An Introduction to Robotics: Mechanical Aspects

MICHEL GERADIN

AN INTRODUCTION TO ROBOTICS: MECHANICAL ASPECTS

Pierre DUYSINX
and
Michel GERADIN

University of Liège

Novembre 2004

Contents

Chapter 1

INTRODUCTION

The object of the following sections is to introduce some terminology and to briefly define the main topics which will be covered during the course.

1.1 Origin of the name robot

The word *robot* was coined in 1920 by the Czech author K. Capek in his play R.U.R. (Rossum's Universal Robot), and is derived from the word *robota*, meaning worker.

Later, an *industrial robot* has been defined as reprogrammable multifunctional manipulator, designed to move materials, parts, tools, or other specialized devices by means of variable programmed motions and to perform a variety of other tasks. In a broader context, the term robot also includes manipulators that are activated directly by an operator. This includes manipulators used in nuclear experiences, medical investigation or surgery as well as robots used for under-water exploration or works.

More generally, an industrial robot has been described by the International Organization for Standardization (ISO) as follows: A machine formed by a mechanism including several degrees of freedom, often having the appearance of one or several arms ending in wrist capable of holding a tool, a workpiece, or an inspection device.

In particular, its central control unit must use a memorizing device and it may sometimes use sensing or adaptation appliances to take into account environment and circumstances. These multipurpose machines are generally designed to carry out a repetitive function and can be adapted to other operations.

Introduced in the early 1960s, the first industrial robots were used in hazardous operations, such as handling toxic and radioactive materials, loading and unloading hot workpieces from furnaces and handling them in foundries. Some rule-of-thumb applications for robots are the three D's (dull, dirty, and dangerous including demeaning but necessary tasks), and the three H's (hot, heavy, and hazardous). From their early uses in worker protection and safety in manufacturing plants, industrial robots have been further developed and have become important components in manufacturing process and systems. They have helped improve productivity, increase product quality, and reduce labor costs. Computer-controlled robots were commercialized in the early 1970s, with the first robot controlled by a minicomputer appearing in 1974.

1.2 Mechanical role of a robot manipulator

Whatever be the function assigned to it (handling, painting, assembly, welding...) a robot manipulator is designed mechanically to *locate in space* a *tool* often called *effector*.

The effector can simply be a gripper designed to grasp a part, a painting gun, or may consist of any other tool.

The *geometric location* of the effector is generally quite arbitrary in the reachable range, called *workspace of the manipulator*, and is continuously changing with time: it follows a *path* or *trajectory* corresponding to the specified *task*. At a crude but important level, it is completely described through a sequence of *positions* of one given point on the effector and its *orientation* about these points. At a higher level, the *trajectory planning* might also take account of *environmental* constraints.

In order to describe the position and orientation of the effector in space, it will be necessary to attach to it a coordinate system, or *frame*, and describe the position and orientation of this frame to some reference system. *Frame transformations* will thus play a fundamental role throughout this course.

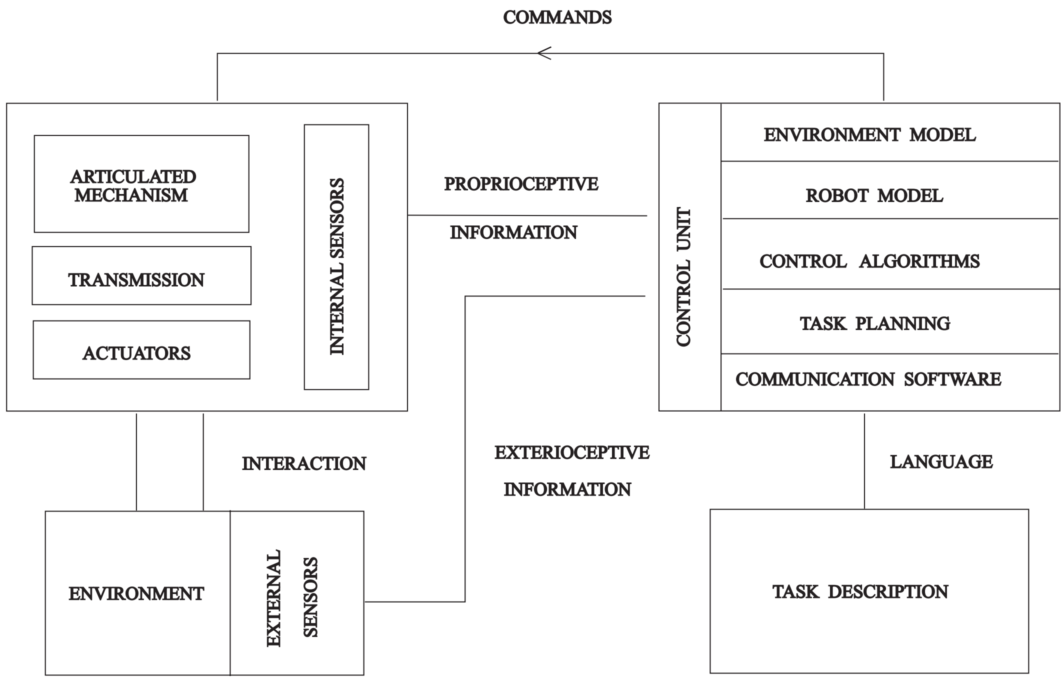


Figure 1.3.1: General structure of a robot manipulator integrated in its environment

1.3 General structure of a robot manipulator

To fulfill the function just described, the general structure of a robot manipulator, when considered in its working environment, may be decomposed into five main components interacting together as described in figure 1.3.1.

1. The *mechanical structure*, or *articulated mechanism*, is ideally made of rigid *members* or *links articulated* together through *mechanical joints*. It carries at its end the *tool* or *effector*.

2. The *actuators* provide the mechanical power in order to act on the mechanical structure against gravity, inertia and other external forces to modify the configuration and thus, the geometric location of the tool. The actuators can be of electric, hydraulic or pneumatic type and have to be *controlled* in the appropriate manner. The choice of their control mode is one of the fundamental options left to the mechanical engineer.

3. The *mechanical transmission devices* (such as gear trains) connect and adapt the actuators to the mechanical structure. Their role is twofold: to *transmit* the mechanical efforts from the power sources to the mechanical joints and to *adapt* the actuators to their load.

4. The *sensors* provide senses to the robot. They can take for example the form of *tactile*, *optical* or *electrical devices*. According to their function in the system they may be classified in two groups:

   - *Proprioceptive sensors* provide information about the mechanical configuration of the manipulator itself (such as velocity and position information);

- *Exteroceptive sensors* provide information about the environment of the robot (such as distance from an obstacle, contact force...)

5. The *control unit* assumes simultaneously three different roles:

   - An *information role*, which consists of collecting and processing the information provided by the sensors;
   - A *decision role*, which consists of planning the geometric motion of the manipulator structure starting from the *task* definition provided by the human operator and from the status of both the system and its environment transmitted by the sensors;
   - A *communication role* , which consists to organize the flow of information between the control unit, the manipulator and its environment.

1.4 Structure of the control unit

In order to assume the functions just described, the control unit must dispose of *softwares* and *knowledge bases* such as

- a *model (kinematic and/or dynamic) of the robot*, which expresses the relationship between the input commands to the actuators and the resulting motion of the structure;
- a *model of the environment*, which describes geometrically the working environment of the robot. It provides information such as the occurrence of zones where collisions are likely to occur and allows to plan the path accordingly;
- *control algorithms* which govern the robot motion at a lower level and are responsible for the mechanical response of the structure and its actuators (assuming thus *position and velocity control* with prescribed accuracy and stability characteristics);
- a *communication protocol*, which assumes the management of the messages (in shape, priority...) exchanged between the various components of the system.

The control unit may have either a *centralized* architecture, in which case the same processor assumes all the functions described above, or a *hierarchical* organization, in which case the system is organized around a *master* unit that assigns to each one of the *slave units* some of the functions to be performed. For example, lower level functions such as position and velocity control of the actuators are often assumed by slave processors.

1.5 Industrial robots at the present day

An industrial robot such as described above has the theoretical ability to adapt itself to any change on the environment on which it operates. However, most of the industrial robots available today, even at a research level, possess this characteristic of adaptability only to a limited extent. This, because the information, decision and communication roles of the control unit which have been identified above are assumed only up to a very low degree.

To improve the situation, much progress has still to be made in various areas such as sensor design and processing, world modelling, programming methods, decision making and system integration.

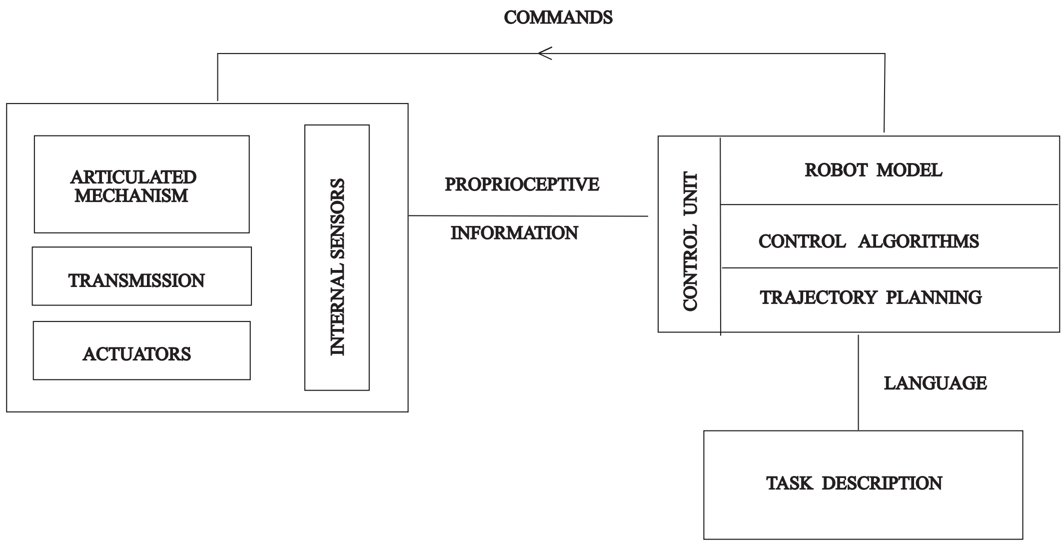


Figure 1.5.1: Schematic representation of present industrial robots

The present day industrial robots correspond thus to a simpler structure. In particular, their interaction with the environment is often almost inexistent, so that they can be regarded as simple robot manipulators which are programmed according to a fixed description of task (Figure 1.5.1).

1.6 The mechanical aspects of robotics

Robots are required to have high *mobility* and *dexterity* in order to work in a large reachable range, or *workspace*, and access crowded spaces, handle a variety of workpieces, perform flexible tasks.

It is the parallelism with the mechanical structure of the human arm which gives them these properties.

Like the human arm, a robot mechanical structure is most often a purely *serial linkage* composed of cantilever beams forming a sequence of *links*, or *members*, connected by *hinge joints*. Such a structure has inherently poor mechanical stiffness and accuracy and is thus not naturally appropriate for heavy and/or high precision applications.

In order to benefit from the theoretical high mobility and dexterity of the serial linkage, these difficulties must be overcome by advanced design, modelling and control techniques.

The *geometry* of manipulator arms, due to its serial nature, is described by *complex nonlinear equations*. Effective analytical tools are needed to construct and understand the geometric and kinematic model of a manipulator. Kinematics is an important area of robotics research, since it had traditionally focused on mechanisms with very limited number of degrees of freedom (generally, single input) while robotic arms have developed the need for models of multi-degree of freedom mechanisms.

Dynamic behavior of robot manipulators is also a complex subject, for several reasons.

- The kinematic complexity of articulated systems affects also the dynamic model: the equations governing the dynamics of a robot arm are thus highly coupled and nonlinear; the motion of each joint is significantly influenced by the motion of all other joints.

- The *gravity and inertia loads* applied to each member vary widely with the configuration. *Coriolis an centrifugal* effects become significant when the manipulator arm moves either at high speed or in a low gravity environment.
- *Structural flexibility* (either concentrated in the joints or distributed over the links) greatly affects the dynamic behavior of the system by introducing supplementary degrees of freedom of elastic nature.
- Undesirable effects such as *structural damping*, friction and backlash at joints are difficult to model and quantify and are also responsible for significant departure from the expected dynamic properties of the system.

From what preceeds, the role of the mechanical engineer in robotics appears to be important at least at three levels:

- At the *overall design* level, he has to define the general architecture of the mechanical structure in order to fulfill correctly the mobility and dexterity requirements of its assigned function. This task requires a strong background in kinematics of mechanisms. The currently available computer aided design software tools are of great help to the mechanical engineer to compare various possible designs.
- At the *detailed design* level, the role of the mechanical engineer consists to design and size the mechanical parts of the system in order to satisfy quality and performance criteria with respect to accuracy, reliability, life cycle time, mechanical strength, lightweight, operating speed, etc.
- At both *design and control* levels, the mechanical engineer has a *modelling* role which consists of building various realistic models of mechanical behavior of the robot. These will be used mainly for two purposes: for off-line simulation of the system with given degree of fidelity and for its control using adequate control modes. The mechanical engineer will also have to define the convenient analytical tools to plan and describe mathematically the trajectories of the system.

1.7 Multidisciplinary aspects of robotics

Let us note that the mechanical structure of a robot is only the 'visible part of the iceberg'. Robotics is an essentially multidisciplinary field in which engineers from various horizons such as electrical engineering, electronics and computer science play equally important roles.

Therefore it is fundamental for the mechanical engineer who specializes in robotics to learn to dialogue with them by getting a sufficient level of understanding in these disciplines.

Bibliography

[1] P. COIFFET, M. CHIROUZE, *An introduction to robot technology*, Kogan Page, London, 1983.

[2] B. GORLA, M. RENAUD, *Modèles des robots manipulateurs. Application à leur commande*, Cepadues Editions, Toulouse, 1984.

[3] P. COIFFET, *Les robots: modélisation et commande*, tome 1, Hermes Publishing, Paris, 1981.

[4] J. J. CRAIG, *Introduction to robotics: Mechanics and Control*, Addison-Wesley, 1985.

[5] H. ASADA, J. J. SLOTINE, *Robot analysis and control*, John Wiley and Sons, 1985.

Chapter 2

BASICS OF ROBOTICS TECHNOLOGY

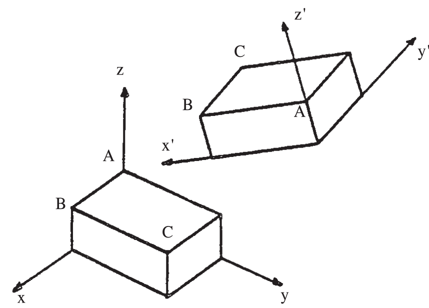


Figure 2.1.1: Position and orientation of a rigid body specified through the position of three non collinear points

2.1 The Mechanical Structure of a Robot

The mechanical structure of a robot manipulator may be regarded as a multibody system, i.e. a system of bodies or *links*, rigid in theory, interconnected together by joints or *kinematic pairs*.

2.1.1 Degrees of freedom of a rigid body

Let us consider a rigid body freely located in space. It is possible to specify both its position and orientation in space by giving the position of three non collinear points attached to it.

This is a set of 9 non independent parameters, since the coordinates of these points are linked through three relationships expressing that the distance between them is fixed.

In the general case, the geometric location of a free rigid body is described with 6 parameters or *degrees of freedom* (DOF) which may be classified in two categories:

- 3 independent *translation parameters* which define the location in space of a reference point on the solid;
- 3 independent *rotation parameters* which define the orientation in space of the solid.

It will be seen in the chapter devoted to the kinematics of the rigid body that the three translation DOF may be expressed indifferently in terms of any system of coordinates (cartesian, cylindrical, polar, etc.) while the orientation of the body is specified through a set of rotation parameters such as Bryant angles, Euler angles or parameters, etc. (as it will be seen later).

2.1.2 Joints and kinematic constraints

Joints or *kinematic pairs* connect the different bodies or *links* of the multibody system to one another. They impose constraints between the motions of the connected bodies.

Basically, each joint can be replaced by a set a algebro-differential constraints. Thus the effect of m constraints restraining the motion of a system of N bodies is the following: Instead of having $6N$ variables describing the system behaviour, the multibody system has only $6N - m$ *degrees of freedom* left.

2.1.3 Generalized coordinates

The choice of a minimal set of independent variables, which are able to describe fully the multibody system is of a primary importance in mechanics. This minimal set of variables is named as the set of *generalized coordinates* of the system. The choice has a great impact on how easily one will be able to solve the motion's equations afterwards. For example an advantages of generalized coordinates is that constraints may often be automatically (implicitly) be satisfied by a suitable choice. This eliminates the needs for writing out separately $6N$ equations of motion subject to m equations of constraints.

2.2 Kinematic Pairs

2.2.1 Number of degrees of freedom of the joint

One major characteristic of a joint is the number of geometric constraints it imposes between the two connected bodies. *The number of degrees of freedom DOF* is equal to the minimal number of parameters necessary to describe the position of C_2 relatively to C_1. A kinematic pair is thus characterized by 5, 4, 3, 2, or 1 DOF and its *class* is defined as the complement to 6 of its number of DOF. Clearly, the class of the joint is the number of geometric constraints imposed by the joint.

2.2.2 Classification of joints

Kinematic pairs between two rigid bodies C_1 and C_2 can be also formally classified according to various criteria such as:

- *Holonomic and non holonomic constraints*

  The kinematic pair may introduce a constraint, which depends only on the positions of the two bodies and not on their velocities. This results in purely algebraic constraints. These constraints are said to *holonomic constraints*.

  Otherwise when pair introduce contraints that are dependent of the velocities too, they are said to be *non holonomic* and the contraints result in a algebro-differential relations. A disk rolling without slipping on a surface is a non-holonomic constraint, while the fixed distance between two points of a rigid body is a holonomic one.

- *The type of relative motion allowed*

  The motion of any point on C_2 relatively to C_1 can be restrained to a line, or surface or be arbitrary in space;

- *The type of contact*

  The contact can be point -, linear - or surface contact;

- *The mode of closure*

  A kinematic pair is *self-closed* if the contact between bodies is guaranteed by the way the kinematic pair is realized. Otherwise, it can be *force-closed*, which means that an external force (such as provided either by gravity or through the compression of a spring) is necessary to maintain contact.

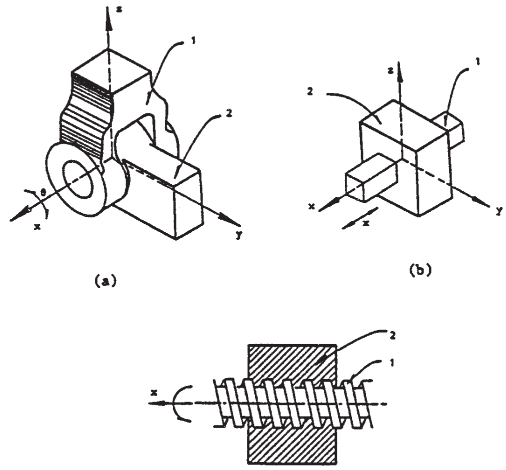


Figure 2.2.1: The three possible lower pairs (a) revolute joint (R); (b) prismatic joint (P); (c) screw joint (H) characterized by only one DOF and reversible motion

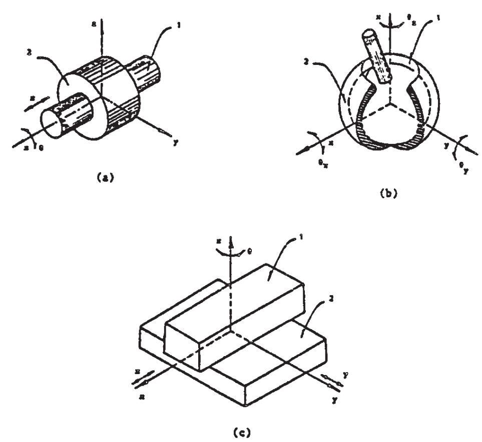


Figure 2.2.2: The three possible lower pairs characterized by more than one DOF and reversible motion : (a) cylindrical joint (C) with 2 DOF; (b) spherical joint (S) with 3 DOF; (c) planar joint (E) with 3 DOF

2.2.3 Higher and lower pairs

Following Reulaux, one can distinguish *higher* and *lower* kinematic pairs.

- In *lower pairs* the elements touch one another over a substantial region of a surface. There are only six lower pairs, namely the revolute joint (R), the prismatic joint (P), the screw joint or helical joint (H), the cylindrical joint (C), the planar joint (E), and the spherical or globular joint (S for spherical or G for globular). The six lower pairs are represented in figures 2.2.1 and 2.2.2. Figure 2.2.1 gives the three class 5 lower pairs. All three of them have only one DOF. In the form represented, the pairs are self-closed. Figure 2.2.2 illustrates the three other lower pairs that are of class 4 and class 3. In the implementation represented, the planar joint is not self-closed.

  Even if contact area has been used as the criterion for lower pairs by Reuleaux, the real concept lies in the particular kind of relative motion permitted between the two links. In lower kinematic pair, coincident points on C_1 and C_2 undergo relative motions that are similar. In other words, the relative motion produced by a lower pair can be said *reversible*, and an exchange of element from one link does not alter the relative motion of the parts. That is, the relative motion between links C_1 and C_2 will be the same no matter whether link C_1 or link C_2 is the moving link.

- In *higher pairs* the elements touch one another with point or line contact. The relative motion of the elements of of higher pairs is relatively complicated. An infinite number of higher pairs exist, and they may usually be replaced by a combination of lower pairs.

  Examples of point contact can be found in ball bearings and helical gears on non-parallel shafts. Also the Hooke or universal joint (U or T) can be seen as having an idealized unique contact point. Line contact is characteristic of cams, roller bearings and most gears. The rigid wheel is also and example of line contact.

  One must also notice that the two elements may touch as wheel at two non material body fixed surfaces, named the circle-point curve and the center-point curve along the actual axis of rotation.

2.2.4 Graphic representation of joints

Several norms propose a conventional graphic representation of kinematic pairs. As an example, the French norm AFNOR is given in figure 2.2.5. To the six pairs that we have just described, have been added the limiting cases of the *"rigid pair"* (0 DOF) and the *"free pair"* (6 DOF), as well as three other higher pairs: the *linear contact joint*, the *annular contact joint* and the *point contact joint*.

2.2.5 Joints used in robots

Let us note that most of these joints are not used in robot design, at least as far as the relative motion of links is concerned. In order to analyze the external architecture of robot manipulators, it is sufficient to consider

- the revolute joint (R), which produces a relative rotary motion;

- the prismatic joint (P), which produces a relative translation motion.

