three relate to the rotation matrix. In the latter case, these three equations cannot come from the same row or column because of the dependency within the rotation matrix. With these nonlinear equations, it is possible that no solutions exist or multiple solutions exist [2.33]. For a solution to exist, the desired position and orientation of the end-effector must lie in the workspace of the manipulator. In cases where solutions do exist, they often cannot be presented in closed form, so numerical methods are required.

Adaptation of DH parameters: conventions of Khalil and Dombre

the foundational convention that has been adapted in a number of different ways, one of which is the convention introduced by Khalil and Dombre.
In all of its forms, the convention requires only four rather than six parameters to locate one coordinate frame relative to another.
four parameters consist of two link parameters, the link length ai and the link twist ˛i, and two joint parameters, the joint offset di and the joint angle i. This parsimony is achieved through judicious placement of the coordinate frame origins and axes such that the xO axis of one frame both intersects and is perpendicular to the zO axis of the following coordinate frame. The convention is applicable to robotic mechanisms consisting of revolute and prismatic joints, so when multiple-degree-of-freedom joints are present, they are modeled as combinations of revolute and prismatic joints
There are essentially four different forms of the convention for locating coordinate frames in a robotic mechanism. Each exhibits its own advantages by managing trade-offs of intuitive presentation.

- In the original **Denavit and Hartenberg** convention, joint i is located between links i and $i + 1$, so it is on the outboard side of link i. Also, the joint offset d_i and joint angle θ_i are measured along and about the $i - 1$ joint axis, so the subscripts of the joint parameters do not match that of the joint axis.
- **Waldron and Paul** modified the labeling of axes in the original convention such that joint i is located between links $i - 1$ and i in order to make it consistent with the base member of a serial chain being member 0. This places joint i at the inboard side of link i and is the convention used in *all of the other modified versions*. Furthermore, Waldron and Paul addressed the mismatch between subscripts of the joint parameters and joint axes by placing the $\hat{z}_i$. axis along the $i + 1$ joint axis. This, of

course, relocates the subscript mismatch to the correspondence between the joint axis and the $\hat{z}$ axis of the coordinate frame.

- **Craig** eliminated all of the subscript mismatches by placing the $\hat{z}_i$ axis along joint i, but at the expense of the homogeneous transformation ${}^{i-i}T_i$ being formed with a mixture of joint parameters with subscript i and link parameters with subscript $i - 1$.
- **Khalil and Dombre** introduced another variation similar to Craig's except that it defines the link parameters a_i and α_i along and about the $\hat{x}_{i-1}$ axis. In this case, the homogeneous transformation ${}^{i-1}T_i$ is formed only by parameters with subscript i, and the subscript mismatch is such that a_i and α_i indicate the length and twist of link $i - 1$ rather than link i. Thus, in summary, the advantages of the convention used compared to the alternative conventions are that the $\hat{z}$ axes of the coordinate frames share the common subscript of the joint axes and the four parameters that define the spatial transform from coordinate frame i to coordinate frame $i - 1$ all share the common subscript i

- The convention for serial chain mechanisms is shown in Fig. 2.2 and summarized as follows. The numbering of bodies and joints follows the convention: The N moving bodies of the robotic mechanism are numbered from 1 to N. The number of the base is 0.
- The N joints of the robotic mechanism are numbered from 1 to N, with joint i located between members $i - 1$ and i.
-

With this numbering scheme, the attachment of coordinate frames follows the convention:

  - The $\hat{z}_i$ axis is located along the axis of joint i,
  - The $\hat{x}_{i-1}$ axis is located along the common normal between the $\hat{z}_{i-1}$ and $\hat{z}_i$ axes.

Using the attached frames, the four parameters that locate one frame relative to another are defined as:

- ai is the distance from $\hat{z}_{i-1}$ to $\hat{z}_i$ along $\hat{x}_{i-1}$,
- α_i is the angle from $\hat{z}_{i-1}$ to $\hat{z}_i$ about $\hat{x}_{i-1}$,
- d_i is the distance from $\hat{x}_{i-1}$ to $\hat{x}_i$ along $\hat{z}_i$, i
- θ_i is the angle from $\hat{x}_{i-1}$ to $\hat{x}_i$ about $\hat{z}_i$

The geometric parameters for the example manipulator shown in Fig. 2.3 are listed in Table 2.7. All of the joints of this manipulator are revolutes, and joint 1

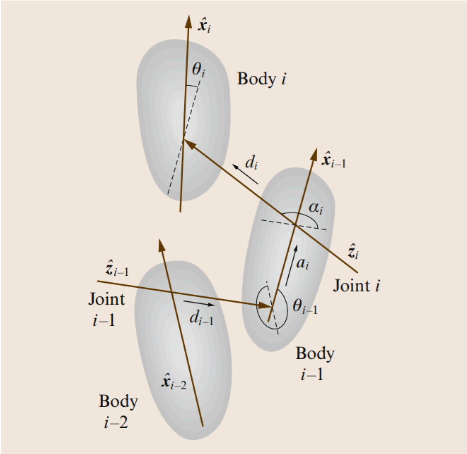
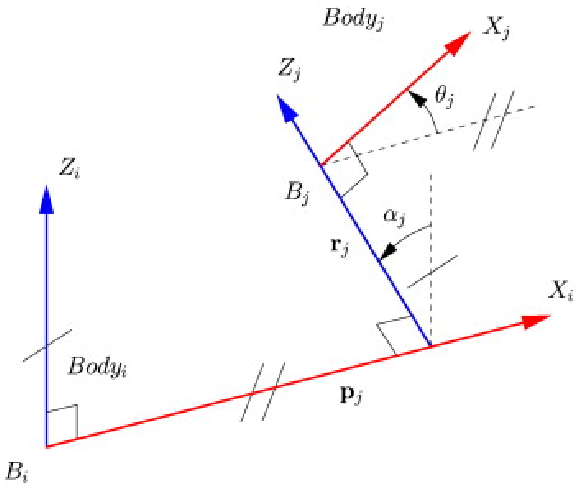


Fig. 2.2 Schematic of the numbering of bodies and joints in a robotic mechanism, the convention for attaching coordinate frames to the bodies, and the definitions of the four parameters, a_i, α_i, d_i, and θ_i, that locate one frame relative to another



$$\mathbf{Rot}(\hat{\boldsymbol{x}}_{i-1}, \alpha_i)\mathbf{Trans}(\hat{\boldsymbol{x}}_{i-1}, a_i)\mathbf{Rot}(\hat{z}_i, \theta_i)$$
$$\mathbf{Trans}(\hat{z}_i, d_i)\ ,$$

Given parameters one can write corresponding homogeneous transformation matrix

$$^{i}T_j = \begin{pmatrix} c\theta_j & -s\theta_j & 0 & d_j \\ c\alpha_j s\theta_j & c\alpha_j c\theta_j & -s\alpha_j & -r_j s\alpha_j \\ s\alpha_j s\theta_j & s\alpha_j c\theta_j & c\alpha_j & r_j c\alpha_j \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

POE in calibration

Kinematic model generation can be achieved through the conventional Denavit–Hartenberg (DH) parameterization [22.2–4, 17] or the coordinate-free local product-of-exponential (POE) approach [22.8, 9, 19]. However, the DH method does not provide a clear distinction between the arranging sequence of modules in a robot chain, and it is an initial position-dependent representation. The local POE formulation of the kinematics and dynamics based on the theory of Lie groups and Lie algebras for rigid motion in SE(3) and SO(3) can avoid this problem. Furthermore, the POE representation can avoid the singularity conditions that frequently occur in the kinematic calibration formulated by the DH method [22.20]. Thus, POE representations provide a uniform and well behaved method for handling the inverse kinematics of both calibrated and un-calibrated robot systems. In local POE modeling, the joint axes are described in the local module (body) coordinate systems, it is progressive in constructing the kinematic models, so it conveniently resembles the assembling action of the physical modular robot components. The machining tolerance, compliance, and wear of the connecting mechanism due to frequent module reconfiguration may introduce errors in positioning the end effector. Hence, kinematic calibration is a must for modular robots. In the POE calibration model, the robot errors are assumed to be in the initial positions of the consecutive modules because the local POE model is a zero reference method. Based on linear superposition and differential transformation, a 6-parameter error model can be established for serial-type robots [22.19]. This model can be obtained through the automatic generation process. An iterative least-square algorithm employed to find the error parameters to be corrected. The corrected kinematic model is then updated in the robot controller for operation. The simulation and experiment have shown that the proposed method can improve the position accuracy up to two orders of magnitude, or to the nominal repeatability of the robot after calibration with measurement noise. A typical 6-DOF articulate-type modular robot can reach a position accuracy of 0:1 mm compared to an accuracy of 1 mm before the calibration [22.20] A formulation of the dynamic model of modular manipulators starts from a recursive Newton–Euler algorithm [22.21, 22]. The generalized velocity, acceleration, and forces can be expressed in terms of linear operations on se.3/ [22.23]. Based on the relationship between the recursive formulation and the closed form Lagrangian formulation for serial-robot dynamics discussed in [22.24, 25], the AIM can assist in the construction of the closed-form equation of motion of a modular robot in any generic topology with redundant and non-redundant configurations [22.19].