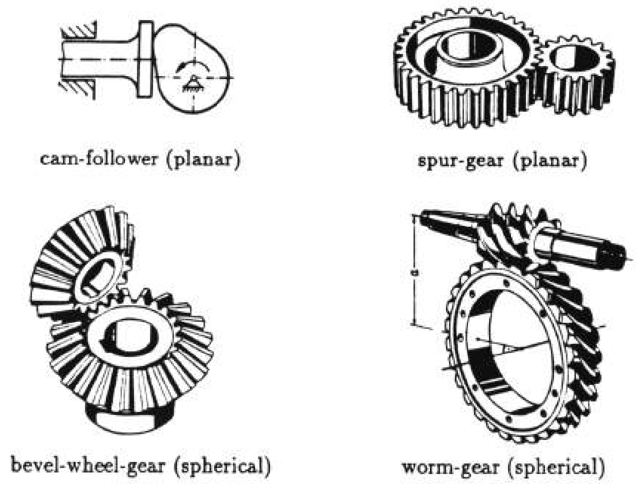


Figure 2.2.3: Examples of higher pairs

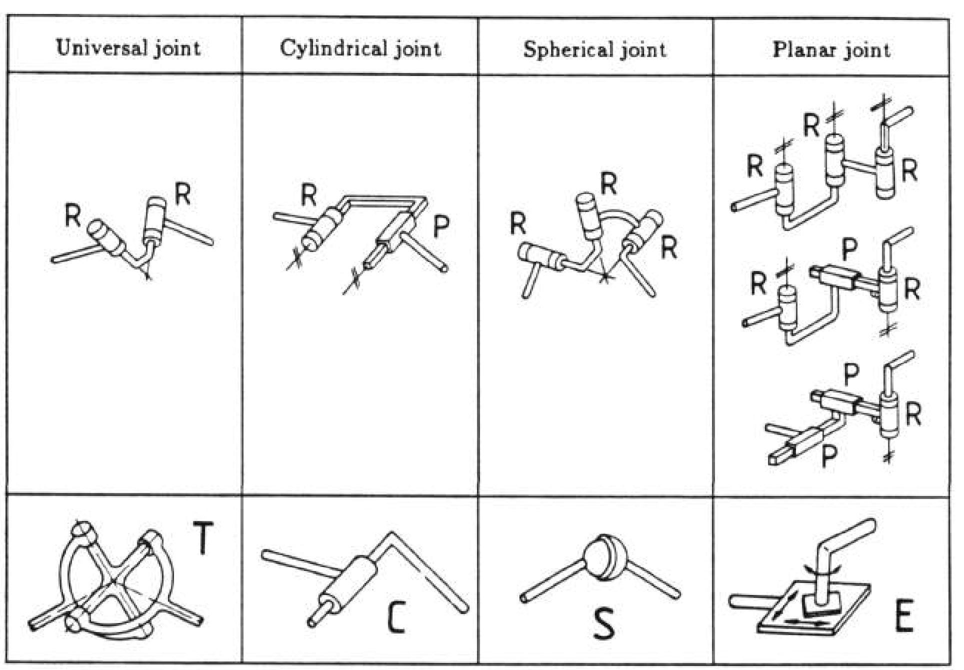
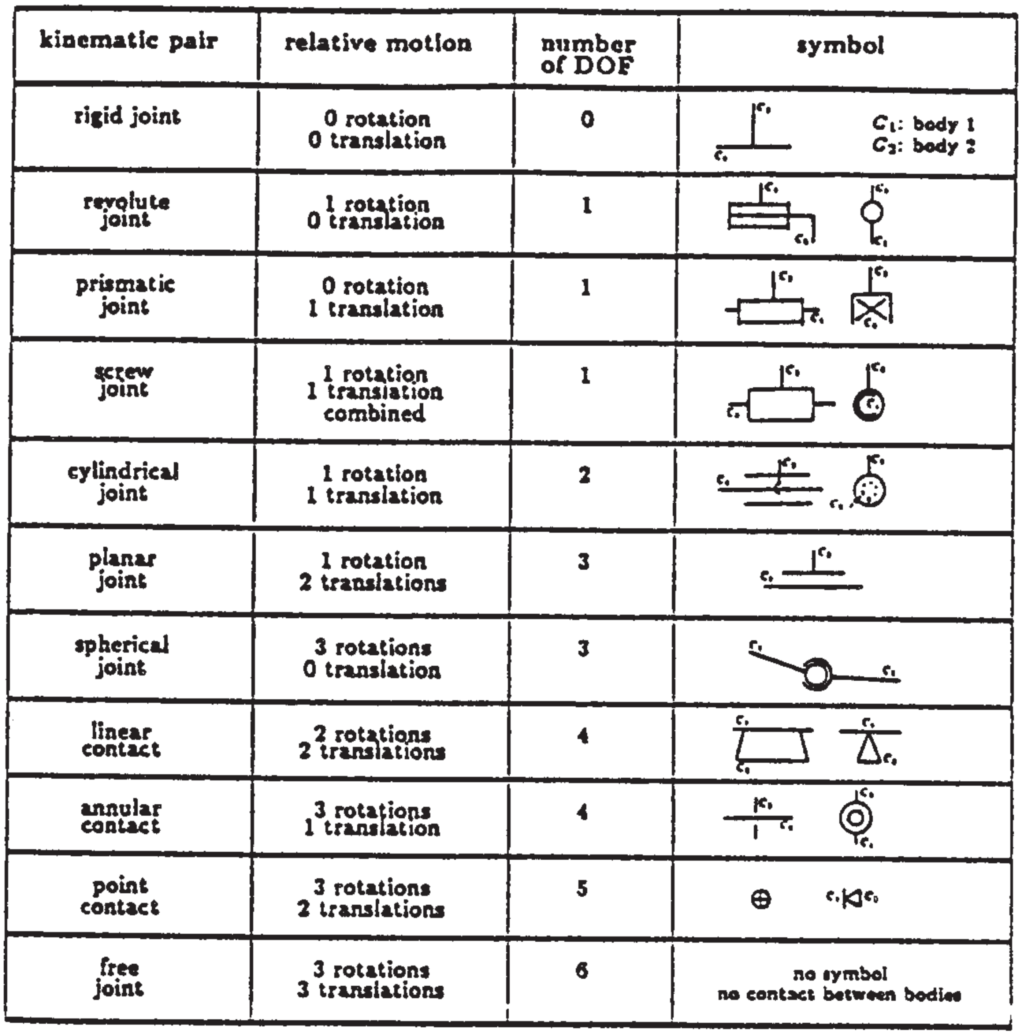


Figure 2.2.4: Building up joints from revolute and prismatic joints



kinematic pair	relative motion	number of DOF	symbol
rigid joint	0 rotation 0 translation	0	C_1: body 1 C_2: body 2
revolute joint	1 rotation 0 translation	1	
prismatic joint	0 rotation 1 translation	1	
screw joint	1 rotation 1 translation combined	1	
cylindrical joint	1 rotation 1 translation	2	
planar joint	1 rotation 2 translations	3	
spherical joint	3 rotations 0 translation	3	
linear contact	2 rotations 2 translations	4	
annular contact	3 rotations 1 translation	4	
point contact	3 rotations 2 translations	5	
free joint	3 rotations 3 translations	6	no symbol no contact between bodies

Figure 2.2.5: AFNOR E04-E015 representation of kinematic pairs

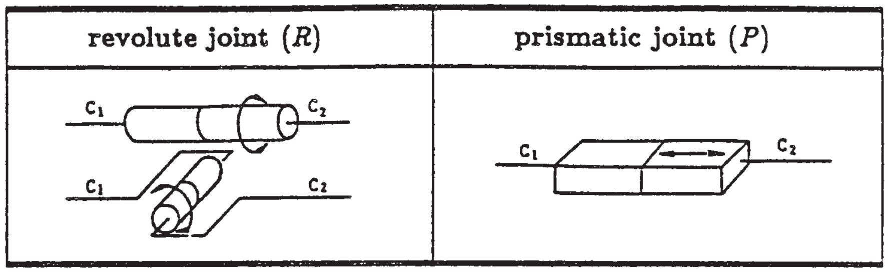


Figure 2.2.6: Schematic representation of revolute and prismatic pairs

Both pairs are commonly represented as shown on figure 2.2.6.

Other kinematic pairs may also appear when examining the mechanical structure of the the transmission components. For example, the screw joint has the interesting property to transform rotary motion into translation motion, and conversely; it allows thus to power a prismatic joint using an actuator with rotary output.

2.3 Topology of Kinematic Chains

Starting from a reference body C_0 forming the basis of the manipulator, the bodies C_i and the kinematic pairs L_i forming the mechanical structure of the robot can be attached together in a wide variety of manners. The *topology* or *morphology* of the kinematic chain is the study of the relative position of the different links and joints of the chain, and the way they are connected to each others. This characteristic has a major impact on its performance of the system .

2.3.1 Classification of robot topologies

The morphology of the manipulator can be classified into differents groups:

- *Simple open-tree structures* (figure 2.3.1)

  The simple open-tree structure is characterized by the fact that all the links and joints are connected in a row, that is every body C_i in the kinematic chains is connected only to the two neighboring elements C_{i-1} and C_{i+1}. Most of the industrial robots currently commercialized possess such a purely serial architecture.

- *Multiple open-tree structures* (figure 2.3.2)

  The multiple open-tree structure is rarely used in robotics. It offers the possibility to integrate in a single system several effectors, and is sometimes used for that reason. Let us note that such a morphology corresponds to that of the human body.

  The connectivity in this kind of architecture is such that the bodies C_i can no longer be numbered in sequence since some of the links are implied in more than two kinematic pairs. To describe it, it is necessary to use the concept of *graph*.

  Remark that in kinematical chains with open-tree structure (simple or multiple tree structures), there is only one path from each body to another one. Therefore it is possible to

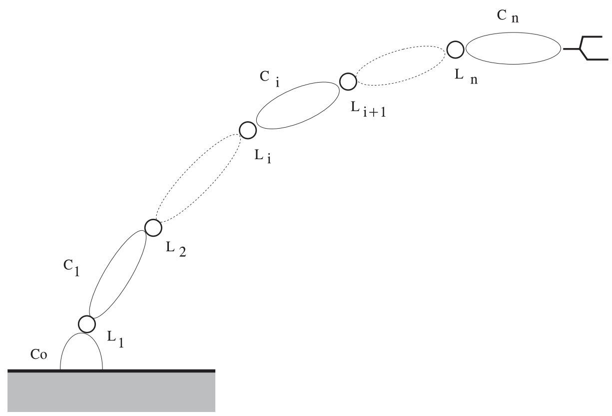


Figure 2.3.1: Simple open-tree structure

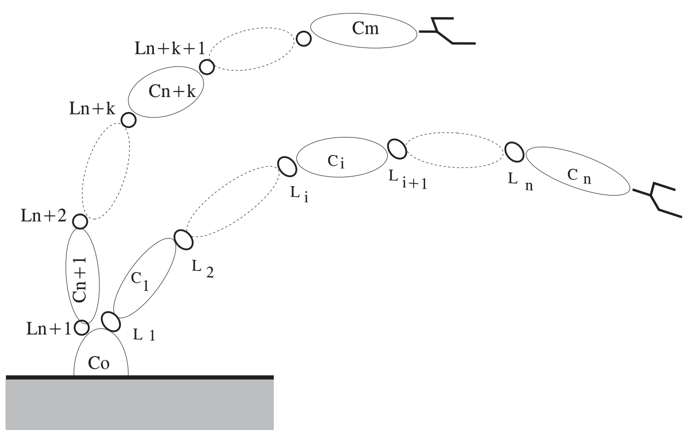


Figure 2.3.2: Multiple open-tree structure

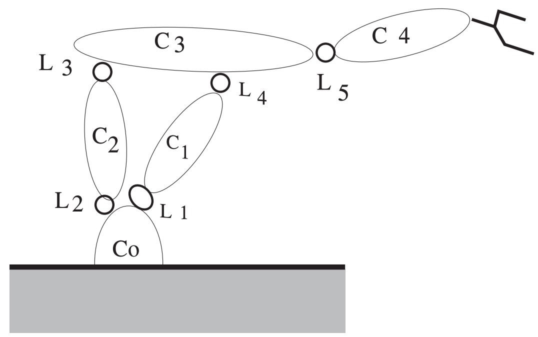


Figure 2.3.3: Multiply-connected structure

relate each body or joint to a following one clearly. Furthermore, if n_B is the number of bodies and if n_J is the number of joints, one shows that there are equal:

$$n_B = n_J \tag{2.3.1}$$

- *Multiply-connected or complex structures* (figure 2.3.3)

  The multiply connected (or *complex*) structure is characterized by the appearance of *mechanical loops*, i.e. paths which start from and come back to a same body after having followed certain sequences of links and joints.

  A significant number of industrial robots currently on the market have such a kinematic architecture. The use of mechanical loops allows one to build a structure with higher structural stiffness, and thus obtaining a greater working accuracy. However, multiply-connected structures have also generally a lower mobility. The importance of complex structure is growing nowadays very fast with the study and the introduction of parallel manipulators to new applications such as High Speed Machining and metrology of large size pieces.

  Let us also note that most manipulators exhibit an external architecture of simple open-loop type, but have to be regarded as multiply-connected structures as soon as one takes into account the model of the transmission devices.

  One can see that a kinematical chain with tree structure can be changed into an independent closed loop by introducing an additional joint. So if n_L is the number of closed-loops, one can see that:

$$n_L = n_J - n_B \tag{2.3.2}$$

  The kinematic analysis of multiply-connected systems implies being able to model structures of multiple open-tree type: indeed, any multiply-connected system may be reduced to a multiple open-tree linkage by removing a number of pairs equal to the number of mechanical loops. The closure conditions are then treated as separate constraints.

As underlined earlier, the morphology of the manipulator has a major impact on its characteristics. For instance, figure 2.3.4 displays three relatively common architectures of industrial robots. The first one is a simple open-tree structure in which all the kinematic pairs are of revolute type. It generates a workspace having the shape of a portion of sphere. The second one is

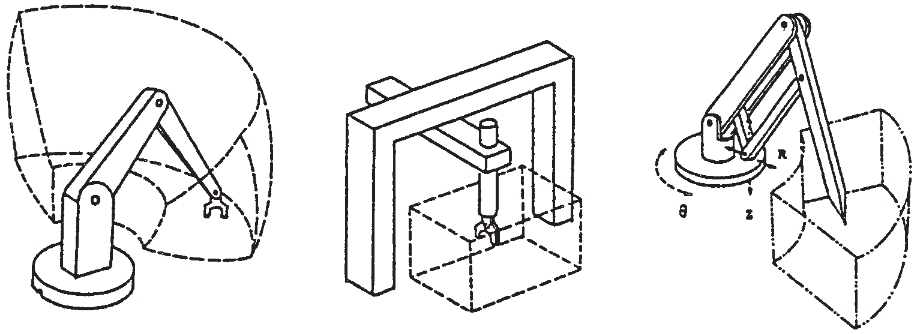


Figure 2.3.4: Common architectures of robot manipulators (a) simple open-tree structure with revolute joints (RRR) (b) simple open-tree structure with prismatic joints (PPP) (c) multiply-connected structure

also a simple open-tree structure, but has all prismatic joints. The workspace generated is thus a parallelipiped. The third one has a multiply-connected architecture with all revolute joints. It has a closed loop in the form of a parallelogram which gives to it the properties of a pantograph. Its 3-dimensional motion can be produced by rotating the waist and translating into vertical and radial directions one of the vertices of the pantograph. The resulting workspace has thus a cylindrical shape. It comes that the choice of a proper morphology of the manipulator must be chosen carefully when selecting a robot or designing it.

2.3.2 Description of simple open-tree structures

From what preceeds, it can be observed that any robot having a simple open-tree structure may be described by a sequence of n letters (n being the number of joints) R and P which define the sequence and the type of kinematic pairs crossed when progressing from the reference body to the effector. For example, the simple open-tree structures represented by figures 2.3.4(a) and 2.3.4(b) are respectively of types RRR and PPP.

2.4 Mobility Index and Number of dof for a Simple Open-Tree Manipulator

2.4.1 Mobility index and GRÜBLER formula

Except for robots with a mobile base, the reference body C_0 is fixed in space, while the other members C_1 to C_n are mobile. Before assembling the system, the position and orientation of the n links are completely arbitrary in space, and are thus defined in terms of $6n$ parameters.

Let us consider a kinematical chain with:

- n_B the number of bodies (without the base body, which is assumed to be fixed),
- n_J the number of joints,
- f_j the number of degrees of freedom of joint j, i.e. joint j is of class $c_j = 6 - f_j$.

The total number of degrees of freedom, i.e. the necessary number of free parameters to determine completely its geometric configuration of the system is given by the formula:

$$M = 6n_B - \sum_{j=1}^{n_J}(6 - f_j)$$

or

$$M = 6(n_B - n_J) + \sum_{j=1}^{n_J} f_j \tag{2.4.1}$$

With the total number of kinematical loops

$$n_L = n_J - n_B$$

the total number of degrees of freedom is:

$$M = \sum_{j=1}^{n_J} f_j - 6n_L \tag{2.4.2}$$

Equations (2.4.1) or (2.4.2) are known as Grübler's formula.
Based on Grübler's formula, one finds the Grübler's criterion of mechanisms. The following cases are distinguished:

- $M = 1$, mobility-1 mechanism;
- $M = 0$, statically determined structure;
- $M < 0$, statically undetermined structure.

Important remarks

In some cases formula (2.4.1) or (2.4.2) lead to wrong results. A first situation is the case of planar or spherical kinematical chains, in which each body is constrained to move in a plane or along the surface of a sphere. Then, each body has only 3 degrees of motion. Thus formula formula (2.4.1) and (2.4.2) become:

$$M = 3(n_B - n_J) + \sum_{j=1}^{n_J} f_j \tag{2.4.3}$$

$$M = \sum_{j=1}^{n_J} f_j - 3n_L \tag{2.4.4}$$

Formula (2.4.1) or (2.4.2) may also be erroneous when they are applied to complex architectures such as that of figure 2.3.4(c) since the formula assumes that all the constraints imposed by the joints are independent of one another. However sometimes, mechanical loops introduce redundancies between the geometric constraints of the different joints and thus, they do not necessarily reduce the effective mobility of the mechanism.

2.4.2 Number of dof of a simple open-tree structure

If we limit ourselves to a simple open-tree structures made of kinematic pairs of types R and P (class 5) for which $f_j = 1$ for all j, then the formula above reduces further to

$$M \;=\; n_B \;=\; n_J \tag{2.4.5}$$

This formula means that the mobility index of the manipulator is then equal to its number of joints. Therefore, since 6 parameters are necessary to describe an arbitrary position and orientation in space for the effector, a simple open-tree robot manipulator possesses normally 6 joints of either revolute or prismatic type.

2.4.3 Number of DOF of a manipulator

The number of DOF N of a robot manipulator is equal to the number of independent parameters which fix the location of the effector. It can be a function of the geometric configuration of the robot, but the following inequality is always verified

$$N \;\leq\; M \tag{2.4.6}$$

2.4.4 Joint space

To represent the configuration of the manipulator and the position of all its bodies, the most obvious solution is to use the *the joint variables* or *joint coordinates*, which are the degrees of freedom of the kinematical chain. Therefore they are sometimes known as *configuration variables* of the robot. In real robots, these degrees of freedom are motorized.

To each set of joint variables, one can attach a point in a hyper-space $\mathbb{R}^M$ of dimension M with M the number of joint variables (equal to the mobility index of the kinematical structure). The hyper-space $\mathbb{R}^M$ that is used to represent the configuration variable sets is the *joint space* of the robot.

2.4.5 Task space

The configuration of the effector, i.e. the tool of the robot, is usually given in terms of the cartesian coordinates of one reference point (called the Tool Center Point or TCP) and the orientation of the tool about this reference point. For 3-dimensional problems, the coordinates of the Tool Center Point can be associated to a point in $\mathbb{R}^3$, but for the orientation things are more complicated: the 3 independent rotation variables can be related to the group $SO(3)$ of all rotations in $\mathbb{R}^3$. Thus the tool configurations belongs to the space $\mathbb{R}^3 \times SO(3)$. This space of all tool configuration point is the *task space*. As $SO(3)$ is not a vector space, the task space is not a vector space either. If the tool has got other additional degrees of freedom related, for instance the configuration state - open or closed - of the gripper, they must also be added to the configuration space.

2.4.6 Redundancy

The case $n_R < M$ corresponds to the existence of a *redundancy* in the DOF of the system. In other words, a robot is *redundant* when the number of degrees of freedom of its effector is less than its index of mobility, i.e. the number of generalized degrees of freedom.

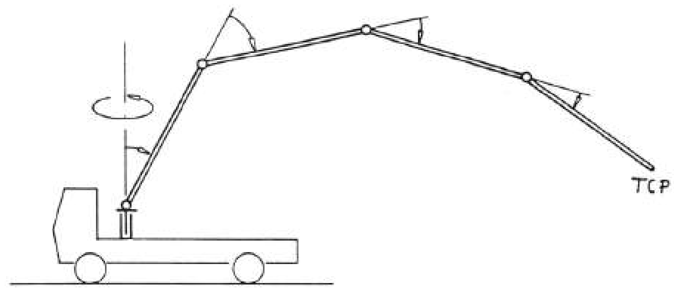


Figure 2.4.1: Illustration of a redundant manipulator

This situation can be viewed as loss of mobility coming from a bad design of the manipulator. However, this kind of architecture may sometimes be desired because, redundancy allows to increase the volume of the design space while preserving the capability of moving the effector around obstacles. Indeed the extra degrees of freedom generally allow to circumvent the obstacles in the environment. The Figure 2.4.1 illustrates the case of a redundant manipulator, which allow to avoid obstacles in the workspace.

The following joint combinations in simple open-tree structures give rise to redundant structures:

- More than 6 joints;
- More than 3 revolute joints with concurrent axes;
- More than 3 revolute joints with parallel axes;
- More than 3 prismatic joints;
- Two prismatic joints with parallel axes;
- Two revolute joints with identical joint axes.

When the robots has several effectors (multiple open-tree structures) the redundancy is evaluated in comparing the number of degrees of freedom acting on each tool and the number of degrees of freedom of the effector under consideration.

Let us consider first the example of Figure 2.4.2 which shows two distinct robot architectures, both of simple open-tree type, made of two links connected by two prismatic joints. According to the definition both have a mobility index $M = 2$. In case (a), the effector remains in a plane and keeps a constant orientation, hence $n_R = 2$. In case (b), however, both kinematic pairs give freedom in the same direction, so that the effector is constrained to move along a constant line. We have in this case $n_R = 1$: the structure represented in case (b) has thus a redundant kinematic architecture.

2.4.7 Singularity

Let us consider next the example of figure 2.4.3 which shows the same manipulator made of three links, with two prismatic and one revolute joint. The mobility index is now $M = 3$.

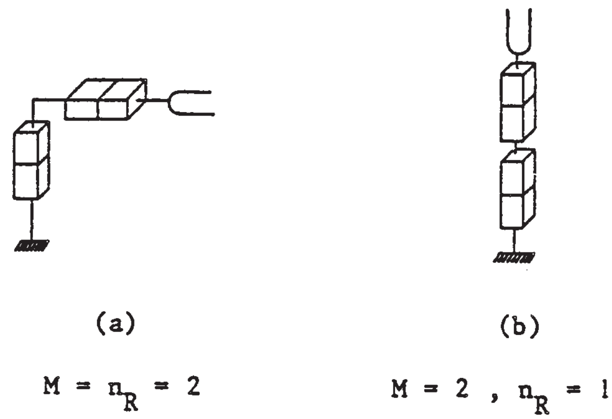


Figure 2.4.2: Comparison of the mobility index and the number of DOF for two robot manipulators with two prismatic joints

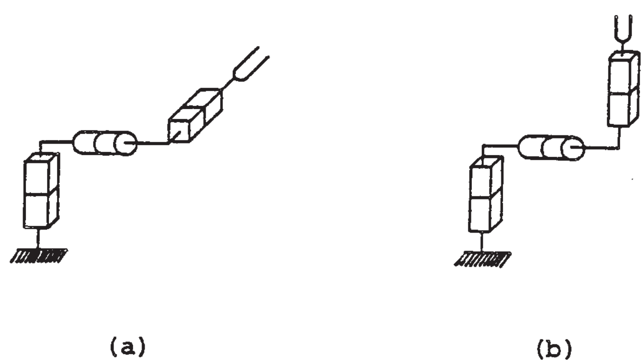


Figure 2.4.3: Comparison of the mobility index and the number of DOF for two robot manipulators with two prismatic joints

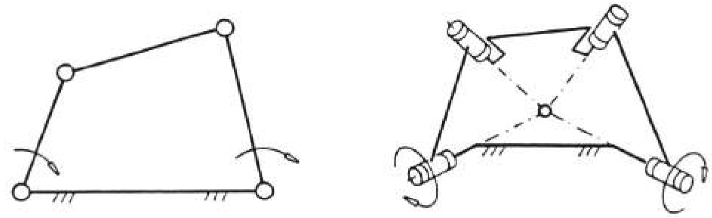


Figure 2.4.4: Planar and spherical four-bar mechanisms

The effector is constrained to move in a plane with arbitrary orientation, hence $n_R = 3$. However, in the particular configuration shown in (b), both prismatic joints give mobility in the same direction: therefore, $n_R = 2$. Such a configuration which gives rise to a local redundancy is called *singular*.