Architectures

A survey of industrial robots has shown that only the following *five* structures [Liégeois 79] are manufactured:

- *anthropomorphic shoulder* represented by the first RRR structure shown in Figure 1.7, like PUMA from Unimation, Acma SR400, ABB IRBx400, Comau Smart-3, Fanuc (S-xxx, Arc Mate), Kuka (KR 6 to KR 200), Reis (RV family), Staubli (RX series), etc.;
- - spherical shoulder RRP: "Stanford manipulator" and Unimation robots (Series 1000,2000,4000);
- *RPR shoulder* corresponding to the first RPR structure shown in Figure 1.7: Acma-H80, Reis (RH family), etc. The association of a wrist with one revolute degree of freedom of rotation to such a shoulder can be found frequently in the industry. The resulting structure of such a robot is called SCARA (Selective Compliance Assembly Robot Arm) (Figure 1.8). It has several applications, particularly in planar assembly. SCARA, designed by Sankyo, has been manufactured by many other companies: IBM, Bosch, Adept, etc.;
- *Cylindrical shoulder* RPP: Acma-TH8, AFMA (ROV, ROH), etc.;
- *Cartesian shoulder* PPP: Acma-PSO, IBM-7565, Sormel-Cadratic, OlivettiSIGMA. More recent examples: AFMA (RP, ROP series), Comau P-Mast, Reis (RL family), SEPRO, etc. The second RRR structure of Figure 1.7, which is equivalent to a spherical joint, is generally used as a wrist. Other types of wrists are shown in Figure 1.9 [Delignieres 87]. A robot, composed of a shoulder with three degrees of freedom and a spherical wrist, constitutes a classical six degrcc-of-freedom structure (Figure 1.10). Note that the position of the center of the spherical joint depends only on the configuration of joints 1, 2 and 3. We will see in Chapter 4 that, due to this property, the inverse Terminology and genera] definitions 9 geometric model, providing the joint variables for a given location of the end effector, can be obtained analytically for such robots.

  According to the survey carried out by the French Association of Industrial Robotics (AFRI) and RobAut Journal [Pages 98], the classification of robots in France (17794 robots), with respect to the number of degrees of freedom, is as follows: 4.5% of the robots have three degrees of freedom, 27% have four, 9% have five and 59.5% have six or more. As far as the architecture of the shoulder is concerned, there is a clear dominance of the RRR anthropomorphic shoulder (65.5%), followed by the Cartesian shoulder (20.5%), then the cylindrical shoulder (7%) and finally the SCARA shoulder (7%).

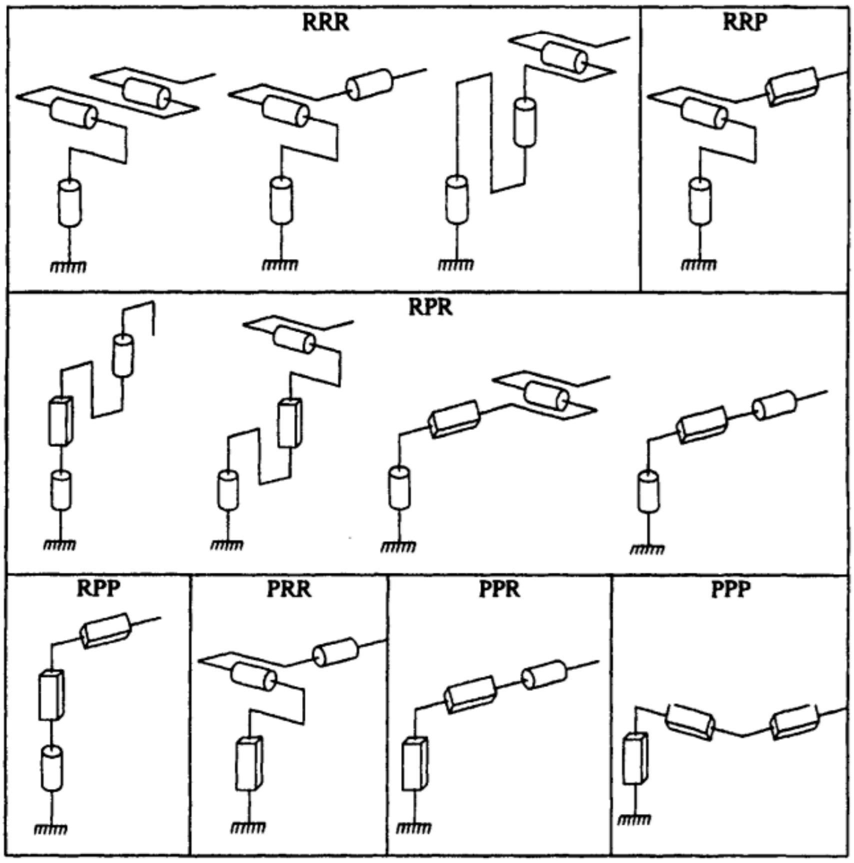


Figure 1.7. *Architectures of the shoulder (from [Milenkovic 83])*

Inverse kinematics methods

Closed-Form Solutions

Closed-form solutions are desirable because they are faster than numerical solutions and readily identify all possible solutions. The disadvantage of closed-form solutions is that they are not general, but robot dependent. The most effective methods for finding closed-form solutions are ad hoc techniques that take advantage of particular geometric features of specific mechanisms. In general, closed-form solutions can only be obtained for six-degree-of-freedom systems with special kinematic structure characterized by a large number of the geometric parameters defined in Sect. 2.4 being zero-valued. Most industrial manipulators have such structure because it permits more efficient coordination software. Sufficient conditions for a six-degree of-freedom manipulator to have closed-form inverse kinematics solutions are [2.34–36]:

1. Three consecutive revolute joint axes intersect at a common point, as in a spherical wrist.
2. Three consecutive revolute joint axes are parallel. Closed-form solution approaches are generally divided into algebraic and geometric methods.

Continuation methods

involve tracking a solution path from a start system with known solutions to a target system whose solutions are sought as the start system is transformed into the target system. These techniques have been applied to inverse kinematics problems [2.43], and special properties of polynomial systems can be exploited to find all possible solutions [2.44].

Algebraic Methods

Algebraic methods involve identifying the significant equations containing the joint variables and manipulating them into a soluble form. A common strategy is reduction to a transcendental equation in a single variable such as,

where C1, C2, and C3 are constants. The solution to such an equation is

Special cases in which one or more of the constants are zero are also common. Reduction to a pair of equations having the form,

C1 cos i C C2 sin i C C3 D 0 ; (2.49) C1 sin i C2 cos i C C4 D 0 ; (2.50)

is another particularly useful strategy because only one solution results

Geometric Methods

Geometric methods involve identifying points on the manipulator relative to which position and/or orientation can be expressed as a function of a reduced set of the joint variables. This often amounts to decomposing the spatial problem into separate planar problems. The resulting equations are solved using algebraic manipulation. The two sufficient conditions for existence of a closed-form solution for a six-degree-of-freedom manipulator that are listed above enable the decomposition of the problem into inverse position kinematics and inverse orientation kinematics. This is the decomposition into regional and orientation structures discussed in Sect. 2.5, and the solution is found by rewriting