Generally speaking, for any robots, redundant or not, it is possible to discover some configurations, called *singular configurations*, in which the number of freedoms of the effector is inferior to the dimension, in which it generally operates.

Singular configurations happens when:

- Two axes of prismatic joints become parallel;
- Two axes of revolute joints become identical.

Remarks

It will be seen later some analytical methods to determine the number of degrees of freedoms of the effector and singular configurations.

A robot that is not redundant in general can be redundant when considering a given task.

2.4.8 Examples

Planar and spherical four-bar mechanisms

Let's consider the planar and spherical mechanisms of figure 2.4.4. On identifies:
Number of bodies: $n_B = 3$
Number of joints: $n_J = 4$ with a number of degrees of freedom of each joints $f_j = 1$
Number of closed loops $n_L = 1$
Using formula (2.4.3) and (2.4.4), one finds:
Number of degrees of freedom $M = 1$ (Mobility 1 mechanisms)

7-revolute-joint mechanism

A general spatial mechanism is presented in figure 2.4.5. Its characteristics are the following:
Number of bodies: $n_B = 6$
Number of joints: $n_J = 7$ with a number of degrees of freedom of each joints $f_j = 1$
Number of closed loops $n_L = 1$
Introducing these data in formula (2.4.2) gives
Number of degrees of freedom $M = 1$ (Mobility 1 mechanisms)

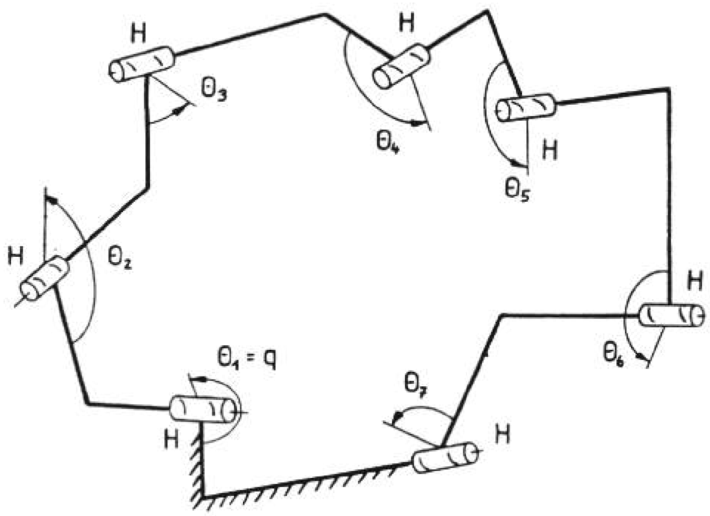


Figure 2.4.5: 7-revolute-joint mechanism

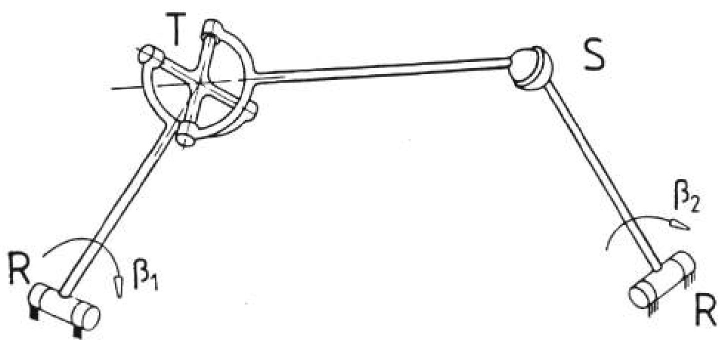


Figure 2.4.6: Spatial four-bar mechanism

Spatial four-bar mechanism (special 7-revolute-joint mechanism)

Let's consider the general spatial mechanism of figure 2.4.6, which is a particular case of the general spatial mechanism of figure 2.4.5. Indeed, universal joint can be replaced by 2 revolute joints, while the spherical joint is equivalent to 3 concurrent revolute joints. According to the previous analysis, this mechanism is a 1 degree of freedom mechanism. This kind of device is used to transmit a rotation motion β_1 towards a output rotation β_2.

If the coupler is connected to the two links by two spherical joints, there would be one additional degrees of freedom, the rotation of the coupler about its longitudinal axis. This "isolated" degree of freedom does not influence the transmission behaviour, i.e. the relation $\beta_2 = f(\beta_1)$.

SAR antenna

The SAR antenna mechanism presented in figure 2.4.7 is a planar mechanism. One particular characteristic of the mechanism is the presence of multiple joints. Each multiple joint can be

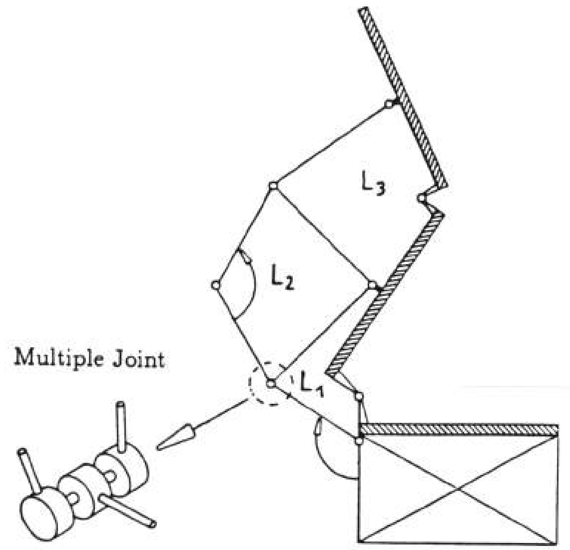


Figure 2.4.7: SAR antenna

broken into several individual revolute joints that connect a reference body and the other ones.

Number of bodies: $n_B = 8$
Number of joints: $n_J = 11$ with a number of degrees of freedom of the joints $f_j = 1$
Number of closed loops $n_L = 3$
Number of degrees of freedom $M = 2$

Five-point wheel suspension

To determine the total number of degrees of freedom, it is necessary to take into account the isolated rotation of the rods around their longitudinal axis. Therefore the joints at the car body are modeled as universal joints.

Number of bodies: $n_B = 7$
Number of joints: $n_J = 11$
A number of degrees of freedom of the joints $f_j = 1$ for revolute joints (R)
$f_j = 2$ for universal joints (T)
$f_j = 3$ for spherical joints (S)
Number of closed loops $n_L = 4$
Number of degrees of freedom $M = 2$

The two dof of the suspension are the spring deformation and the steering gear angle.

2.4.9 Exercices

Singular and redundant manipulators

For the four robots sketched in figure 2.4.9, identify the redundant manipulators and the singular configurations.

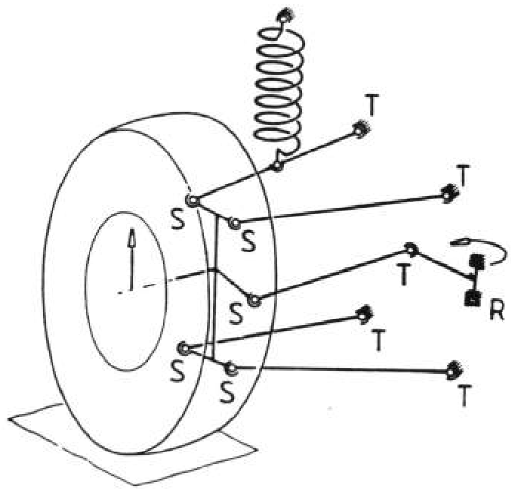


Figure 2.4.8: Five-point wheel suspension

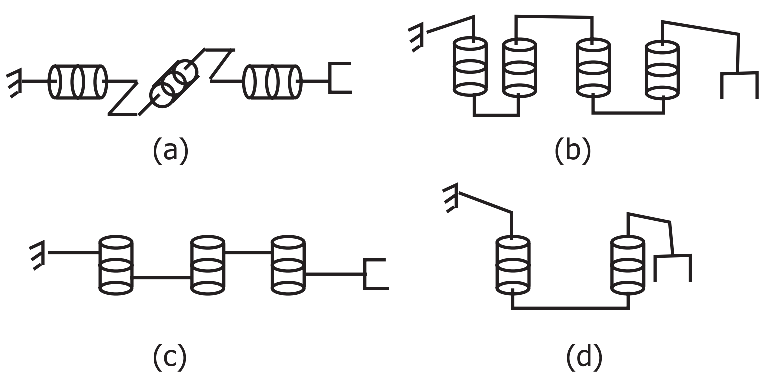


Figure 2.4.9: Illustration of redundant and singular manipulators

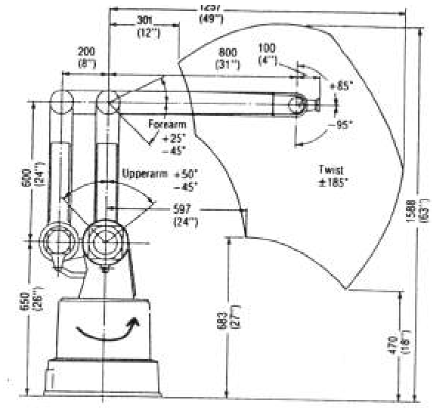


Figure 2.4.10: General Electric P5 robot

Number of dof of mechanisms and manipulators

For the robots given in figures 2.4.10, 2.4.11, and 2.4.12, find the number of degrees of freedom. Do the same for the wheel suspension given in 2.4.13.

2.5 Number of dof of the task

The number of dof n_T of a given task is equal to the number of independent parameters necessary to describe the position and orientation of the effector for all the configurations that it has to reach during the task.

Generally speaking, to give an arbitrary location to the effector (position and orientation) requires 6 dof. However, because of the geometry of the tools or of the pieces on which the work is made, many tasks may be realized with a lower number of degrees of freedom, i.e. many tasks are characterized by a number of DOF $n_T < 6$. For example:

- Any manipulation task involving a cylindrical piece is such that $n_T \leq 5$, since the task is not influenced by a rotation about the revolution axis (see Fig2.5.1, a/) ;
- Any task of vertical insertion (such as insertion of components in electronic assembly) implies at most 4 parameters: positioning and orientation in a plane, vertical positioning (see Fig2.5.1, b/). Hence $n_T \leq 4$;
- Positioning a spherical object requires $n_T = 3$ degrees of freedom, because the task is not influenced by any rotation (see Fig2.5.1, c/).

In other words, a necessary but non sufficient condition to insure the compatibility of the manipulator and the specified task to perform is that:

$$n_R = n_T \qquad (2.5.1)$$

Under these conditions, there is an adequacy of the manipulator for a specified task, i.e. the possibility of finding a configuration of the robot manipulator which makes possible to reach each specified location of the effector depends upon it.

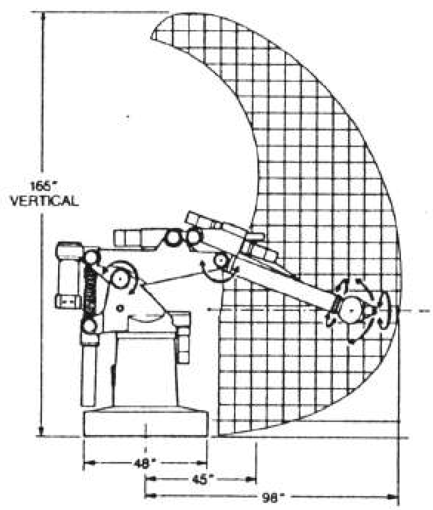


Figure 2.4.11: ML series CNC robot

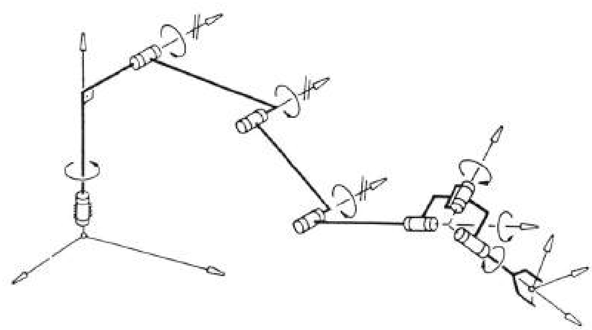


Figure 2.4.12: Articulated structure scheme of a PA10 Mitsubishi robot

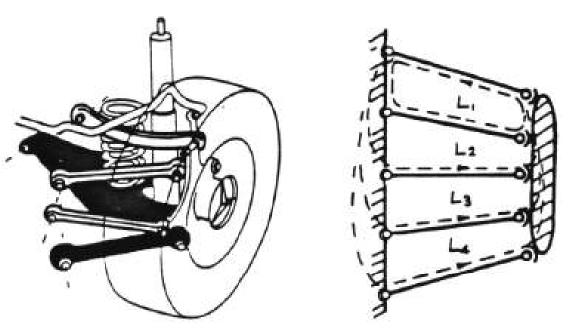
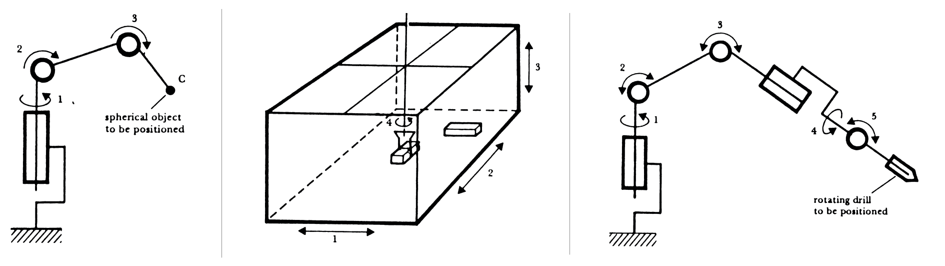


Figure 2.4.13: Wheel suspension



a/ 3 dof positioning task **b/ 4 dof insertion task** **c/ 5 dof drilling task**

Figure 2.5.1: Number of dof of different tasks

Table 2.1: Number of morphologies as a function of the number of dof of the simple open-tree manipulator

Number of dof of the robot	Numbers of possible structures
2	8
3	36
4	168
5	776
6	3508

2.6 Robot morphology

In the later chapter, our study will focus on the open-tree structure robots. Therefore, one is considering here mainly the different open-tree morphologies.

2.6.1 Number of possible morphologies

In order to consider the different architectures, one identifies two parameters:

- The type of joint (revolute -R- or prismatic -P- joint);
- The angle between two successive joints: 0 or 90°. Indeed, except in some very particular cases, the successive joints are either parallel or perpendicular.

The number of different configurations (see 2.1) comes when combining those parameters.

2.6.2 General structure of a manipulator

The general architecture of a robot may be divided into three parts (see Figure 2.6.1): the *base* or *carriage* when it is mobile, *the arm*, and *the wrist*. For fixed robots, *the arm of the robot* is usually the 3 first degrees of freedom (either of revolute or prismatic type), while the other remaining degrees of freedom, which are characterized by lower sizes and a lower mass are the *the wrist*.

Based on Table 2.1, one can enumerate 36 morphologies for the arm. Among these 36 basic configurations, only 12 are mathematically different and non redundant (one does not consider the configurations of the arms that gives rise only to linear or planar motions, for instance 3 parallel prismatic joints or 3 parallel revolute joints.

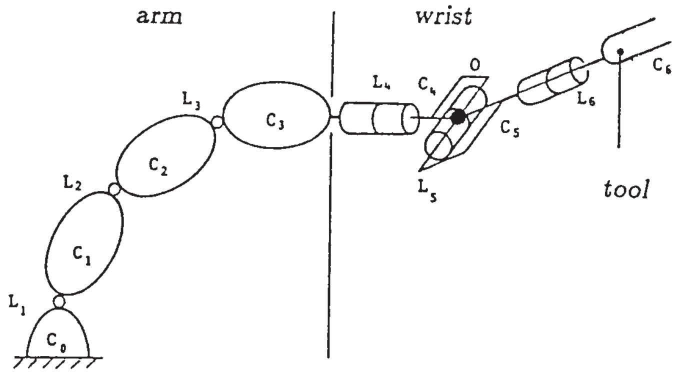


Figure 2.6.1: Uncoupling between effector position and orientation

The mechanical role of the *arm* is to position the effector in space. The *wrist* is in a charge of giving the prescribed orientation to the tool.

2.6.3 Possible arm architectures

The most usual architectures of the arm (with a simple open-tree mechanical structure) derive from the various systems used to define space coordinates. The four most common architectures are represented by figure 2.6.2. They correspond to the kinematic pair sequences PPP, RPP, RRP and RRR. Some of them have been used historically, others are current industrial solutions.
More recently, a fifth possible architecture has been introduced (figure 2.6.3): RPR, PRR or RRP, which is based on the principle of translating along the prismatic hinge a articulated coplanar mechanism. The resultant structure is called SCARA robot (Selective Compliance Assembly Robot Arm), which is very efficient for planar assembly operations such as insertion of mechanical and electronic components. This type of robot it is now gaining an increasing part of the market.

According to a study [7] realized by the Association Française de de Robotique Industrielle (AFRI) and the Journal RobAuto in 1997, there is a large majority of RRR type robot (44%), then follow the cartesian manipulators (20, 5%), cylindrical arms (7%), and SCARA robots (7%).

2.6.4 Kinematic decoupling between effector orientation and position

Most of the robots have a wrist made of three revolute joints with intersecting axes and orthogonal two by two. Even if there are other types of wrist architecture, the wrist of spherical type is very important. Indeed this architecture leads to the following fundamental properties:

- The point O, called the *center*, plays the role of a *virtual spherical joint*;
- The position position of this virtual spherical joint depends on nothing else than the displacements at hinges L_1, L_2 and L_3;
- The displacements at joints L_4, L_5 and L_6 determine the orientation of the effector about point O.

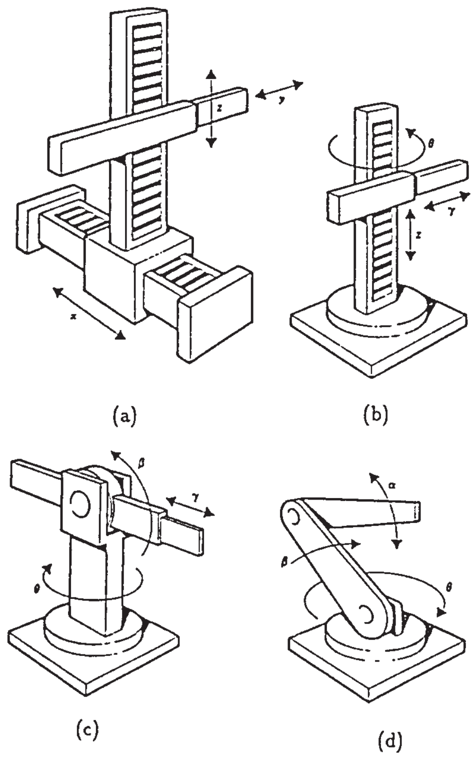


Figure 2.6.2: Most common robot architectures (a) cartesian coordinates: PPP architecture (b) cylindrical coordinates: RPP architecture (c) polar coordinates: RRP architecture (d) universal coordinates: RRR architecture

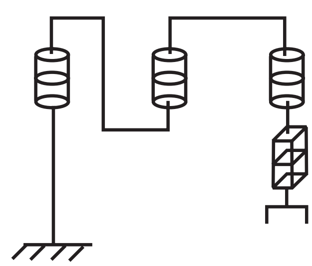


Figure 2.6.3: PRR or SCARA robot architecture

It is easy to perceive the practical consequences of such a decoupling: It allows to break the problem of determining the six configuration parameters into two independent problems, each one with only three parameters.

2.7 Workspace of a robot manipulator

2.7.1 Definition

Intuitively the *position workspace* of a robot manipulator is the physical space generated by a specified point on the tool when the system is taking all possible geometric configurations. Let us note the relative inaccuracy of this definition, for the following reasons:

- The choice of a point on the tool is arbitrary. Some manufacturers take as a reference the center O of the wrist if it exists, and obtain in this manner the position workspace with least volume. Others take as a reference either a point located at the end of the tool, or a point located at its fixture;
- The possible orientations of the tool at a given point of the workspace are not taken into account in this definition. Some authors complement it by introducing the concept of *dexterity*. The *dexterous workspace* of a manipulator is then defined as the subspace of the position workspace in which the wrist can generate an arbitrary orientation of the tool.

More precisely, one distinguishes [3] the following different workspaces :

- *Workspace of a robot* : The set of all positions and orientations that can be reached by a given frame attached at the effector of the robot while the joint degrees of freedoms are allowed to take all their admissible values.
- *Reachable workspace*: The set of all positions that a given point of the effector of the robot can reach for at least one orientation while the joint degrees of freedoms are allowed to take all their admissible values.
- *Dexterous workspace or primary workspace*: The set of all positions of the workspace that can be reached with all possible orientations. Secondary workspace: The set of all positions of the workspace that can be reached with some specified orientations. Workspace with a given orientation

Of if the workspace is the set of points in $\mathbb{R}^3 \times SO(3)$, which is impossible to draw. It is fairly common to represent the workspace with two orthogonal sections chosen according to the type of manipulator considered. This geometric representation is generally preferable to a perspective view.

2.7.2 Comparison of the workspaces with different arm configurations

In order to compare the different architectures described in section 2.7, it is necessary to adopt the following assumptions:

- The 'length' of each member is equal to L;
- The rotation allowed to each revolute joint is 360 deg;

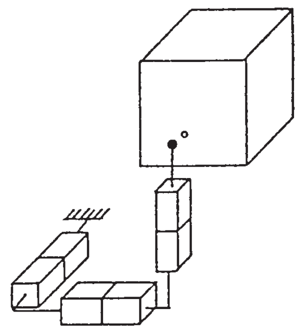


Figure 2.7.1: Structure PPP: cubic workspace with volume $V = L^3$

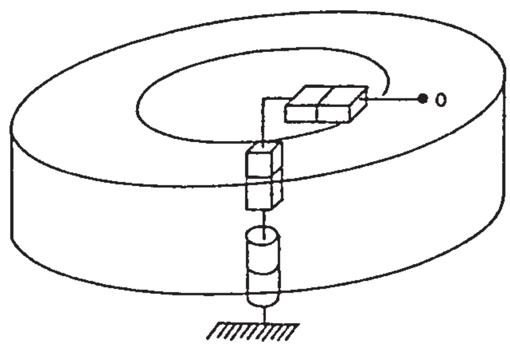


Figure 2.7.2: Structure RPP: toric workspace with square section internal radius L, external radius $2L$, volume $V = 3\pi L^3 \simeq 9L^3$

- The translation allowed to each prismatic joint is equal to L.

Under these conditions, the position workspace corresponding to each one of the five basic architectures described in section 2.7 is represented hereafter (exceptionally, in perspective form). These volumes correspond to a manipulator having an arbitrary wrist with its center located at O.

This comparison demonstrates the higher mobility of the architectures RRP and RRR, since they possess a workspace volume about 30 times larger than the PPP architecture. The RPP and RPR, on the other hand, provide a volume of intermediate magnitude.

The present analysis of workspace does not provide any information on the dexterity properties of these structures such as the ability to go around obstacles or to penetrate a cavity.

2.7.3 Workspace optimisation

An effective evaluation of the workspace has to take into account the displacement range allowed to revolute and prismatic joints as well as the length ratios of the members. The architectures

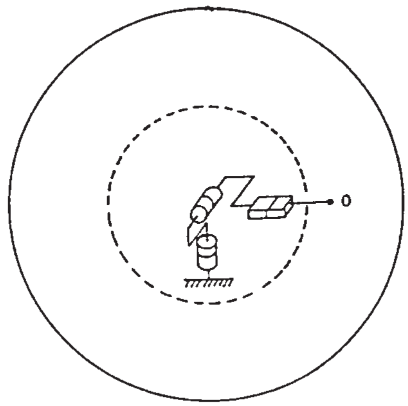


Figure 2.7.3: Structure RRP: hollow spherical workspace internal radius L, external radius $2L$, volume $V = 28/3\pi L^3 \simeq 29L^3$

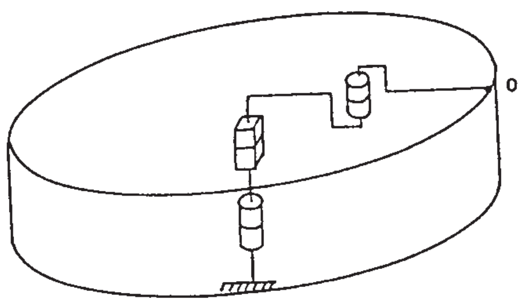


Figure 2.7.4: Structure RPR: cylindrical workspace radius $2L$, height L, volume $V = 4\pi L^3 \simeq 13L^3$

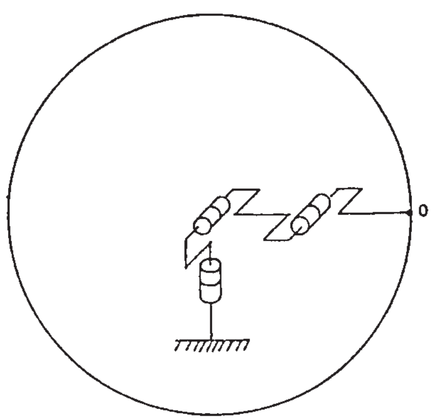


Figure 2.7.5: Structure RRR: spherical workspace radius $2L$, volume $V = 32/3\pi L^3 \simeq 34L^3$

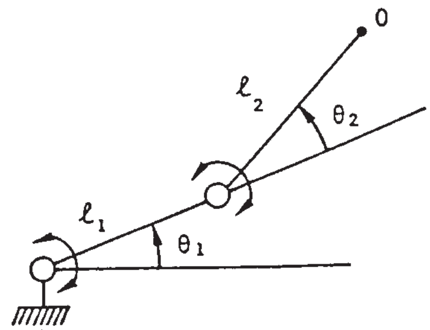


Figure 2.7.6: Planar manipulator with 2 DOF

of types PPP, RPP and RRP remain easy to analyze and it is relatively straightforward to maximize the volume of their workspace for a prescribed elongation.

The problem becomes more complex for PRR and RRR structures and the solution is no longer obvious.

Figures 2.7.7 show different workspace configurations for a planar manipulator with 2 DOF.

In both latter cases, the problem can be reduced in a first step to a planar one: the 3-dimensional workspace is then obtained by sweeping through either translation or rotation the geometric figure 2.7.7, obtained in the plane.

Let us thus consider the planar mechanism of figure 2.7.6 such that

- The maximum elongation $L = \ell_1 + \ell_2$ is specified;
- The move limits are given on both DOF θ_1 and θ_2:

$$\theta_{1m} \leq \theta_1 \leq \theta_{1M} \qquad \text{and} \qquad \theta_{2m} \leq \theta_2 \leq \theta_{2M} \tag{2.7.1}$$

  According to the variation domain of θ_2, the workspace can take one of the fundamental shapes represented on figure 2.7.7

Evaluation of the swept area

The area of the domain swept by point O is evaluated by

$$A = \int dx\ dy = \int |det\ J|\ d\theta_1\ \theta_2 \tag{2.7.2}$$

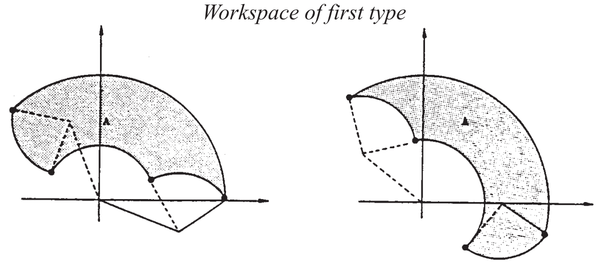
where J is the *Jacobian matrix* of the coordinate transformation

$$\begin{aligned} x &= \ell_1 \cos\theta_1 + \ell_2 \cos(\theta_1 + \theta_2) \\ y &= \ell_1 \sin\theta_1 + \ell_2 \sin(\theta_1 + \theta_2) \end{aligned} \tag{2.7.3}$$

or

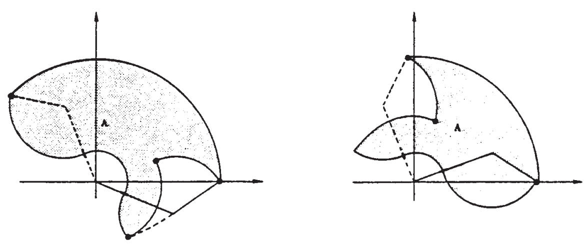
$$J = \begin{bmatrix} -\ell_1 \sin\theta_1 - \ell_2 \sin(\theta_1+\theta_2) & -\ell_2 \sin(\theta_1+\theta_2) \\ \ell_1 \cos\theta_1 + \ell_2 \cos(\theta_1+\theta_2) & \ell_2 \cos(\theta_1+\theta_2) \end{bmatrix} \tag{2.7.4}$$

Workspace of first type



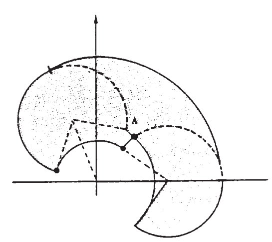
$0 \leq \theta_{2m} \leq \theta_2 \leq \theta_{2M} \leq \pi$ $\quad -\pi \leq \theta_{2m} \leq \theta_2 \leq \theta_{2M} \leq 0$

Workspace of second type



$0 \leq \theta_{2m} \leq \theta_2 \leq \theta_{2M}, \quad \theta_{2M} \geq \pi$ $\quad \theta_{2m} \leq -\pi, \quad \theta_{2m} \leq \theta_2 \leq \theta_{2M} \leq 0$

Workspace of third type



$\theta_{2m} \leq \theta_2 \leq \theta_{2M}, \quad -\pi < \theta_{2m} < 0, \quad 0 < \theta_{2M} < \pi$

Figure 2.7.7: Possible shapes of the workspace for a planar manipulator with 2 DOF

from which one deduces that

$$det\ J \ = \ \ell_1 \ell_2 \ \sin\theta_2 \tag{2.7.5}$$

Therefore the area swept by the point O is given by

$$A \ = \ \ell_1 \ell_2 \ (\theta_{1M} - \theta_{1m}) \int_{\theta_{2m}}^{\theta_{2M}} |\sin\theta_2| d\theta_2 \tag{2.7.6}$$

or, formally

$$A \ = \ \ell_1 \ell_2 \ (\theta_{1M} - \theta_{1m}) f(\theta_{2m}, \theta_{2M}) \tag{2.7.7}$$

Optimization of the swept area

Optimization of the length ratio: the maximal elongation $L = \ell_1 + \ell_2$ being specified, the ratio $\lambda = \frac{\ell_1}{\ell_2}$ will maximize the volume

$$A \ = \ \frac{\lambda}{(1+\lambda)^2} L^2 \ (\theta_{1M} - \theta_{1m}) f(\theta_{2m}, \theta_{2M}) \tag{2.7.8}$$

if

$$\frac{dA}{d\lambda} = 0 \qquad \text{and} \qquad \frac{d^2 A}{d\lambda^2} < 0 \tag{2.7.9}$$

It is observed effectively that

$$\frac{dA}{d\lambda} \ = \ \frac{1-\lambda}{(1+\lambda)^3} L^2 \ (\theta_{1M} - \theta_{1m}) f(\theta_{2m}, \theta_{2M}) \ = \ 0 \qquad \text{for} \quad \lambda = 1 \tag{2.7.10}$$

and

$$\frac{d^2 A}{d\lambda^2} \ = \ \frac{2(\lambda-2)}{(1+\lambda)^4} L^2 \ (\theta_{1M} - \theta_{1m}) f(\theta_{2m}, \theta_{2M}) \ \leq \ 0 \qquad \text{for} \quad \lambda = 1 \tag{2.7.11}$$

We may thus conclude that *the workspace area is the largest when both links have equal lengths.*

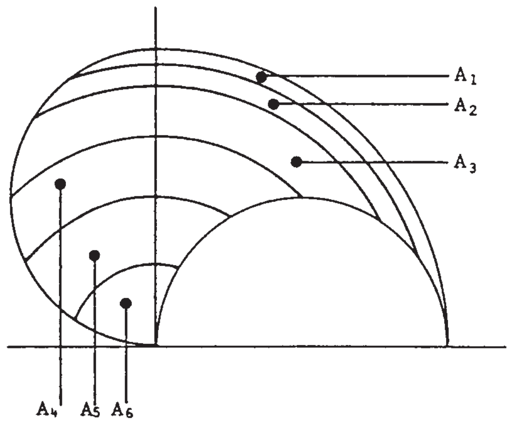
Optimisation of θ_{2m} and θ_{2M}

The integrand of equation (2.7.6)

$$\ell_1 \ell_2 \ |\sin\theta_2| \ (\theta_{1M} - \theta_{1m}) \tag{2.7.12}$$

represents the variation rate of the workspace with respect to θ_2. It vanishes if $\theta_2 = 0$ and reach its maximum when $\theta_2 = \pm\pi/2$. It means that a variation of θ_2 centered about $\pm\pi/2$ allows to sweep a maximal area. This conclusion is confirmed by the table of figure 2.8.11 for the particular case where $\theta_{1M} - \theta_{1m} = 90\,\text{deg}$ and $\ell_1 = \ell_2 = 1$.

This simplified analysis leads to the conclusion that a kinematic architecture of types RRR or PRR *provides a good design if the links have equal lengths and that the variation range for* θ_2 *is centered around* $\pm 90\,\text{deg}$. It is interesting to observe that the human arm matches these conditions.



$0 \leqslant \theta_2 \leqslant 30°$	$A_1 = 0{,}210$
$30° \leqslant \theta_2 \leqslant 60°$	$A_2 = 0{,}575$
$60° \leqslant \theta_2 \leqslant 90°$	$A_3 = 0{,}785$
$90° \leqslant \theta_2 \leqslant 120°$	$A_4 = 0{,}785$
$120° \leqslant \theta_2 \leqslant 150°$	$A_5 = 0{,}575$
$150° \leqslant \theta_2 \leqslant 180°$	$A_6 = 0{,}210$

Figure 2.7.8: Variation of workspace area with angular range θ_2

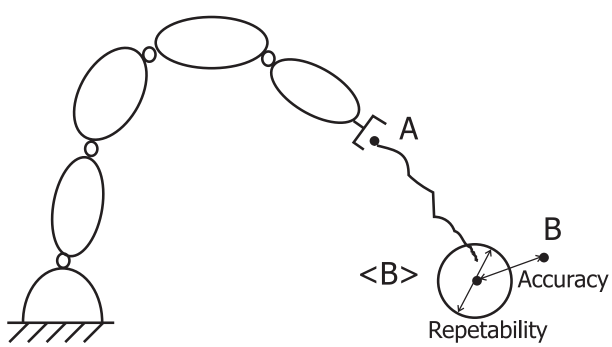


Figure 2.8.1: Repeatability and accuracy experiment of a robot

2.8 Accuracy, repeatability and resolution

In order to explain these three complementary concepts, let us consider the following experiment (see fig 2.8.1). An arbitrary robot at rest in a configuration A and programmed in such a way that its effector is moved and stops in a specified configuration B. The experiment is repeated a large number of times and statistical processing can be realized of the results.

Let us repeat this operation a large number of times: there exists an average location B_m corresponding to the mean value of the locations reached during the successive displacements.

2.8.1 Static accuracy

The deviation between the programmed situation B and the average location B_m provides a measure of the *static accuracy* of the manipulator. It characterizes the ability of the manipulator to locate the tool in accordance to the programmed situation: it can be a function of the geometric configurations A and B themselves and of the trajectory selected to perform the motion. It may also depend on parameters such as

- the static accuracy of the feedback systems controlling the actuators;
- the structural flexibility of members and joints;
- the load carried by the tool;
- the friction in backlash in joints;
- the resolution of the position and velocity sensors;
- etc.

When account is taken of all the possible sources of error, it is found that the static accuracy of most industrial robots, when measured at tool level, does not exceed the following values:

- Positioning accuracy: a few millimeters;
- Orientation accuracy: a few tenths of degrees.

2.8.2 Repeatability

The deviation between the average location B_m and the successive locations reached effectively gives a measure of the *repeatability* of the manipulator. Since from its very definition the repeatability is not affected by any systematic cause of inaccuracy, it is generally much better than the static accuracy.

2.8.3 Resolution

The *resolution* of a robot manipulator is the smallest distance between the initial configuration A and the programmed configuration B which generates an effective change in the location of the tool. It may depend on

- the resolution of the position and velocity sensors;

- the word length used in arithmetic computations;
- the gear ratio of the transmission chains.

It is important to have clearly in mind the fundamental difference between these three concepts. Let us note that most manufacturers, in the technical notices, provide only a measure of the repeatability.

2.8.4 Normalisation

The problem of repeatability and static accuracy of robots is a primary importance for industrial robot applications. This led to a normalization work from ISO and it has been written down in the normalisation document ISO 9283, which describes the tests conditions under which these characteristics can be measured. This document specifies also the other performances of robots in order to insure their comparison for the choice. Additionally, normalization allows also the users to assess the interchangeability of robots in production conditions.

2.9 Robot actuators

We have up to now described the arm and wrist architectures which form together the mechanical structure of the robot. Let us now have a quick look to the way this mechanical structure is put into motion. For activating each joint, *motorization* implies

- To *provide* a primary *energy*, most often in pneumatic, hydraulic or electrical form;
- To *modulate* the energy brought to the system;
- To *convert* the primary energy into mechanical energy;
- To *transmit* the mechanical energy to the joint;
- To control and measure the motion variables (position, velocity, acceleration, force, etc.)

Figure 2.9.1 summarizes these different functions. Production and modulation of energy are functions which fall completely out of our scope. We will briefly describe the functions of energy conversion by the actuators, transmission of the mechanical energy by the transmission chains and position and velocity measurement by the sensors.

The selection of appropriate transmission chains greatly differs according to the choice of either *distributed motorization* or *centralized motorization*: it is thus important to have some ideas about the advantages and drawbacks of both solutions.

2.9.1 Distributed motorization

The use of actuators which drive directly the joint axis on which they act looks like the simplest solution that can be adopted for obtaining an efficient design. Presently this solution is however not the most common one, for a certain number of technological reasons:

1. Most of the actuators provide a high velocity output with low torque, while what is needed is to provide a high torque at low velocity. A tramsmission system is thus needed to *adapt the mechanical impedances* of the actuator and its load: even when distributed motricity is adopted, in many cases the use of a speed reductor cannot be avoided.

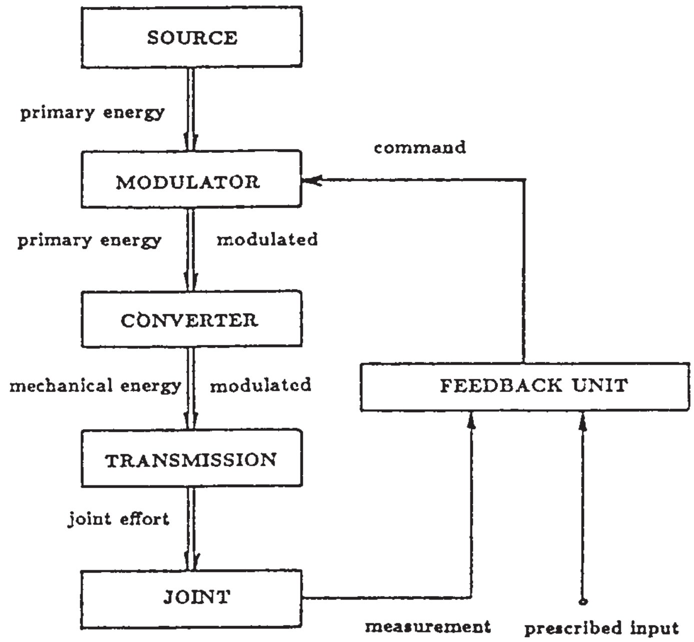


Figure 2.9.1: Motorization of robot joints

2. The actuator equipped with the appropriate speed reductor has a non negligeable volume and locating it at the joint might reduce the mobility of the manipulator in a significant manner;

3. Likewise, each actuator equipped with its speed reductor possesses a mass and a rotary inertia which have to be added to that of the passive structure; this additional load has to be into taken account in the sizing of the system and may significantly affect its dynamic behavior.

2.9.2 Centralized motorization

In opposition to the previous solution, it is possible to locate in a systematic manner all the actuators at the basis of the manipulator and transmit the motion to the joints through appropriate transmission chains. Obviously, the transmission components have also their own weight and generate a significant weight penalty for the mechanical structure. Moreover, centralized motorization raises some specific problems such as

1. Transmitting the motion to several joints through a mechanical structure with varying geometric configuration can only be achieved at the cost of a complex internal kinematics;

2. The *friction* between mechanical components generates significant losses of energy and reduces the accuracy of the system (backlash, hysteresis);

3. The *low stiffness* of some mechanical components may be the origin of *vibration and jerk* in the structure.

2.9.3 Mixed motorization

In practice, it is observed that there exists a whole range of intermediate solutions which attempt to make the best compromise between both extreme solutions.

Let us also note that a significant effort is currently made to produce efficient *direct drive actuators* of electric type, in which case distributed motorization will become the most relevant choice.

2.10 Mechanical characteristics of actuators

A *robot actuator* is a device able to generate a force or a torque at variable speed, and thus capable to modify at any instant the geometric configuration of the mechanical structure.

If we limit ourselves to the types of actuators that can be used in robotics, it is possible to classify them according to the following criteria:

- *The type of motion generated*: with the present technology, it is possible to use *linear actuators* which develop a force and generate a translation motion in the same direction, and *rotary actuators* which develop a torque and generate a rotation motion about the torque axis;
- *The nature of the primary source of energy:* one disposes of
  - pneumatic actuators, whose source of energy is compressed air;
  - hydraulic actuators, which develop their power from oil under pressure;
  - electrical actuators.

The *power to mass ratio* and the *acceleration power* are two important criteria to make an objective comparison of the different actuator types.

2.10.1 Power to mass ratio

The power to mass ratio of an actuator is defined as the ratio of the mechanical power developed by an actuator to its mass. Typical values that can be reached are

- For an electric actuator, 0.5 to 0.6 kW/kg;
- For a hydraulic actuator, 2.5 to 5 kW/kg.

The power to mass ratio of hydraulic actuators is thus from 5 to 10 times better, and furthermore hydraulic actuators can give an efficiency higher than 50% since they generally allow to drive the joints directly.

Hydraulic energy provides thus very compact actuators, perfectly well adapted to the needs of a distributed motorization. Hydraulic actuators fit well within a general architecture of the manipulator of simple open-tree type and allow for a certain modularity in the design (which means that a given arm can be made from an assembly of standardized elements providing each the mechanical structure and the actuation of one DOF). One has however to mention that there is a need for small power hydraulic actuators as required in light duty robotics such as electronic assembly.

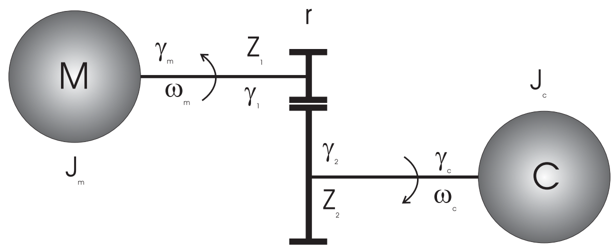


Figure 2.10.1: Model of a 1 DOF mechanical transmission

The pneumatic actuator would benefit very much of the same properties as hydraulic ones if on one hand, the alimentation pressure was not limited to about 15 bars approximately, mainly due do constraints arising from workshop environment, and if on the other hand the air was not characterized by a high compressibility which makes pneumatic feedback technology difficult. Therefore, pneumatic actuators are only used for small robots of “pick-and-place” type and for robots used in deflagrating environment.

Due to their relatively power to mass ratio, electric actuators are better adapted to a centralized motorization, or at least to a motorization where the actuator locations are chosen in the kinematic chain at a certain distance from the driven joints. A compromise has to be found between lightening of the structure through location of actuators in its less mobile parts (such as the waist and the upper arm) and the mechanical complexity arising from the passage of transmission chains through joints located up-stream in the mechanism. It is generally observed that the use of electric actuators leads to designs with complex kinematic architecture (and thus, high structural stiffness) and high level of integration (hence, with no or very little modularity).

2.10.2 Maximum acceleration and mechanical impedance adaptation

Reduction devices are present in many robotic and mechatronic systems. Indeed motor characteristics are not always adapted to the desired motions. At first the velocity of the load have to be generally much smaller than the rotation speed of electric motors to achieve the desired precision values. Secondly most of the actuators are rotary motors, while the load is motion is often linear. So a mechanism to transform the rotation motion into a linear motion is necessary. Finally as most of electric motors have high speed their torque is low, so they provide low torque, which is not good. Therefore a reduction the velocity is also mandatory.

Modeling of a 1 DOF reduction system

The proper choice of a system transmission ratio is necessary to produce maximum system acceleration. In order to illustrate the concept of adaptation of mechanical impedances, one considers the model of 1 DOF mechanical system consisting of an inertial load driven by a motor through a mechanical transmission (Figure 2.10.1). One defines:

- ω_m, the velocity of the motor shaft,

- ω_c, the velocity of the load,

- J_m, the internal inertia of the actuator,

- J_c, the inertia of the driven load,

- γ_m, the torque delivered by the actuator,
- γ_c, the torque that is absorbed by the load.

The reduction ratio r of the transmission device is defined as the ratio of the output velocity to the input velocity:

$$r = \frac{\omega_m}{\omega_c} \tag{2.10.1}$$

For gear pairs, this ratio can be also expressed in terms of the number of teeth of the two wheels (see Figure 2.10.1)

$$r = \frac{Z_2}{Z_1} \tag{2.10.2}$$

One can generally neglect the internal frictions of the motor but not the efficiency of the transmission device. This efficiency ration is defined in terms of the input and output power of the reduction system.

$$\eta = \frac{\gamma_2 \ \omega_c}{\gamma_1 \ \omega_m} \tag{2.10.3}$$

where one defines:

- γ_1, the resistant torque that is presented by the transmission device to the actuator,
- γ_2, the torque that comes out from the reduction device to drive the load.

The efficiency ration is generally not well known. This value is defined for steady state regimes and given standard conditions, but generally the system is used in different conditions. Nonetheless it makes sense to use this efficiency ratio *for instantaneous power* in transient regimes even under non standard conditions. To account for the approximation, it is necessary to take a quite large security margin: a security margin of 25% is often a good guess. Introducing the reduction ratio r, the input and output torques are related by:

$$\gamma_1 = \frac{\gamma_2}{r\eta} \tag{2.10.4}$$

The dynamic equilibrium of the motor shaft and of the load shaft leads to:

$$J_m \frac{d\omega_m}{dt} = \gamma_m - \gamma_1 \tag{2.10.5}$$

$$J_c \frac{d\omega_c}{dt} = \gamma_2 - \gamma_c \tag{2.10.6}$$

Eliminating the intermediate variables γ_1 and γ_2, one gets the equations of the dynamic equilibrium of the system:

$$\gamma_m = \frac{\gamma_c}{r\eta} + \frac{J_c}{r\eta}\frac{d\omega_c}{dt} + J_m \frac{d\omega_m}{dt}$$

The dynamic equilibrium can be expressed in terms of the acceleration of the motor or of the load:

$$\gamma_m = \frac{\gamma_c}{r\eta} + \left(\frac{J_c}{r^2\eta} + J_m\right) \frac{d\omega_m}{dt} \tag{2.10.7}$$

$$\gamma_m = \frac{\gamma_c}{r\eta} + \left(\frac{J_c}{r\eta} + rJ_m\right) \frac{d\omega_c}{dt} \tag{2.10.8}$$

Equations (2.10.7) or (2.10.8) are useful for many design problems. An important design problem consists in determining the torque to operate the system to move a given load with prescribed speed and acceleration.