(2.46), 0 T6 6 T5 5 T4 4 T3 D 0 T1 1 T2 2 T3 : (2.52)

$$ {}^{0}T_6 {}^{6}T_5 {}^{5}T_4 {}^{4}T_3 = {}^{0}T_1 {}^{1}T_2 {}^{2}T_3 $$

The example manipulator in Fig. 2.3 has this structure, and its regional structure is commonly known as an articulated or anthropomorphic arm or an elbow manipulator. The solution to the inverse position kinematics problem for such a structure is summarized in Table 2.9. Because there are two solutions for θ_1 and likewise two solutions for both θ_2 and θ_3 corresponding to each θ_1 solution, there are a total of four solutions to the inverse position kinematics problem of the articulated arm manipulator. The orientation structure is simply a spherical wrist, and the corresponding solution to the inverse orientation kinematics problem is summarized in Table 2.10. Two solutions for θ_5 are given in Table 2.10, but only one solution for both θ_4 and θ_6 corresponds to each. Thus, the inverse orientation kinematics problem of a spherical wrist has two solutions. Combining the regional and orientation structures, the total number of inverse kinematics solutions for the manipulator in Fig. 2.3 is eight.

Numerical Methods

Unlike the algebraic and geometric methods used to find closed-form solutions, numerical methods are not robot dependent, so they can be applied to any kinematic structure. The disadvantages of numerical methods are that they can be slower and in some cases, they do not allow computation of all possible solutions. For a six-degree-of-freedom serial-chain manipulator with only revolute and prismatic joints, the translation and rotation equations can always be reduced to a polynomial in a single variable of degree not greater than 16 [2.37]. Thus, such a manipulator can have as many as 16 real solutions to the inverse kinematics problem [2.38]. Since closed-form solution of a polynomial equation is only possible if the polynomial is of degree four or less, it follows that many manipulator geometries are not soluble in closed form. In general, a greater number of nonzero geometric parameters corresponds to a polynomial of higher degree in the reduction. For such manipulator structures, the most common numerical methods can be divided into categories of symbolic elimination methods, continuation methods, and iterative methods.

Symbolic Elimination Methods

Symbolic elimination methods involve analytical manipulations to eliminate variables from the system of nonlinear equations to reduce it to a smaller set of equations. Raghavan and Roth [2.39] used dialytic elimination to reduce the inverse kinematics problem of a general 6R serial-chain manipulator to a polynomial of degree 16 and to find all possible solutions. The Part A | 2.8 roots provide solutions for one of the joint variables, while the other variables are computed by solving linear systems. Manocha and Canny [2.40] improved the numerical properties of this technique by reformulating the problem as a generalized eigenvalue problem. An alternative approach to elimination makes use of Gröbner bases [2.41, 42].

Iterative Methods

A number of different iterative methods can be employed to solve the inverse kinematics problem. Most of them converge to a single solution based on an initial guess, so the quality of that guess greatly impacts the solution time. *Newton–Raphson* methods provide a fundamental approach that uses a first-order approximation of the original equations. *Pieper* [2.34] was among the first to apply the method to inverse kinematics, and others have followed [2.45, 46]. Optimization approaches formulate the problem as a *nonlinear optimization* problem and employ search techniques to move from an initial guess to a solution [2.47, 48]. Resolved motion rate control converts the problem to a differential equation [2.49], and a modified predictor– corrector algorithm can be used to perform the joint velocity integration [2.50]. Control-theory-based methods cast the differential equation into a control problem [2.51]. Interval analysis [2.52] is perhaps one of the most promising iterative methods because it offers rapid convergence to a solution and can be used to find all possible solutions. For complex mechanisms, the damped least-squares approach [2.53] is particularly attractive, and more detail is provided in Chap. 10.

Paden-Kahan sub problems

Using the product of exponentials formula for the forward kinematics map, it is possible to develop a geometric algorithm to solve the inverse kinematics problem. This method was originally presented by Paden [85] and built on the unpublished work of Kahan [46]. To solve the inverse kinematics problem, we first solve a number of subproblems which occur frequently in inverse solutions for common manipulator designs. One then seeks to reduce the full inverse kinematics problem into appropriate subproblems whose solutions are known. One feature of the subproblems presented here is that they are both geometrically meaningful and numerically stable. Note that this set of subproblems is by no means exhaustive and there may exist manipulators which cannot be solved using these canonical problems. Additional subproblems are explored in the exercises. For each of the subproblems presented below, we give a statement of the geometric problem to be solved and a detailed solution. On a first reading of this section, it may be difficult to understand the relevance of the specific subproblems presented here. We recommend that the first time reader skip the solutions until she sees how the subproblems are used in the examples presented later in this section.