Mechanical impedance adaptation problem

When the load torque γ_c and the maximum acceleration of the load $d\omega_c/dt$ are given, there is a value of the reduction ratio that minimizes the torque to be delivered by the motor. The value of the actuator torque in terms of r is given by the equation (2.10.8). The optimal value of the reduction ratio is the one for which the derivative of the γ_m with respect to r is equal to zero, that is:

$$J_m \frac{d\omega_c}{dt} - \frac{1}{r^2\eta}\left(J_c\frac{d\omega_c}{dt} + \gamma_c\right) = 0$$

One gets the optimal value of the reduction ratio:

$$r^\star = \sqrt{\frac{J_c\frac{d\omega_c}{dt} + \gamma_c}{\eta J_m \frac{d\omega_c}{dt}}} \quad (2.10.9)$$

In the expression of γ_m, one can distinguish two terms: The first one ($rJ_m\frac{d\omega_c}{dt}$) depends linearly on r, while the second one ($\frac{\gamma_c}{r\eta} + \frac{J_c}{r\eta}\frac{d\omega_c}{dt}$) depends on $1/r$. An interesting property of the optimum is that the two the terms become equal and one has:

$$\gamma_m = 2J_m r^\star \frac{d\omega_c}{dt} \quad (2.10.10)$$

When there is no resistant torque (i.e. $\gamma_c = 0$) and the reductor is ideal (i.e. $\eta = 1$), it comes:

$$r^\star = \sqrt{\frac{J_c}{J_m}} \quad (2.10.11)$$

This can also be written:

$$J_m = \frac{J_c}{r^{\star\,2}} \quad (2.10.12)$$

This means that the design torque is minimum when the inertia of the load reported at the actuator is equal to the inertia of the motor.

Optimal reduction ratio for maximal acceleration of the load

Another interesting problem is to find the reduction ratio that maximises the acceleration of the load for given values of the motor and resistant torques γ_m and γ_c. Equation (2.10.8) gives:

$$\frac{d\omega_c}{dt} = \frac{\gamma_m - \frac{\gamma_c}{r\eta}}{\frac{J_c}{r\eta} + rJ_m} \quad (2.10.13)$$

The optimality conditions of this problem is obtained by expressing that the first derivatives of the acceleration of the load is zero:

$$\frac{d}{dr}\,\frac{d\omega_c}{dt} = \frac{\frac{\gamma_c}{r^2\eta}(\frac{J_c}{r\eta} + rJ_m) - (\gamma_m - \frac{\gamma_c}{r\eta})(\frac{-J_c}{r^2\eta} + J_m)}{(\frac{J_c}{r\eta} + rJ_m)^2} = 0$$

After some algebra one finds that the optimal reduction ratio $r^\star$ is the solution of the quadratic equation:

$$\frac{\gamma_m J_c}{\eta}\,\frac{1}{r^2} + 2\frac{\gamma_c J_m}{\eta}\,\frac{1}{r} - \gamma_m J_m = 0$$

It comes:

$$\frac{1}{r^\star} \;=\; \frac{\gamma_c}{\gamma_m}\frac{J_m}{J_c} \;+\; \sqrt{\frac{\gamma_c^2}{\gamma_m^2}\frac{J_m^2}{J_c^2}+\eta\frac{J_m}{J_c}} \qquad (2.10.14)$$

So the general solution of the problem of impedance adaptation given by (2.10.9) and of the maximization of the acceleration power given by (2.10.14) are different. However they are equal when the transmission system is ideal.

Let's consider the ideal problem again, that is when there is no resistant torque (i.e. $\gamma_c = 0$) and the reductor is ideal (i.e. $\eta = 1$). The acceleration of the load is:

$$\frac{d\omega_c}{dt} \;=\; \frac{\gamma_m}{\frac{J_c}{r}+rJ_m} \qquad (2.10.15)$$

The optimal reduction ratio satisfies :

$$\frac{d}{dr}\,\frac{d\omega_c}{dt} \;=\; \frac{\gamma_m\,(-\frac{J_c}{r^2}+J_m)}{(\frac{J_c}{r}+rJ_m)^2} \;=\; 0$$

This equation has the same solution as for the minimal torque problem:

$$r^\star \;=\; \sqrt{\frac{J_c}{J_m}} \qquad (2.10.16)$$

Acceleration power of actuators

For hydraulic actuators which develop the torque γ_m at low velocity, the transmission ratio can be very close to 1. They have thus a very high acceleration power which makes them specially well adapted to robots from which high accelerations are required.

The acceleration power of pneumatic actuators gives them characteristic properties close to those of hydraulic ones as far as dynamic response is concerned.

On the other hand, electric actuators provide a low torque with high rotation speed: they have thus to be used with high transmission ratios, and it is impossible to realize the optimal reduction ratio gven by equation (2.10.16) that gives the highest acceleration to the load. Thus their acceleration power remains limited.

Rotation to translation transformation

The approach developed in the previous section can easily be extended to various of transmission devices. For instance, the rotation to translation transmission system (helicoidal pair) is presented here after.

Let us define

- T, the torque or force developed by the actuator,

- J, the internal (rotary or translational) inertia the actuator,

- M, the inertia of the load driven by the actuator,

- r, the transmission ratio defined as the ratio of output to input (angular or linear) velocity,

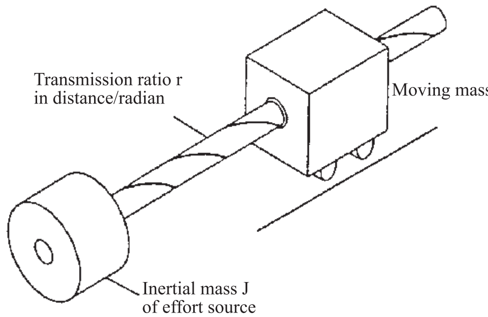


Figure 2.10.2: 1 DOF model of mechanical transmission

- a, the output linear acceleration of the system.

By the very definition of the transmission ration one has the following relationship between the displacement x of the load and the angular displacement θ of the motor

$$x = r\,\theta \quad \text{with} \quad r = \frac{p}{2\pi}$$

where p is the advance length per rotation. Thus for accelerations one has:

$$\ddot{\theta} = \frac{a}{r}$$

One has also conservation of the work through the transmission device:

$$C\,\theta = F\,a \quad \text{or} \quad C = r\,F$$

Therefore when seen from the motor side, the inertia force of the load is

$$C = r\,M\,a$$

Hence the dynamics of the system is governed by the equation

$$T = (\frac{J}{r} + Mr)\,a \tag{2.10.17}$$

and the resulting acceleration is

$$a = \frac{T\,r}{J + Mr^2} \tag{2.10.18}$$

It takes a maximum value when the transmission ratio is such that

$$\frac{da}{dr} = \frac{T}{J + Mr^2} - \frac{2r^2\,T\,M}{(J + Mr^2)^2} = 0 \tag{2.10.19}$$

or

$$r_{opt} = \sqrt{\frac{J}{M}} \tag{2.10.20}$$

Therefore, the acceleration power of the actuator is given by the formula

$$a_{max} = \frac{1}{2}\frac{T}{\sqrt{MJ}} = \frac{1}{2}\frac{T}{r_{opt}M} \tag{2.10.21}$$

Equation (2.10.21) provides the value of the maximum acceleration that an actuator can communicate to a given load M when developing a starting torque T.

2.11 Different types of actuators

The ideal characteristics that are expected from a robot actuator are:

- A low inertia in order to increase the acceleration power of the joints;
- A high mechanical stiffness in order to minimize the deflection at tool level under the action of the load;
- A low operating speed V_0: a few rad/s in rotary motion and a few tenths of cm/s in linear motion in order to obtain tool velocities from 1 to 5 m/s;
- A velocity range with continuous variation from $-V_0$ to $+V_0$;
- A high output force even at zero velocity, in order to provide an adequate holding torque;
- Low non-linearities (dry friction, backlash, etc.)
- The possibility to perform velocity and/or force control.

No actuator currently available fulfills all the conditions mentioned here above. The most suitable actuators are:

- Among the electric actuators:
  - The step motors,
  - The direct current (D.C.) motors with constant inductor flux (generated either by a permanent magnet or through a constant inductor current) and controlled by the armature current;
- Among the hydraulic actuators:
  - The linear and rotary pistons,
  - The rotary motors with axial pistons;
- Among the pneumatic actuators:
  - The linear and rotary pistons.

2.11.1 Step motors

The principle of step motors is to convert directly a electric *digital* signal into *incremental* angular positioning: the *step motor driver*, which plays the interfacing role between the control unit and the motor, receives clock pulses at a varying frequency, each pulse generating an angular displacement of fixed amplitude called the *angular step*.

The average rotation speed N_m, expressed in rpm, is equal to

$$N_m = \frac{60\ f}{n} \tag{2.11.1}$$

where

- f is the frequency of clock pulses

- n is the number of steps per revolution of the motor (usually, from 200 to 400 steps per revolution).

This relationship remains verified under limited load and acceleration/deceleration conditions above which positioning errors may occur.

Due to its very principle, step motors are velocity controlled, and their main advantage is to avoid the need of a closed-loop servo system since the revolution angle is proportional to the number of clock pulses received.

Step motors can be of three different types:

- Motors with permanent magnet armature,
- Motors with variable reluctance,
- Hybrid motors.

The last ones gather the advantages of both former categories and are thus the more performing. They develop a relatively low torque, which can however be hold at zero velocity, and their internal inertia is relatively high, so that they have a low power to mass ratio.

2.11.2 Direct current motors

In robotics, the most used type of D.C. motor is the armature-controlled one with constant inductor flux. It possesses a relatively high operating speed (from 2000 to 3000 rpm) and implies thus a transmission chain with high reduction ratio which is a source of backlash and reduces the acceleration power.

D.C. motors can be of different types: *standard motors* in which the armature is wound to a magnetic material, *bell motors* in which the armature conductors are attached to an insulated cylinder and *disk motors* in which the armature conductors are attached or wound to an insulated disk. Among these, the disk motor possesses the smallest internal inertia, but still has a relatively low acceleration power.

The models needed to control D.C. motors are well known: when they are voltage-controlled, the transfer function rotation speed versus input voltage is of second order but can be reduced to a first order one under certain approximations. When current-control is used, the transfer function rotation speed versus input current is of first order and the torque developed is then proportional to the input current (if friction effects are neglected).

D.C. motors can thus be either *velocity - or torque - controlled*, and the appropriate drivers are *servo-amplifiers* and *pulse width modulation - amplifiers*.

2.11.3 Hydraulic actuators

Pistons are very simple and effective hydraulic systems which exist in either linear or rotary form. In their linear version, the displacement range is generally of a few centimeters or dizains of centimeters, while in the rotary version the angular displacement is limited to about 270 deg.

Axial piston motors provide a continuous rotation but their technology is significantly more complicated.

Their control is achieved either through a small electric motor acting on a distributor or using a servovalve which provide the interface between the control unit and the actuator itself.

The main advantages of hydraulic actuators are, as already mentioned, their low internal inertia and high acceleration power. They are also characterized by a high mechanical stiffness.

However, they exhibit a large number of nonlinearities of different origins which make the associated feedback system difficult to stabilize.

The ordinary pistons do not raise significant problems when maintained at zero speed, but this is not the case of axial piston actuators which present important torque and velocity oscillations near zero speed.

2.12 The sensors

Our goal in this section is not to make a complete review of all the sensors used in robotics. We are only interested in the proprioceptive sensors which provide information on the current state of the mechanical structure and thus, allow us to control its motion. Controlling the motion of the manipulator implies having a system allowing us to know at any instant the configuration parameters of the system and their first order time derivatives. In this way, information is always available on position and velocity for all the members.

Two types of sensors have to be considered:

- position sensors,
- velocity sensors.

Let us note that we will not consider the most elaborate category of sensors which measure directly the location of the tool and its velocity in a 3-D space, for the reason that they still make the object of intensive research and are not yet well developed.

2.13 Integration of sensors in the mechanical structure

The role of the sensor in the feedback loop is to measure either the position or the velocity of the joint whose displacement constitutes the configuration variable.

Integration of the position and velocity sensors in the power transmission chains linking the actuators to the joints can be made in two ways.

- *solution 1: direct measurement* The first choice corresponds to the case where the sensor is attached to the joint and measures thus directly its displacement: it is a theoretical situation.

  The accuracy on the measurement of the configuration variable is then equal to the resolution of the sensor. However, all the imperfections present in the design of the joint and its transmission, if they do not alterate the measurement itself, affect both the accuracy and stability of the feedback system. Therefore, it is important to minimize the structural compliance, the friction and the backlash in the transmission. It has unevitably consequences on the cost of the design.

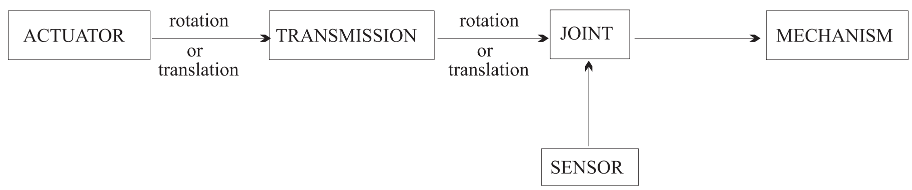


Figure 2.13.1: Direct measurement of joint position and velocity

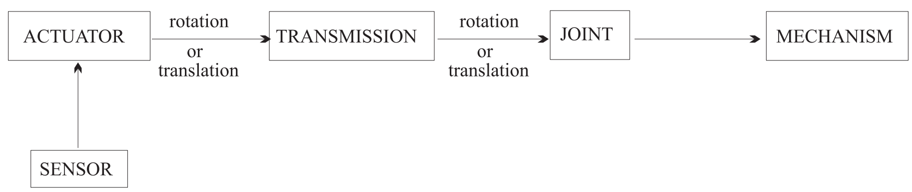


Figure 2.13.2: Indirect measurement of joint position and velocity

Note also that making the measurement directly at joint level implies the availability of sensors designed to measure low velocities.

- *solution 2: indirect measurement*

The alternative solution consists of attaching the sensor to the actuator axis, in which case the result of the measurement has to be multiplied by the transmission ratio r as defined in section 2.11 in order to provide the value of the configuration parameter.

To provide the same accuracy in the measurement, the resolution of the sensor does not need to be as good as in the former case since the measured quantity is multiplied by the transmission ratio. However, the quality of the measurement is now affected by the loss of accuracy in the transmission.

2.13.1 Position sensors

Figure 2.13.3 summarizes the different methods to perform position measurements which we will briefly review:

- digital position sensors
- incremental encoders

The principle of incremental encoders (Figure 2.13.4(a)) is to generate a pulse for any variation of the configuration parameter equal to the basic step of the encoder. Incremental encoders have thus to be associated to an algebraic counting device which sums with their correct sign the pulses generated by the encoder. The counter contains at any instant the value of the displacement made from the reference configuration in which it has been reset to zero.

- *Absolute digital encoders:*

  They provide a binary code information which corresponds to the current value of the

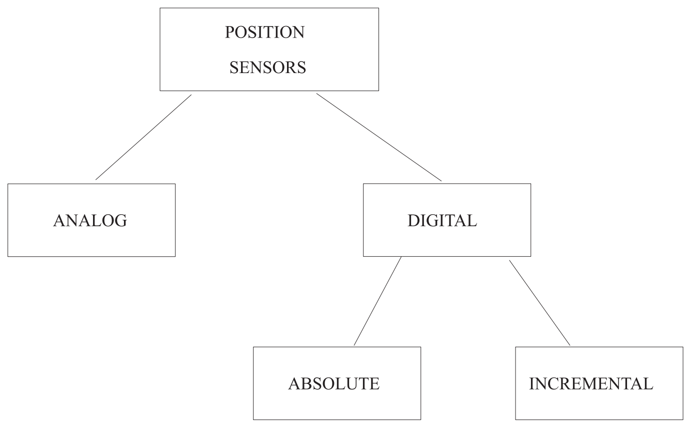


Figure 2.13.3: The various methods for measuring joint positions

Figure 2.13.4: Digital sensors: (a) incremental encoder (b) absolute encoder

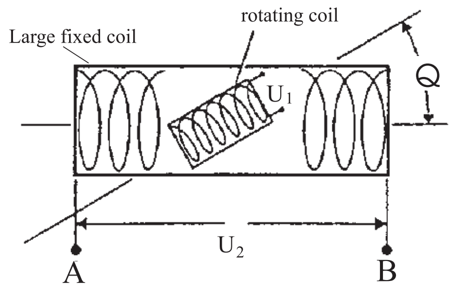


Figure 2.13.5: Inductive potentiometer

configuration parameter. Compared to incremental encoders, they are much more difficult to construct (figure 2.13.4(a)).

Thermal dilatation is a serious problem for linear encoders. One prefers thus to use rotary encoders since angular values are insensitive to temperature. It is possible to find rotary encoders of incremental type numbering from 40 to 40000 steps/turn, and absolute encoders providing an information (on one turn) encoded on 6 to 16 bits.

The accuracy obtained with digital sensors is quite comparable to that given by analog sensors. For similar accuracy, incremental encoders are much cheaper than absolute ones. However, due to their incremental principle, they loose the positioning information when the system is shut down. It is then necessary to reinitialize the reference position of the robot.

- *Analog position sensors*

  Analog position sensors integrate in the same design the sensor itself which delivers an analog signal proportional to the measured position and the analog-to-digital converter which provides a digitized value. Three types of analog position sensors are widely used in robotics:

  1. *Resistive potentiometers:*

     Of linear or rotary type, they convert the cursor position into a proportional D.C. tension. A analog-digital converter converts then this tension into a digital signal. The resistive potentiometers made of a conducting plastic track have a quasi-infinite resolution, their linearity is excellent (0.1%) and their resistance to wear is very high. They are widely used because of their relatively low cost.

  2. *Inductive potentiometers:*

     They consist of a transformer with variable coupling which delivers a tension proportional to the displacement of the mobile winding. They exist in rotary version as shown by figure 2.13.5 and in linear form. The analog-to-digital converter has to achieve the demodulation of the output signal before transforming it into a form suitable to the control system. A linearity of about 0.1% is obtained on a limited displacement range (a few cm in translation and a few tens of degrees in rotary motion). Their use is thus limited to short displacement ranges.

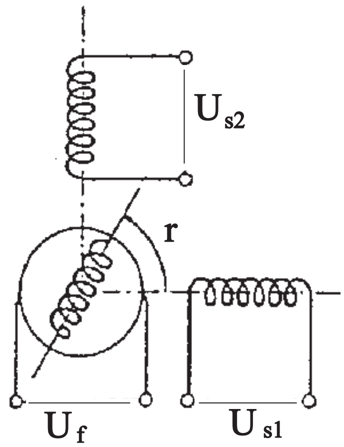


Figure 2.13.6: Principle of the resolver

3. *Resolvers:*

   They are also transformers with variable coupling, and they exist in both linear and rotary versions. The stator is made of two coils as represented by figure 2.13.6, and both are fed with alternative voltages set at 90 deg of each other. They deliver thus at the output of the mobile winding a voltage with a phase (measured from a given reference) proportional to the displacement. Since they are based on a phase measurement, the associated converter is relatively complicated. It delivers either directly one digital signal proportional to the phase or two signals giving both cosine and sine of the phase angle.

   Although any analog sensor device is based on the measurement of an continuous signal, in all cases a analog-to-digital converter digitizes the signal measured. In practice, the digitalization is performed on 10 to 12 bits. This choice corresponds to an optimum since a finer digitalization would generate too much noise in the signal fed to the control unit.

2.13.2 Velocity sensors

Figure 2.13.7 summarizes the different methods to measure velocities on a robot arm. If we except the measurement of velocities using a tachometer, all the systems generally used are based on the use of a position sensor of incremental type.

- *velocity measurement by pulse counting*

  Since velocity is equal to displacement divided by time, there are two ways of measuring them. One can either measure the delay between two successive pulses, or count the number of pulses occurring during a specific interval. Both methods have their advantages and drawbacks

- *The encoder pulse counting method*

  A counter is active during a fixed time interval t and counts the number of pulses generated

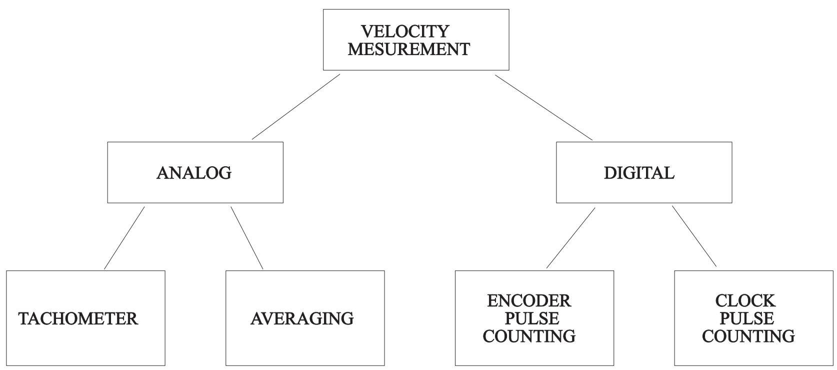


Figure 2.13.7: The various methods for measuring joint velocities

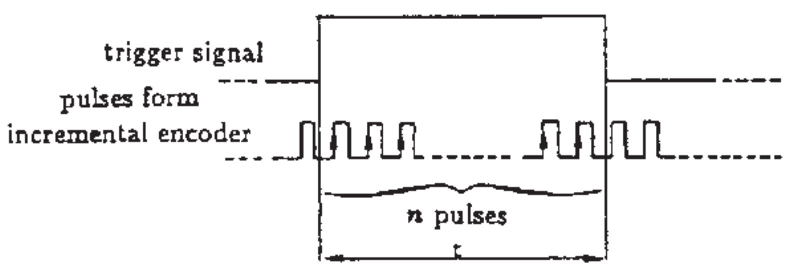


Figure 2.13.8: Velocity measurement by encoder pulse counting

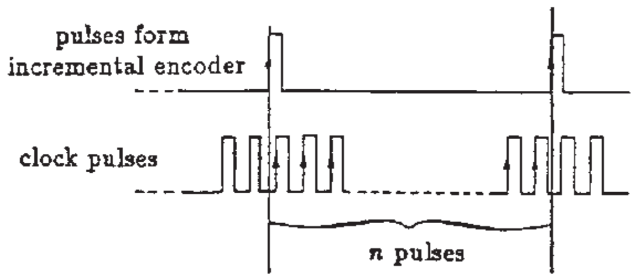


Figure 2.13.9: Velocity measurement based on clock pulse counting

by the encoder during that period (figure 2.14.8). The measured velocity (in m/s or rad/s) is then given by the formula

$$V = \frac{n}{p\,t} \tag{2.13.1}$$

where

- n is the number of pulses detected;
- p is the number of steps (per meter or per radian) of the incremental encoder;
- t is the time interval.

The relative error affecting the measurement is given by

$$\frac{\Delta V}{V} = \frac{\Delta n}{n} + \frac{\Delta t}{t} + \frac{\Delta p}{p} \simeq \frac{\Delta n}{n} \tag{2.13.2}$$

since p is known exactly and $\frac{\Delta t}{t}$ is negligible by comparison with $\frac{\Delta n}{n}$. Moreover, since Δn is equal to 2, the relative error on V is inversely proportional to n. The measurement has thus increasing accuracy with n. To increase n is equivalent to

- Using an incremental encoder with a large number of steps per meter or radian,
- Measuring large velocities,
- Increasing the counting time t, which however alters the instantaneous character of the measurement since it has then the meaning of an average velocity over t.

In practice, with the encoder pulse counting method the sensor has to be mounted on the actuator rather than at the output of the transmission device.

- *The clock pulse counting method*

A counter, validated during the time interval occurring between two successive pulses of the incremental encoder, counts the pulses emitted by a high frequency clock (figure 2.13.9) The measured velocity (in m/s or rad/s) is then given by

$$V = \frac{f}{n\,p} \tag{2.13.3}$$

in which

- n is the number of pulses detected;

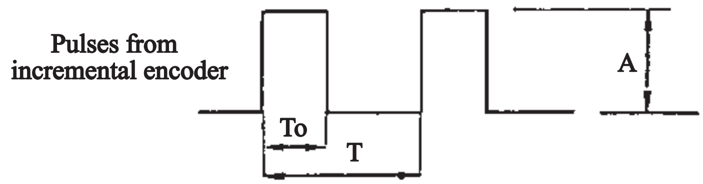


Figure 2.13.10: Velocity measurement based on averaging

- f is the clock frequency;
- p is the number of steps per meter or per radian of the encoder.

Again, the relative error is

$$\frac{\Delta V}{V} = \frac{\Delta n}{n} + \frac{\Delta f}{f} + \frac{\Delta p}{p} \simeq \frac{\Delta n}{n} \tag{2.13.4}$$

since the clock frequency and the number of steps per length unit are exactly known. This relationship shows that the measurement accuracy improves with greater n, which can be obtained by

- Using an incremental encoder with small number per length unit, which is a limitative procedure since the measurement looses its instantaneous character;
- Increasing f, but there is a technological limit to the clock frequency;
- Measuring slow velocities, which can be obtained by mounting the incremental encoder on the joint axis.

- *Analog velocity measurement through averaging*

  Let us suppose that the pulses generated by an incremental encoder are perfectly calibrated both in amplitude A and in duration T_0 (figure 2.13.10) If T is the time elapsed between two successive pulses, the average value of the output signal E_m on the period T is equal to

$$E_m = \frac{AT_0}{T} \tag{2.13.5}$$

  On the other hand, the velocity can be evaluated by

$$V = \frac{1}{pT} \tag{2.13.6}$$

  where p is the number of steps per unit length of the encoder. After elimination of the period T between equation 2.13.5 and equation 2.13.6, one obtains a relationship between velocity and the average value of the output signal

$$V = \frac{E_m}{pAT_0} \tag{2.13.7}$$

  Velocity is thus proportional to the average value of the output signal which can be generated through a low-pass filter.

  Let us note that the physical condition $E_m < A$ has the consequence that the maximum measurable velocity is equal to $\frac{1}{pT_0}$.

- *Analog velocity measurement using a tachometer*

  A tachometer is a D.C. machine which provides a D.C. tension proportional to the rotation velocity of its armature. Its sensitivity is generally of the order of a few volts per 1000 rpm.

  Its use is thus reserved to the measurement of relatively high velocities and it has to be mounted on the actuator axis.

  It has good linearity properties ($\simeq 10^{-3}$), but its use remains limited for cost reasons.

  It has to be associated with an analog-to-digital converter in order to provide a measurement into suitable digital form for further treatment by the control unit.

2.14 The robot manipulator ASEA-IRb-6

In order to illustrate the preceding considerations, let us consider as an example the robot manipulator ASEA-IRb-6 represented by figure 2.14.1 It possesses a mechanical architecture in which the main kinematic chain is of RRRRR type. Its wrist possesses only 2 DOF in the standard version. The five degrees of freedom correspond to:

1. the waist rotation ψ producing arm sweep,
2. the shoulder rotation ϕ producing upper-arm motion,,
3. the elbow rotation α producing lower-arm motion,,
4. the wrist bending t,
5. the wrist twist v.

The manufacturer has opted for the choice of electrical motorization (servo-controlled D.C. motors) centralized in the basis. Therefore, the different DOF are actuated through secondary kinematic chains parallel to the main one.

The displacement ranges of the successive DOF and the associated velocities are:

waist rotation	340°	95 °$/s$
shoulder rotation	±40°	radial velocity of 0.75m/s
elbow rotation	±25°	vertical velocity of 1.1m/s
wrist bending	±90°	115°$/s$
wrist twist	±180°	195°$/s$

The corresponding workspace is represented hereafter in perspective view on one hand, and through a vertical plane cut on the other hand. The manufacturer does not specify the accuracy of the machine, but its repeatability is $0.2mm$. A certain number of mechanical characteristics are given by the technical sheet provided at the end of the chapter.

Mechanical design

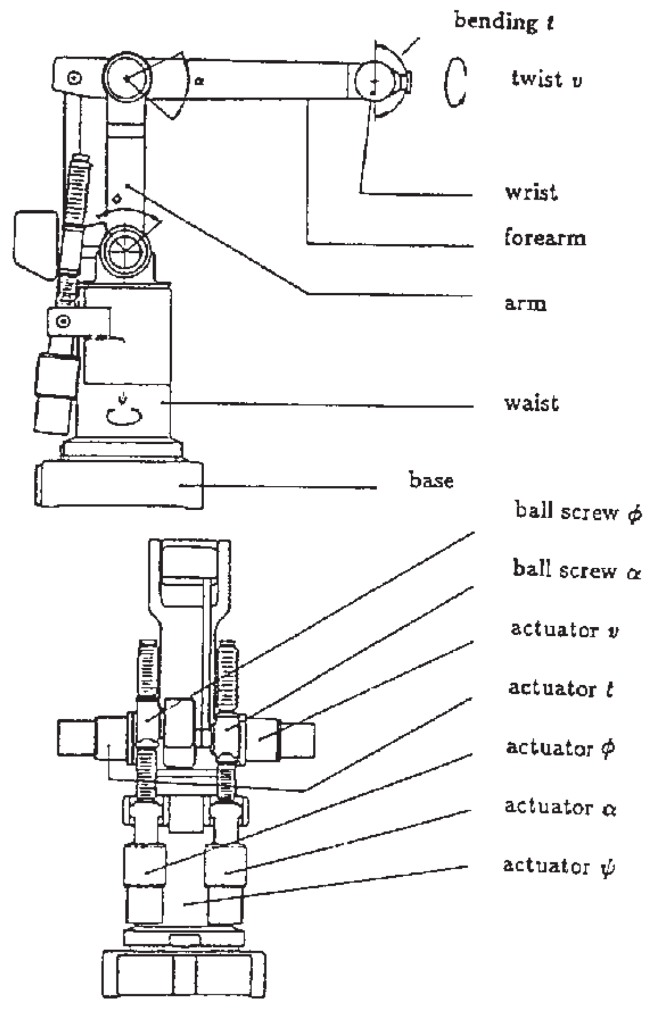


Figure 2.14.1: Mechanical design of the robot ASEA IRb-6

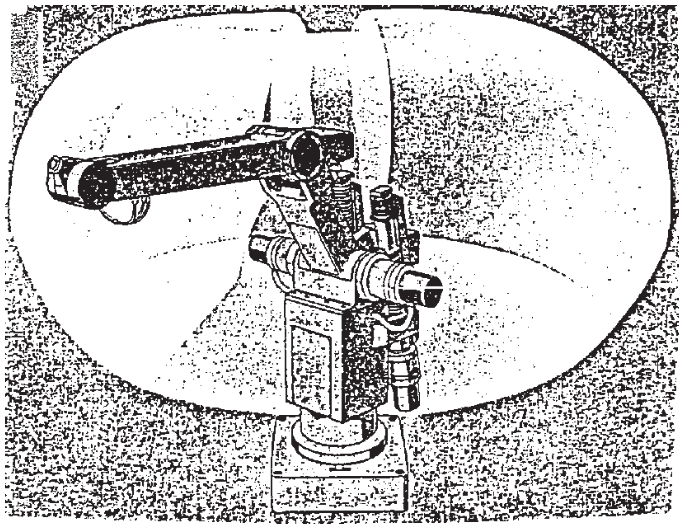


Figure 2.14.2: Workspace of the ASEA robot IRb-6: perspective representation

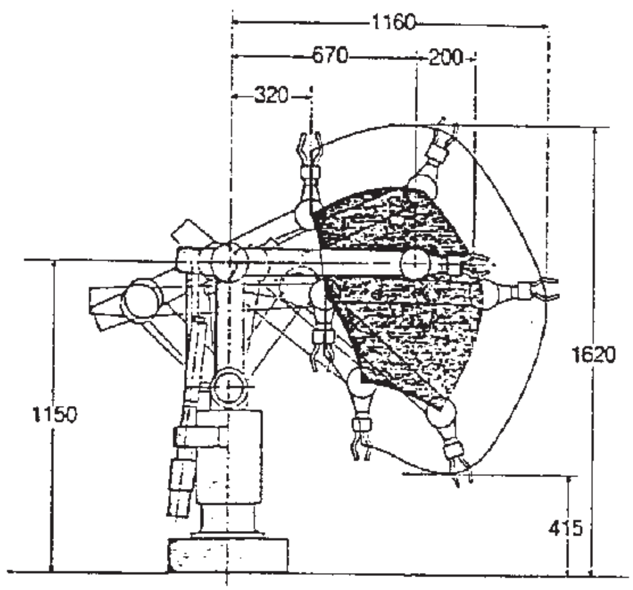


Figure 2.14.3: Workspace of the ASEA robot IRb-6: vertical cut representation

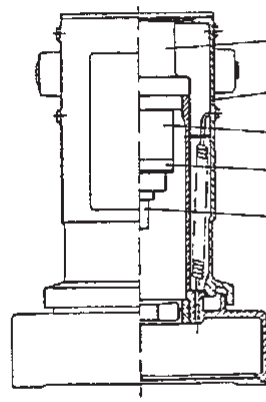


Figure 2.14.4: Motor unit for waist rotation

Figure 2.14.5: Four bar mechanisms for motion transmission

Figure 2.14.4 displays the mechanical design of the first DOF: the whole motor unit is rigidly attached to the base of the manipulator and generates rotary motion of the base through a gear reductor. The tachometer provides velocity information and the resolver, positioning information. The motor units for the DOFs ϕ and α (shoulder and elbow rotations) are also localized in the base, the motion transmission being achieved through four bar linkages which constitute the most elementary closed-loop mechanisms (figure 2.14.5). Figure 2.14.6 displays the motor unit of the shoulder: the actuator rotary motion is transformed into translation motion through a *ball screw* whose nut acts on a lever hinged to the arm. In this way, a rotation-rotation transformation of motion with a high reduction rate is obtained while allowing us to locate the motor unit of the shoulder in the base of the manipulator. The ball screw is a very efficient implementation of the screw joint (figure 2.14.7). It is made of a helicoidal screw whose thread is a rolling path for the balls and a nut containing them. The balls are recirculated from one end of the nut to the other through external ducts.

The motor unit of the elbow is designed on the same principle (figure 2.14.8). An additional mechanical loop is however needed to transmit the motion to the end of the arm.

The motor units for wrist bending and twist are also located in the base of the structure, and the motion transmission from the base to the wrist is obtained through a parallelogram mechanism, as shown by figure 2.14.9 The transmission is made of three quaternary links having the form

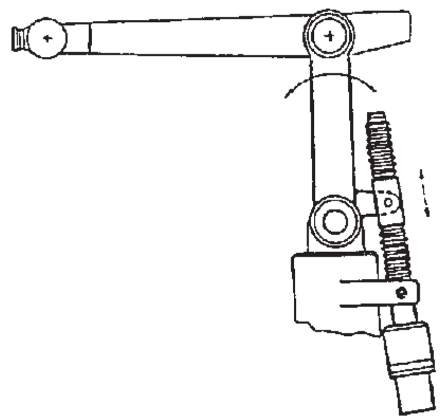


Figure 2.14.6: Motor and transmission unit for shoulder rotation

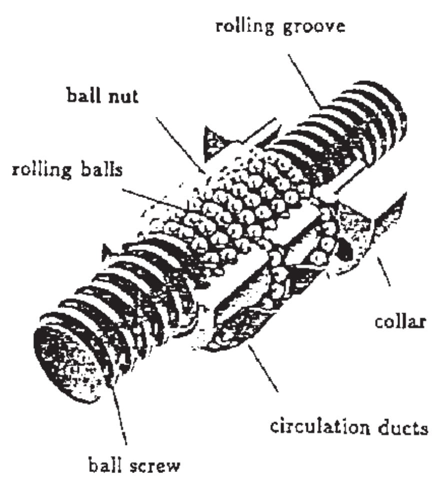


Figure 2.14.7: Ball screw implementation of the screw joint

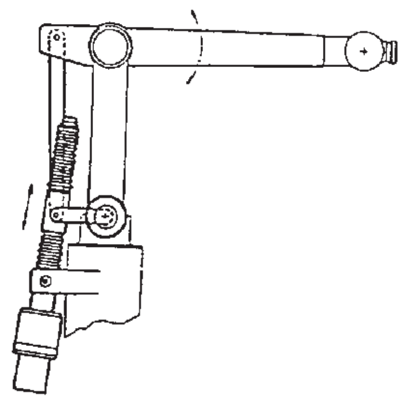


Figure 2.14.8: Motor and transmission unit for elbow rotation

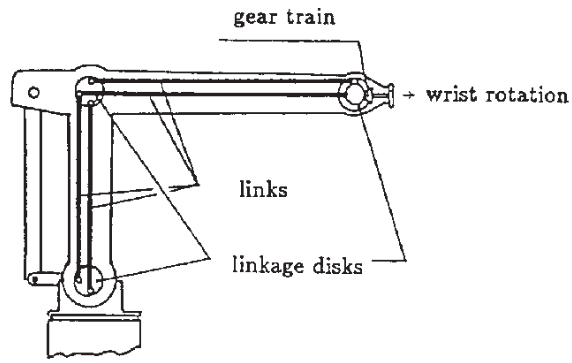


Figure 2.14.9: Transmission unit for wrist bending

of disks and located at motor, elbow hinge and wrist levels, respectively. They are connected together through parallel links which are hinged on the disks at points distant from 90°. Two similar linkages going through the main structure are necessary to transmit both bending and twist DOF (figure 2.14.9). For the wrist DOFs, the speed reduction is achieved first through gear reductors of *harmonic drive* type. Harmonic drives are highly performing reductors based on a rather simple but clever principle. They are made of three parts:

- the output element is a *rigid planetary* of annular shape with internal teeth ;
- the input element is a *elliptic wave generator* mounted on a roller bearing of same shape;
- the intermediary element is a *deformable satellite*, with a number of teeth slightly inferior (generally, 2) to that of the planetary.

One can show that, if Z_p and Z_s are respectively the numbers of teeth on the planetary and on the satellite, the reduction ratio is given by the formula

$$N = \frac{Z_p}{Z_p - Z_s} \qquad (2.14.1)$$

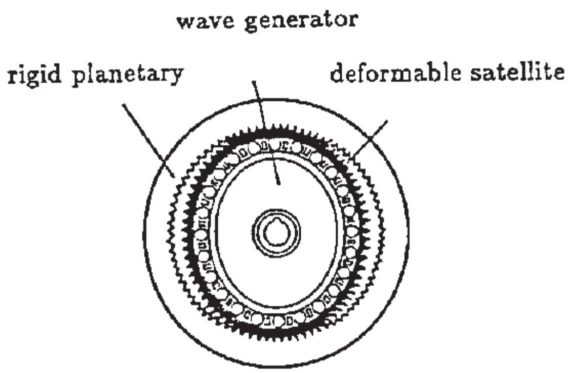


Figure 2.14.10: Reductor of harmonic drive type

In this way, reduction ratios as high as 320 can be obtained in a single reductor stage. The number of teeth participating in the contact is rather high (about 10 %); this is favorable to wear and backlash reduction and increases at the same time the mechanical stiffness.

2.15 Technical sheets of some industrial robot manipulators

Starting from the technical data provided by manufacturers, it is generally possible to summarize the characteristics of a specific manipulator in a technical sheet made on the same model as the ones given hereafter. However, it is not always possible to gather all the data which describe completely the manipulator under consideration.

2.15.1 Industrial robot ASEA IRb-6/2

KINEMATIC ARCHITECTURE: 5 R
DEGREES OF FREEDOM: q_1 R 340° $90°/s$ q_2 R 80° $1m/s$ q_3 R 50° $1.35m/s$ q_4 R 180° $90°/s$ q_5 R 360° $150°/s$
MOTORIZATION: D.C. servomotors
MECHANICAL PERFORMANCES: capacity: 6kg max. linear velocity: r direction 0.75 m/s; z direction 1.1m/s. repeatability: 0.2 mm accuracy: not specified
EFFECTOR: various grippers adapted to specific applications
SENSORS: resolvers + tachometers
CONTROL UNIT: central unit with 16K ram
PROGRAMMING MODE: - teach box - computer link
APPLICATIONS: - arc welding - assembly - inspection - machining - surface coating - glueing
Industrial robot ASEA IRb-6/2: technical data

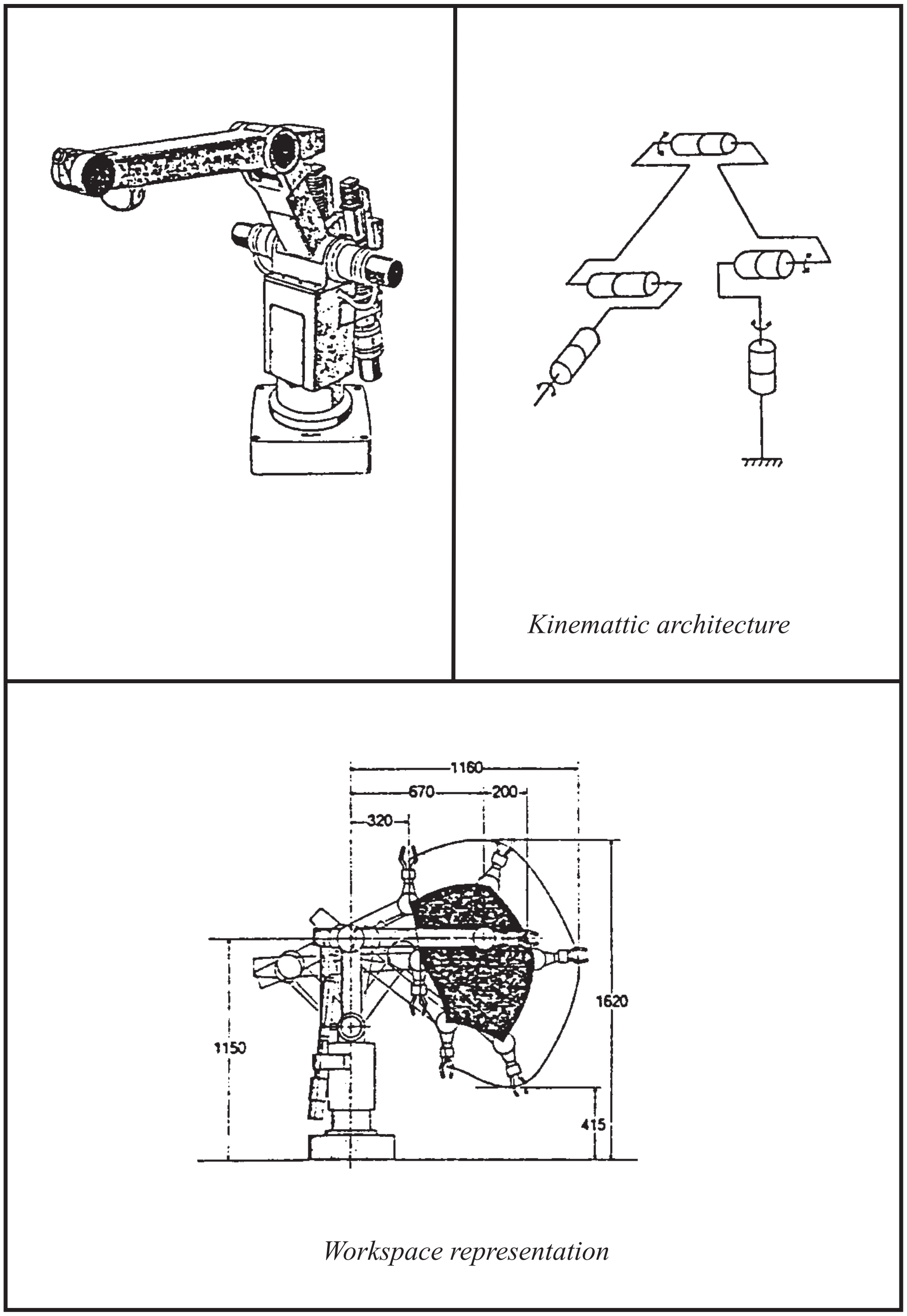


Figure 2.15.1: Industrial robot ASEA IRb-6/2: geometric configuration

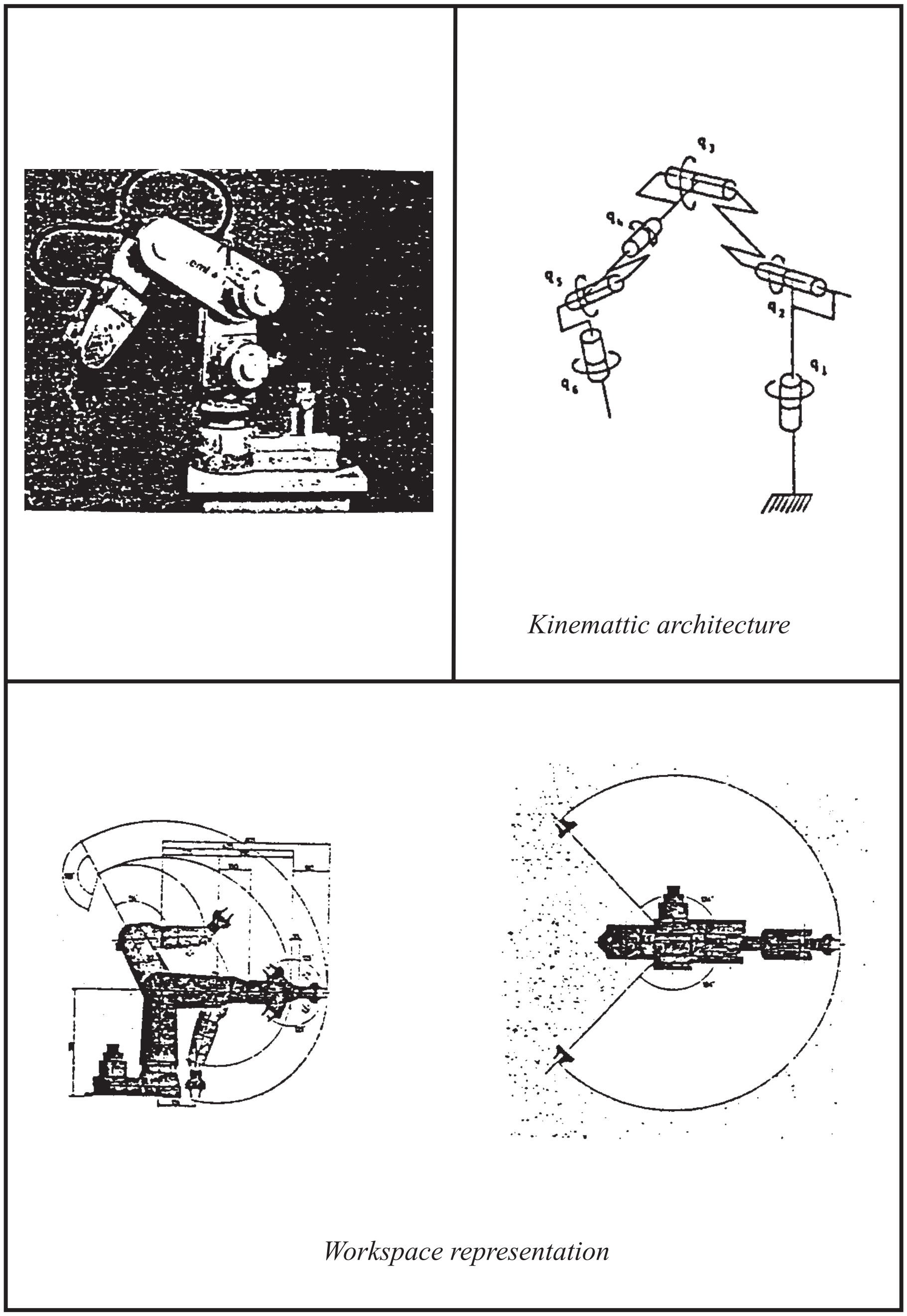


Figure 2.15.2: Industrial robot SCEMI 6P-01: geometric configuration

2.15.2 Industrial robot SCEMI 6P-01

KINEMATIC ARCHITECTURE: 6 R			
DEGREES OF FREEDOM:			
q_1	R	268°	$233°/s$
q_2	R	120°	$233°/s$
q_3	R	116°	$233°/s$
q_4	R	360°	$233°/s$
q_5	R	250°	$108°/s$
q_6	R	360°	$233°/s$
MOTORIZATION: D.C. servomotors			
MECHANICAL PERFORMANCES: capacity: 1.5kg max. linear velocity: 2.7m/s. repeatability: 0.04 mm accuracy: not specified			
EFFECTOR: standard gripper (electrically or pneumatically actuated)			
SENSORS: incremental encoders (1000 and 800 steps/turn)			
CONTROL UNIT: central unit LSI - 11/2 with 64K ram 1 microprocessor per axis, position/velocity control			
PROGRAMMING MODE: - teach box - programming language LM - via computer link, language $RS\ 232 - C$			
APPLICATIONS: -flexible manufacturing; - assembly - inspection; - palletizing - component insertion; - research and teaching			

Industrial robot SCEMI 6P-01: technical data

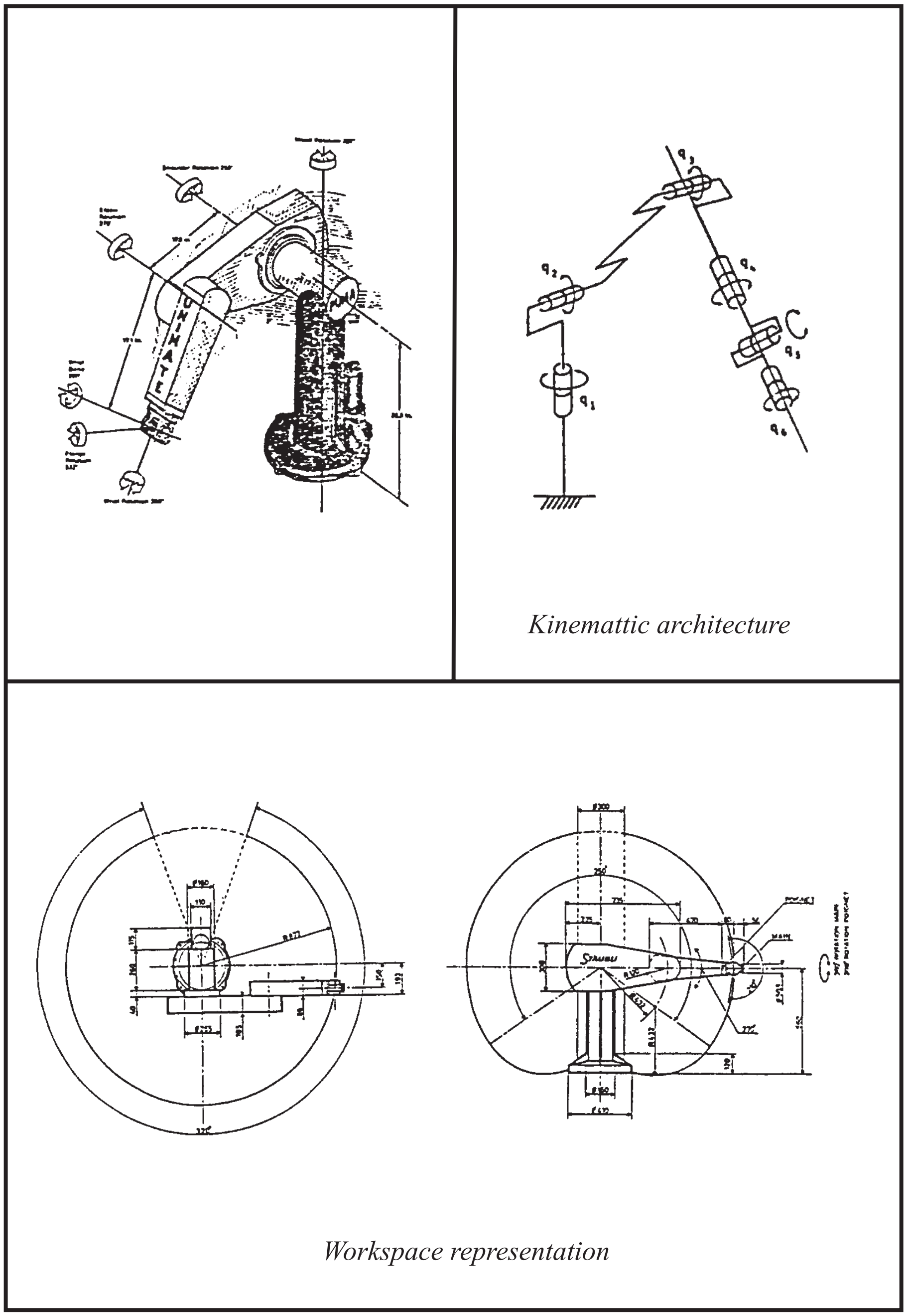


Figure 2.15.3: Industrial robot PUMA 560 Geometric configuration

2.15.3 Industrial robot PUMA 560

KINEMATIC ARCHITECTURE: 6 R

DEGREES OF FREEDOM:

q_1	R	320°	$80°/s$
q_2	R	250°	$52°/s$
q_3	R	270°	$108°/s$
q_4	R	300°	$230°/s$
q_5	R	200°	$108°/s$
q_6	R	360°	$453°/s$

MOTORIZATION: D.C. servomotors

MECHANICAL PERFORMANCES:
capacity: 2.5kg
max. linear velocity: 1.0m/s.
repeatability: 0.1 mm
accuracy: not specified

EFFECTOR: standard gripper (optional, pneumatically actuated)

SENSORS: incremental encoders (0.73 to 1.1710^{-4} rad/bit)

CONTROL UNIT: central unit LSI - 11/2 with 16 or 32K ram
1 microprocessor per axis

PROGRAMMING MODE:
- teach box
- programming language VAL^{TM}
- via computer link, language $VAL\ II^{TM}$

APPLICATIONS:
- arc welding
- assembly
- inspection
- machining
- component insertion
- research and teaching
- palletizing

Industrial robot PUMA 560: technical data

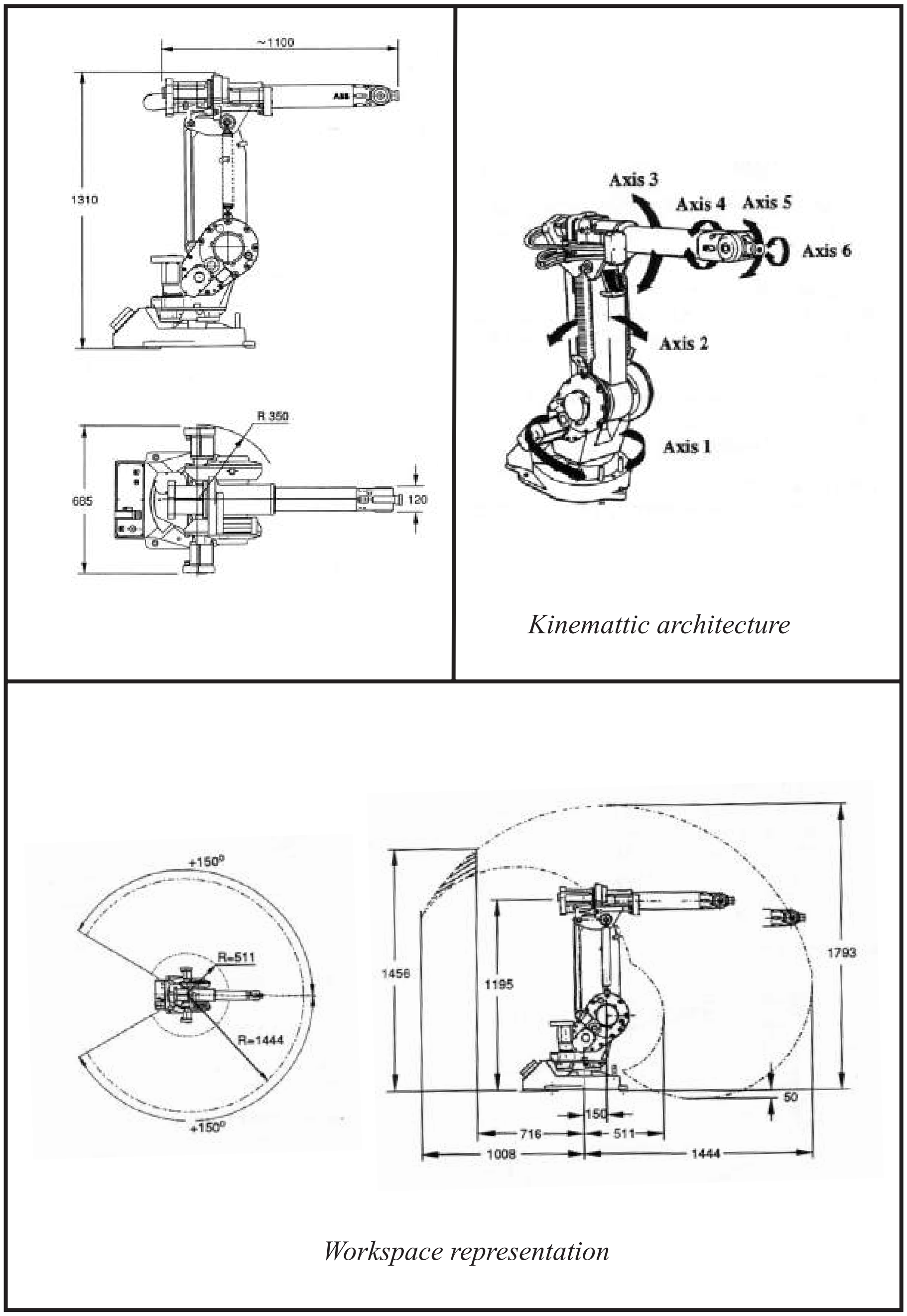


Figure 2.15.4: Industrial robot ASEA-IRB 1400 Geometric configuration

2.15.4 Industrial robot ASEA IRB1400

KINEMATIC ARCHITECTURE: 6 R
DEGREES OF FREEDOM: q_1 R ±150° 110°/s q_2 R ±70° 110°/s q_3 R 70° – −65° 110°/s q_4 R ±150° 280°/s q_5 R ±115° 280°/s q_6 R ±300° 280°/s
MOTORIZATION: D.C. servomotors
MECHANICAL PERFORMANCES: capacity: max 5.kg max. linear velocity: 1.0m/s. repeatability: 0.1 mm (during unidirectional pose) accuracy: 0.2mm (during unidirectional pose) stabilization time: $< 0.15s$
EFFECTOR:
SENSORS:
CONTROL UNIT: central unit LSI - 11/2 with 4 Mbyte (basic) RAM extended to 10 Mbyte
PROGRAMMING MODE: - RAPID programming language
APPLICATIONS: - flexible manufacturing - research and teaching - arc welding - assembly
Industrial robot IRB-1400: technical data

Bibliography

[1] P. ANDRE, J. M. KAUFMAN, F. LHOTE, J. P. TAILLARD, *Les robots: Constituants Technologiques*, tome 1, Hermes Publishing, Paris, 1983.

[2] H. ASADA, J. J. SLOTINE, *Robot analysis and control*, John Wiley and Sons, 1985.

[3] P. CHEDMAIL, E. DOMBRE, P. WENGER, *La CAO en Robotique, outils et Méthodologies*, Hermes Publishing, Paris, 1998.

[4] P. COIFFET, *Les robots, modélisation et commande*, tome 1, Hermes Publishing, Paris, 1981.

[5] P. COIFFET, M. CHIROUZE, *An introduction to robot technology*, Kogan Page, London, 1983.

[6] J. J. CRAIG, *Introduction to robotics: Mechanics and Control*, Addison-Wesley, 1985.

[7] F. FAGE "Statistiques 97", *RobAut*, No 21, April-March 1998, pp 28–32.

[8] B. GORLA, M. RENAUD, *Modèles des robots manipulateurs. Application à leur commande*, Cepadues Editions, Toulouse, 1984.

[9] *Handbook of Industrial Robotics*, edited by S. Y. NOF, John Wiley and Sons, 1985.

[10] R. S. HARTENBERG, J. DENAVIT, *Kinematic Synthesis of Linkages*, Mc Graw Hill, 1964.

[11] M. HILLER, *Computer-Aided Kinematics and Dynamics of Multibody Systems*, Chaire Francqui Internationale, University of Liège, March-April 1991.

[12] S. HIROSE, Y. UMETANI, "A Cartesian coordinates Manipulator with articulated Structure", 11th ISIR Conference, Tokyo, Japan, 1981.

[13] R. KAFRISSEN, M. STEPHANS, *Industrial Robots and Robotics*, Reston Publishing Cy, Reston, Virginia, 1984.

[14] W. KHALIL, E. DOMBRE, *Modélisation, identification et commandes des robots*, Hermes Publishing, Paris, 1999.

[15] P. MAYE, *Moteurs électriques pour la robotqiue*, Dunod, Paris, 2000.

[16] W. E. SNYDER, *Industrial Robots: Computer Interfacing and Control*, Prentice Hall, 1985.

[17] Y. C. TSAI, A. H. SONI, "Accessible regions and synthesis of robot arms", Jnl of Mech. Design, Oct. 1981, vol. 103, 803-884.

Chapter 3

BASIC PRINCIPLES OF ROBOT MOTION CONTROL

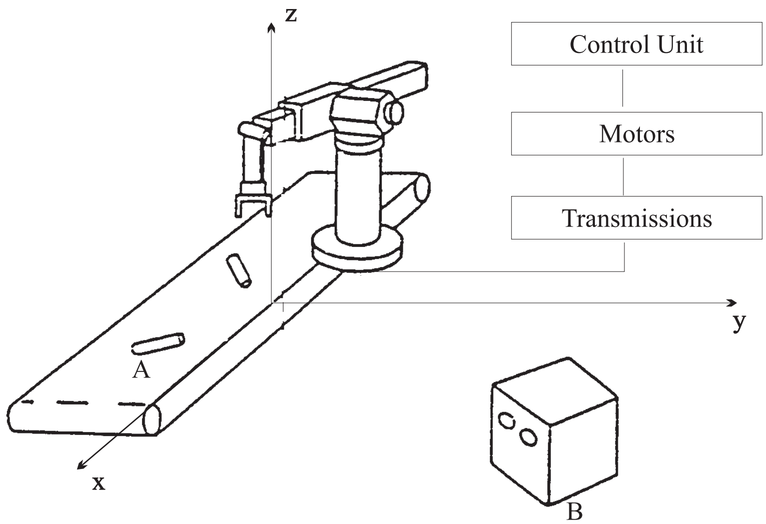


Figure 3.1.1: A typical assembly task

3.1 Objectives of robot control

When considered at a higher level, the objective of robot control is to let it achieve a specified task. Let us consider an elementary assembly task such as suggested in figure 3.1.1. It consists in picking up a cylindrical pin ***A*** on a conveyor and inserting it in a hole made in object ***B***.

It can obviously be decomposed in a sequence of elementary orders such as

< MOVE... >;
< GRASP >;
< MOVE... >;
< RELEASE >;

in which case each motion phase imposes the effector to describe an elementary and well specified operation.

3.1.1 Variables under control

Achieving a task such as described above implies to know at every moment:

- the position, orientation and state (open/ closed) of the gripper;
- the position and orientation of pin ***A***, which depends itself on the instantaneous position of the conveyor;
- the motion of the conveyor;
- the position and orientation of object ***B***;
- the position and orientation of the hole relatively to it.

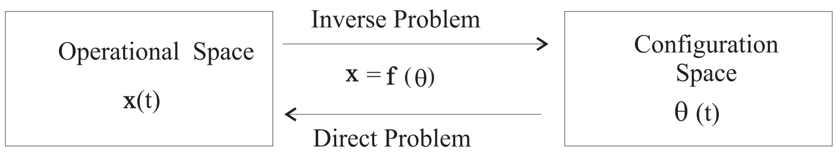


Figure 3.1.2: Fundamental relationship between operational and configuration spaces for a robot manipulator

All these geometric data are generally know in different reference frames. In order to have global control of the task, all of them have to be expressed in a common reference frame which defines the *operational space*, or *task space*.

The sequence of successive configurations of the effector in the operational space defines *its trajectory*. It can be written in terms of a certain number of position, orientation and state parameters functions of time.

$$\boldsymbol{x}(t) \;=\; [\, x_1(t) \ldots x_n(t)\,]^T$$

Each configuration of the effector is obtained through an appropriate modification of the geometrical configuration of the articulated mechanism by acting on the m joint (or configuration) variables.

$$\theta(t) \;=\; [\, \theta_1(t) \ldots \theta_m(t)\,]^T$$

which define the *joint* (or *configuration*) *space*. From a geometrical point of view, controlling the motion of the robot consists in verifying in time the n relationships:

$$\begin{aligned} x_1(t) &= f_1(\theta_1, \ldots \theta_m) \\ x_2(t) &= f_2(\theta_1, \ldots \theta_m) \\ &\vdots \\ x_n(t) &= f_n(\theta_1, \ldots \theta_m) \end{aligned}$$

which establish the correspondence between the *operational space* and *configuration space*. They can be expressed in the matrix form

$$\boldsymbol{x}(t) \;=\; \boldsymbol{f}(\theta) \tag{3.1.1}$$

which constitutes the fundamental relationship for kinematic analysis of robots (Fig. 3.1.2).

As will be seen, equation (3.1.1) always be verified and has a unique solution when calculating $\boldsymbol{x}$ from given θ. This is the *direct problem of kinematics*. However, the solution of the *inverse problem* $\boldsymbol{x} \rightarrow \theta$ does not necessarily exist and, if it does, can be multiple or even in infinite number. Developing the solution of the inverse problem is thus one of the difficult aspects of robot control.

3.1.2 Robot motion control

The process of motion generation in joint space is then the following:

- joint motion $\theta(t)$ results from the action of couples $\boldsymbol{C}(t)$ which are developed in the articulations;

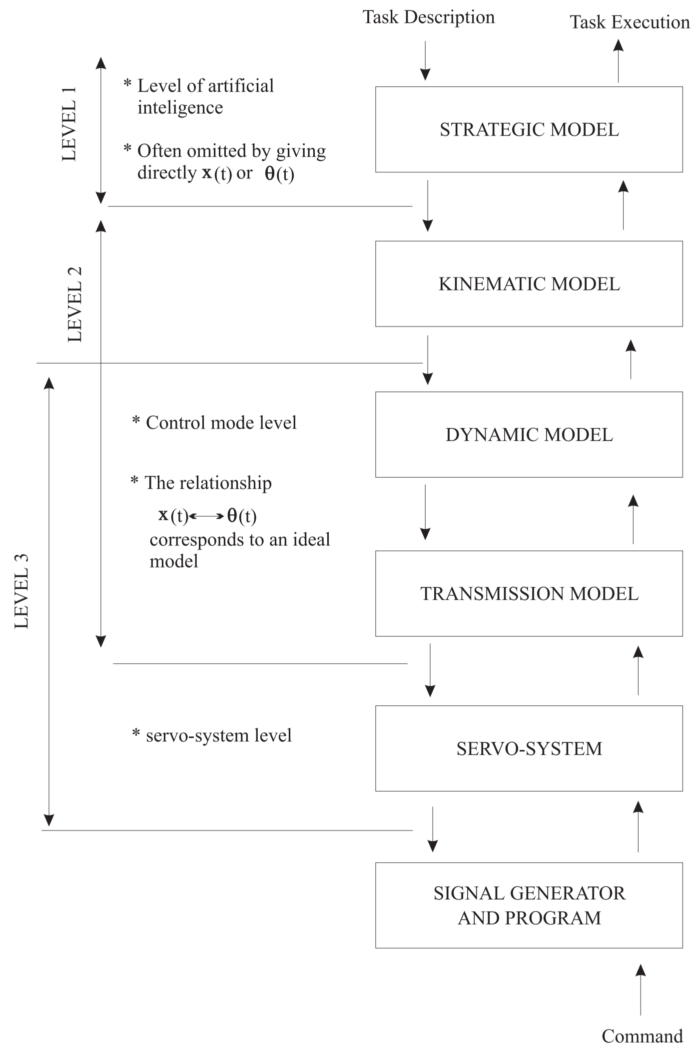


Figure 3.1.3: Three main levels of robot control

- the couples $\boldsymbol{C}(t)$ result themselves from torques $\boldsymbol{F}(t)$ generated by the actuators and transmitted to the articulations by transmission devices (gear trains, mechanical linkages);
- the actuators produce the torques $\boldsymbol{F}(t)$ in response to current or voltage orders sent by the control unit and assembled in a vector $\boldsymbol{V}(t)$.

As a result, controlling robot motion consists essentially in controlling the bi-directional equation.

$$\boldsymbol{V}(t) \leftrightarrow \boldsymbol{F}(t) \leftrightarrow \boldsymbol{C}(t) \leftrightarrow \theta(t) \leftrightarrow \boldsymbol{x}(t) \leftrightarrow \text{task} \tag{3.1.2}$$

Three levels of robot control can be distinguished in equation (3.1.2) which are relevant from different disciplines:

- level 1 which is the *artificial intelligence* level,
- level 2 or the *control mode* level,

- level 3 or the *servo control* level.

Each of them will be briefly described (see Figure 3.1.3).

3.1.3 Level 1 of control: artificial intelligence level

The objective of this first and highest control level consists in translating the sequence of orders sent to the control unit into a trajectory information $\boldsymbol{x}(t)$.

Its degree of evolution depends on the point of view adopted. Coming back to the typical task of figure 3.1.1, a sequence of orders such as

< PICK $\boldsymbol{A}$ >;
< INSERT $\boldsymbol{A}$ *INTO* $\boldsymbol{B}$ >;

is a possible and very concise way of programming the objective. However, its achievement implies solving a certain number of problems which are relevant of artificial intelligence such as:

- *Interpreting the natural language* in terms of which the orders are expressed;
- *Perceiving the state of the robot* and *its environment*;
- *Describing the task in terms of elementary operations*;
- *Planning the motion* and *generating the trajectories.*

This highest level of control is still largely a research subject and is this reduced to its simplest expression in current industrial practice. The normal programming procedure of present industrial robots is to give directly the elements concerning the trajectory (either in configuration space θ or in operational space $\boldsymbol{x}$) during a training phase.

Motion planning and trajectory, generation are the subjects by which the mechanical engineer is the most concerned in this highest level of robot control.

3.1.4 Level 2 of control or the control mode level

This is the level at which the bi-directional relationships between the trajectory in operational space $\boldsymbol{x}(t)$ and the torques $\boldsymbol{F}(t)$ supplied by the actuators can be verified.

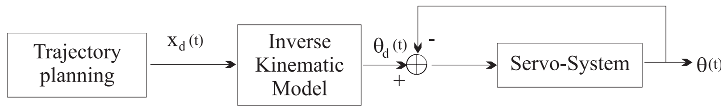
It is important to notice that several modes of control can be imagined. The variety in the proposed solutions arises from the fact that the relationship

$$\boldsymbol{x}(t) \leftrightarrow \theta(t) \leftrightarrow \boldsymbol{C}(t) \leftrightarrow \boldsymbol{F}(t) \tag{3.1.3}$$

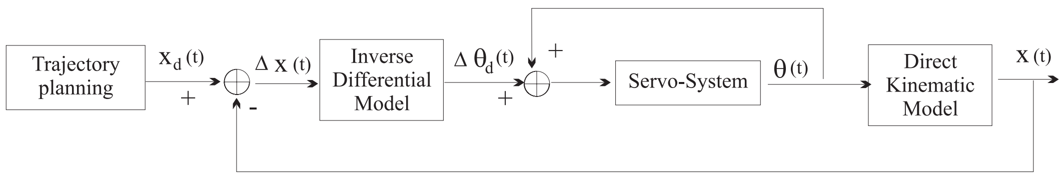
pose various problems in practice. They differ mainly by the level of knowledge of system that one accepts to integrate in the model.

In particular,

- The relationship $\boldsymbol{F}(t) \leftrightarrow \boldsymbol{C}(t)$ involves an appropriate model of transmissions;
- Controlling the relationship $\boldsymbol{C}(t) \leftrightarrow \theta(t)$ implies using a *dynamic model* of the articulated structure;
- Controlling the relationship $\theta(t) \leftrightarrow \boldsymbol{x}(t)$ relies upon a *kinematic model* of the mechanism.



(a) Position Control



(b) Velocity control

Figure 3.1.4: Comparison of the principles of position control and velocity control

s

Among these various models, the dynamic one is the most difficult to build and use for mainly two reasons:

- There is no way of knowing how to model the imperfections of the mechanical components correctly (mainly flexibility, friction and back-lash in joints);
- Even if it seemed possible to take these into account, the model would include hundreds of parameters and the processor would be unable to perform all the necessary in-line operations at an adequate speed.

This explains why in most present industrial robots only two control modes are generally applied which correspond to the use of two types of models:

- Position control which consists in assuming that the robot passes through a sequence of predetermined states $x_1, x_2 \ldots x_n$ corresponding to configurations $\theta_1, \theta_2 \ldots \theta_m$. Position control makes thus use of the geometric kinematic model describing equation (3.1.1) and implies inverting it for all specified states;
- Velocity control makes use of a differential kinematic model which results from a linearization of the kinematic model. The differential model relates displacement increments $\Delta\boldsymbol{x}$ in operational space to joint displacement increments in $\Delta\theta$ configuration space.

Figure 3.1.4 illustrates the principles of position control and velocity control and shows the difference between them.

3.1.5 Level 3 of control or servo-system level

This concerns the standard practive of robotics. The scope of this last level of control is variable however, according to the control mode adopted.

- When the control is based on the use of a dynamic model, what is controlled at this level is one of the bi-directional equations.

$$\boldsymbol{V}(t) \leftrightarrow \boldsymbol{F}(t) \tag{3.1.4}$$

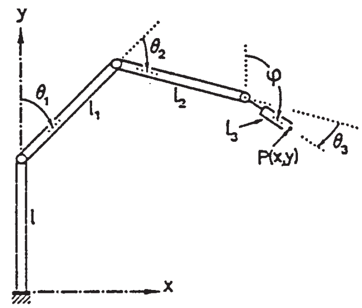


Figure 3.2.1: Planar manipulator with three links

or

$$\boldsymbol{V}(t) \leftrightarrow \boldsymbol{F}(t) \leftrightarrow \boldsymbol{C}(t) \tag{3.1.5}$$

Let us note that the current tendency in research is to design motors with integral stepdown gears, in which case it is the relationship (3.1.4) which is controlled directly.

- In the case of control based on a kinematic model, the only way to control the dynamics of the mechanical structure is through the servo-system.

3.2 Kinematic model of a robot manipulator

In order to illustrate the concept of kinematic model let us consider the planar robot structure of figure 3.2.1. It has three hinge joints with rotation axis along z directions. Let θ_1, θ_2, and θ_3 be the corresponding angular displacements.

The operational space is defined here by the position (x, y) of the end point of the effector and its orientation. It can be expressed by three equations.

$$\begin{aligned} x =& \ l_1 \sin\theta_1 + l_2 \sin(\theta_1+\theta_2) + l_3 \sin(\theta_1+\theta_2+\theta_3) \\ y =& \ l_1 \cos\theta_1 + l_2 \cos(\theta_1+\theta_2) + l_3 \cos(\theta_1+\theta_2+\theta_3) + l \\ \phi =& \ \theta_1+\theta_2+\theta_3 \end{aligned} \tag{3.2.1}$$

or, in matrix form

$$\boldsymbol{x} \ = \ \boldsymbol{f}(\theta) \tag{3.2.2}$$

where $\boldsymbol{x} = [\, x \ y \ \theta \,]^T$ represents the operational space, and $\theta = [\, \theta_1 \ \theta_2 \ \theta_3 \,]^T$ defines the configuration space. The system of equations (3.2.1) is the *geometric kinematic model* of the manipulator.

Its inversion is straight forward when calculating first θ_2 and θ_1, what can be made by evaluating the position of point $P' = (x', y')$ at the wrist hinge. From the coordinates of P'

$$\begin{aligned} x' =& \ x \ - \ l_3 \sin\phi \\ y' =& \ y \ - \ l_3 \cos\phi \ - \ l \end{aligned} \tag{3.2.3}$$

or

$$\begin{aligned} x' &= l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) \\ y' &= l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) \end{aligned} \tag{3.2.4}$$

Taking the sum of these two equations, one deduces

$$x'^2 + y'^2 \;=\; l_1^2 \;+\; l_2^2 \;+\; l_1 l_2 \;\left[\cos\theta_1 \cos(\theta_1+\theta_2) \;+\; \sin\theta_1 \sin(\theta_1+\theta_2)\right]$$

Two solutions for θ_2:

$$\theta_2 \;=\; \pm\,\cos^{-1}\left(\frac{x'^2 + y'^2 - l_1^2 \;-\; l_2^2}{2\,l_1\,l_2}\right) \tag{3.2.5}$$

To identify θ_1, let us note that

$$\frac{x'}{y'} \;=\; \frac{l_1 \sin\theta_1 \;+\; l_2 \sin(\theta_1+\theta_2)}{l_1 \cos\theta_1 \;+\; l_2 \cos(\theta_1+\theta_2)} \tag{3.2.6}$$

which can be rewritten

$$\frac{x'}{y'} \;=\; \frac{l_1 \tan\theta_1 \;+\; l_2 \tan\theta_1 \cos\theta_2 \;+\; l_2 \sin\theta_2}{l_1 +\; l_2 \cos\theta_2 \;-\; l_2 \tan\theta_1 \sin\theta_2} \tag{3.2.7}$$

and solved with respect to θ_1. One can see at first that:

$$\tan\theta_1 \;=\; \frac{\frac{x'}{y'}(l_1 + l_2 \sin\theta_2) \;-\; l_2\; sin\theta_2}{l_1 \;+\; l_2 \sin\theta_2 \;+\; \frac{x'}{y'} l_2 \sin\theta_2}$$

The solution can be simplified if one notices that $\tan\theta_1 \;=\; \tan(\alpha-\beta)$ with $\tan\alpha = x'/y'$ and $\tan\beta = \frac{l_2 \sin\theta_2}{l_1 + l_2 \cos\theta_2}$. It comes:

$$\theta_1 \;=\; \tan^{-1}\left(\frac{x'}{y'}\right) \;-\; \tan^{-1}\left(\frac{l_2 \sin\theta_2}{l_1 + l_2 \cos\theta_2}\right) \tag{3.2.8}$$

Finally the degree of freedom describing the orientation of the effector is calculated by

$$\theta_3 \;=\; \phi - (\theta_1 + \theta_2) \tag{3.2.9}$$

Analyzing the results (3.2.3-3.2.9) shows that the inverse kinematic model is characterized by the following properties:

1. It is described by a system of highly nonlinear equations. Its *closed form* solution has implied a specific step which consisted in decoupling the parameter describing effector orientation. Its solution could have also be obtained *numerically* but would have raised the problem of robustness for the algorithms used.

2. *A solution exists* only as far as the point $P' = (x', y')$ is located *inside the workspace* defined at the level of the wrist hinge; equations (3.2.3) and (3.2.5) shows that the necessary condition is here

$$(l_1 - l_2)^2 \;<\; (x - l_3 \sin\phi)^2 \;+\; (y - l - l_3 \cos\phi)^2 \;<\; (l_1 + l_2)^2 \tag{3.2.10}$$

   and depends on effector orientation.

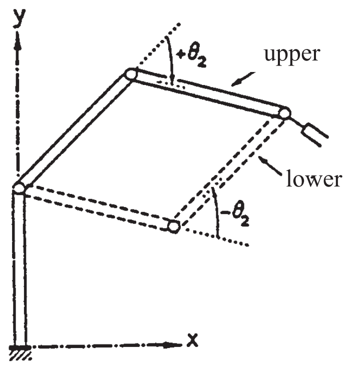


Figure 3.2.2: Upper-and lower-arm configurations of the 3 DOF manipulator

The dexterous workspace, which can be described as the part of the workspace which is reachable with arbitrary effector orientation, is further restrained by both conditions

$$\begin{aligned} (\sqrt{x^2+(y-l)^2}\ +\ l_3)^2 &\leq (l_1\ +\ l_2)^2 \\ (\sqrt{x^2+(y-l)^2}\ -\ l_3)^2 &\leq (l_1\ -\ l_2)^2 \end{aligned} \tag{3.2.11}$$

This can be show by looking for the extrema of

$$g(\phi)\ =\ (x-l_3\sin\phi)^2\ +\ (y-l-l_3\cos\phi)^2$$

3. Inside the workspace, two solutions exist according to the sign chosen for solution equation (3.2.5): they correspond respectively to the upper and lower-arm configurations (Fig. 3.2.2).

4. When $l_1 = l_2$, a degeneracy occurs at point

$$x\ =\ l_3\sin\phi \qquad y\ =\ l+l_3\cos\phi$$

Since the function $\tan^{-1}(\frac{x'}{y'})$ becomes then infinite: θ_2 is the equal to 180° and 0°, has an undetermined value. This corresponds to the situation where location points of joints 1 and 3 come into coincidence.

5. The kinematic model should also take into account the mobility restraints which would occur in an actual design.

The differential kinematic model is obtained in the general case by differentiating (3.1.1) with respect to time in order to obtain velocities in configuration space. For a given component x_i one obtains

$$\dot{x}_i\ =\ \sum_{j=1}^{m}\frac{\partial f_i}{\partial\theta_j}\,\dot{\theta}_j \tag{3.2.12}$$

or, in matrix form

$$\dot{\boldsymbol{x}}\ =\ \boldsymbol{J}\,\dot{\theta} \tag{3.2.13}$$

where the jacobian matrix of the system is defined by

$$\boldsymbol{J}\ =\ [J_{ij}]\ =\ \left[\frac{\partial f_i}{\partial\theta_j}\right]$$

In the particular case of three link manipulator defined in (3.2.1), equation (3.2.12) can be given in the explicit form

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} l_1C_1 + l_2C_{12} + l_3C_{123} & l_2C_{12} + l_3C_{123} & l_3C_{123} \\ -l_1S_1 - l_2S_{12} - l_3S_{123} & -l_2S_{12} - l_3S_{123} & -l_3S_{123} \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \tag{3.2.14}$$

with the definitions

$$\begin{aligned} S_{ij\ldots} &= \sin(\theta_i + \theta_j + \ldots) \\ C_{ij\ldots} &= \cos(\theta_i + \theta_j + \ldots) \end{aligned}$$

Its inversion is also easy to perform in closed form, provided that the velocities at the wrist hinge are extracted from (3.2.14)

$$\begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix} = \begin{bmatrix} \dot{x} \; - \; l_3 \; \cos\phi \; \dot{\phi} \\ \dot{y} \; + \; l_3 \; \sin\phi \; \dot{\phi} \end{bmatrix} \tag{3.2.15}$$

under the form

$$\begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix} = \begin{bmatrix} l_1C_1 + l_2C_{12} & l_2C_{12} \\ -l_1S_1 - l_2S_{12} & -l_2S_{12} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \tag{3.2.16}$$

The inverse form to (3.2.16) is

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \frac{1}{l_1 l_2 S_2} \begin{bmatrix} l_2S_{12} & l_2C_{12} \\ -l_1S_1 - l_2S_{12} & -l_1C_1 - l_2C_{12} \end{bmatrix} \begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix} \tag{3.2.17}$$

Let us next replace $\dot{x}'$ and $\dot{y}'$ by the expressions (3.2.15). The inverse relationships take then the form

$$\begin{aligned} \dot{\theta}_1 &= \frac{1}{l_1 l_2 S_2} \left(l_2 \, S_{12} \, \dot{x} \; + \; l_2 \, C_{12} \, \dot{y} \; + \; l_2 \, l_3 \, S_3 \, \dot{\phi}\right) \\ \dot{\theta}_2 &= \frac{1}{l_1 l_2 S_2} \left((l_1 \, S_1 \; + \; l_2 \, S_{12}) \, \dot{x} \; + \; (l_1 \, C_1 \; + \; l_2 \, C_{12}) \, \dot{y} \; + \; (l_1 \, l_3 \, S_{23} \; + \; l_2 \, l_3 \, S_3) \, \dot{\phi}\right) \\ \dot{\theta}_3 &= \dot{\phi} \; - \; \dot{\theta}_1 \; - \; \dot{\theta}_2 \end{aligned}$$

It is straight forward to deduce from them the inverse jacobian matrix

$$\boldsymbol{J}^{-1} = \frac{1}{l_1 l_2 S_2} \begin{bmatrix} l_2 \, S_{12} & l_2 \, C_{12} & l_2 \, l_3 \, S_3 \\ -(l_1 \, S_1 \; + \; l_2 \, S_{12}) & -(l_1 \, C_1 \; + \; l_2 \, C_{12}) & -(l_1 \, l_3 \, S_{23} \; + \; l_2 \, l_3 \, S_3) \\ l_1 \, C_1 & l_1 \, C_1 & l_1 \, l_2 \, S_2 \; + \; l_1 \, l_3 \, S_{23} \end{bmatrix} \tag{3.2.18}$$

It is need for control purpose, the kinematic model can also provide accelerations at joints in terms of accelerations in the operational space and joint velocities. Indeed a second derivation of equation (3.2.12) yields to the relationschip

$$\ddot{x}_i = \sum_{j=1}^{m} \frac{\partial f_i}{\partial \theta_j} \, \ddot{\theta}_j \; + \; \sum_{j=1}^{m} \sum_{k=1}^{m} \frac{\partial^2 f_i}{\partial \theta_j \, \partial \theta_k} \, \dot{\theta}_j \, \dot{\theta}_k \tag{3.2.19}$$

which can be rewritten in matrix form with the following definitions:

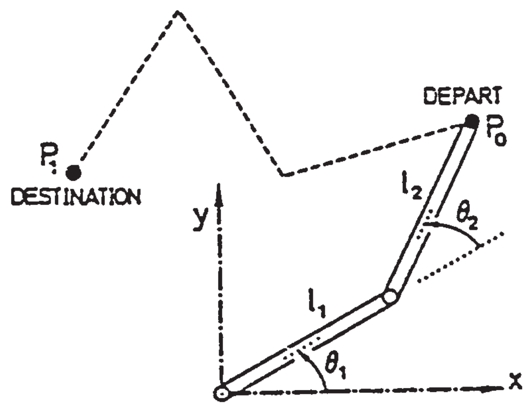


Figure 3.3.1: Trajectory description of the manipulator

- The matrix of squared velocities

$$\dot{\theta}^2 = \left[\dot{\theta}_1^2 \; (\dot{\theta}_1 \, \dot{\theta}_2) \; \dot{\theta}_2^2 \ldots \right] \tag{3.2.20}$$

- The matrix of associated coefficients

$$\boldsymbol{A} = \left[\frac{\partial^2 f_i}{\partial \theta_j \, \partial \theta_k} \right] \tag{3.2.21}$$

The resulting expression

$$\ddot{\theta} = \boldsymbol{J}^{-1} \left[\ddot{\boldsymbol{x}} - \boldsymbol{A} \, \dot{\theta}^2 \right] \tag{3.2.22}$$

shows that the accelerations at joints are also deduced in terms of the inverse jacobian matrix. In the case of the three-link manipulator, the matrices $\dot{\theta}^2$ and $\boldsymbol{A}$ have respectively the dimensions 6 and 3×6.

3.3 Trajectory planning

Suppose that the two-link manipulator is in the configuration shown as P_0 in figure 3.3.1; and suppose that it is required that it be moved to the destination configuration shown as P_1.

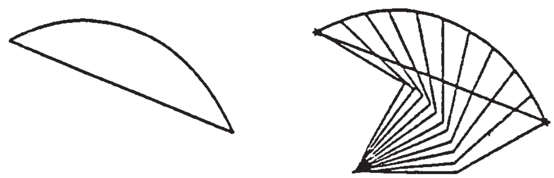
Trajectory planning converts a description of the desired motion

< MOVE ($P1$) >;

into a trajectory defining the time sequence of intermediate configurations of the arm between the origin P_0 and the destination P_1. The output of trajectory planning is a sequence

$$\{\theta_{\boldsymbol{k}}\} \qquad k = 1, \ldots, n$$

of configurations of the arm. The configurations, possibly together with their first and second time derivatives, are then shipped off in succession to the servo-mechanisms controlling the actuators that actually move the arm.



(a) (b)

Figure 3.3.2: Comparison of trajectory linear interpolations in joint space and in cartesian space

The actual trajectory between points P_0 and P_1 depends on the point of view adopted to describe it. Two descriptions of configurations can be adopted, namely configuration (or joint) space description and operational (cartesian) space description.

To illustrate the difference between both approaches let us consider the simple problem of generating a linear trajectory from P_0 to P_1 with the two-link manipulator of figure 3.3.1.

3.3.1 Joint space description

A linear interpolation between origin and destination provides the trajectory description

$$\theta(t) \;=\; (1-t)\,\theta(0) \;+\; t\,\theta(1) \qquad t \,\in\, [0,1] \tag{3.3.1}$$

Here, the t parameter describes simply the distance along the curve. It is thus normalized so that $t = 0$ at the origin configuration and $t = 1$ at the destination.

Figure 3.3.2(a) compares the resulting trajectory to the desired one and figure 3.3.2(b) displays a few intermediary configurations of the two-link arm.

It shows that interpolating in joint space is acceptable only as far as the path followed between P_0 and P_1 is completely free. This corresponds to point-to-point motion control mode which characterizes many of the industrial robots available today.

3.3.2 Cartesian space description

In the operational space, the linear interpolation

$$\boldsymbol{x}(t) \;=\; (1-t)\,\boldsymbol{x}(0) \;+\; t\,\boldsymbol{x}(1) \qquad t \,\in\, [0,1] \tag{3.3.2}$$

provides directly the desired trajectory. It can be achieved either through velocity control mode based on the kinematic differential model (3.2.13), in which case parameter t has meaning of the time, or using the concept of bounded deviation path.

The latter approach uses the fact the linear interpolation in joint space is very efficient to implement. It does not achieve straight line motion in cartesian space, but may depart from it by an acceptably small amount if the source and destination points correspond to nearby points in space. An efficient method for obtaining a bounded deviation path consists to introduce control mid points, or knot points, in a recursive manner.

The concept is illustrated by figure 3.3.3 which shows how a straight line trajectory can be refined progressively up to obtaining the result (c). The well known Taylor's algorithm for generation of trajectories is based on it.

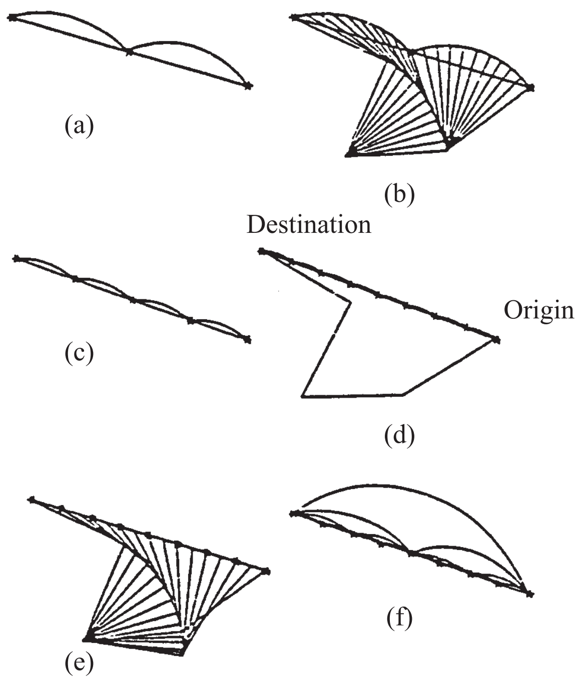


Figure 3.3.3: Generation of straight line motion in a recursive manner

- (a) First recursive call to the knot point generation algorithm
- (b) Intermediate configurations of case (a)
- (c) Knot points generated by a second recursive call
- (d) Knot points generated by a third recursive call
- (e) Intermediate configurations of case (d)
- (f) Comparison of the successive trajectories

3.3.3 Description of operational motion

The concept of cartesian path that we have just described does not necessarily include the concept of time course along it. Taking the time as a path parameter allows also for a control of velocity and acceleration along the trajectory: it is thus customary to distinguish between a path and a trajectory by defining the trajectory as the time course along the path.

Achieving velocity and acceleration control of the robot arm along the trajectory may be important for several reasons:

- Certain types of applications imply real time control of the position / orientation of the effector: arc welding, picking objects on a moving conveyor, etc.
- Trajectory specifications have to be compatible with the available acceleration power of the various joints;

- Time optimization of trajectories will become a subject of increasing importance.

Describing operational motion implies controlling the speed and the acceleration of the effector along its trajectory. In a particular, mechanical shocks have to be avoided by achieving continuous velocity and acceleration transition between consecutive trajectory segments. Appropriate time interpolation of trajectory segments is thus an important aspect of trajectory generation.

3.4 Dynamic model of a robot manipulator

3.4.1 The concept of dynamic model and its role

The dynamic model of the robot manipulator expresses the relationship between the various forces acting on the mechanical structure and the resulting displacements, velocities and accelerations. The forces implied in robot motion may have several origins:

- the torques delivered by the motors,
- the inertia of members,
- the gravity,
- the loss of energy (damping, friction . . .),
- the interaction with the undergoing task.

Just as in the kinematic analysis, one may distinguish between the direct dynamic model and the inverse dynamic model.

Given an initial state of the mechanical structure (i.e. displacements and velocities at joints at time $t = 0$) and the time history of torques $\boldsymbol{c}(t)$ acting at joints, the direct dynamic model allows to predict the resulting motion $\theta(t)$. When associated with the direct kinematic model, it yields to a prediction of the trajectory $\boldsymbol{x}(t)$.

The mechanical form of the dynamic model is a nonlinear system of second order differential equations to be integrated in time.

Its primary purpose is to obtain a computer simulation of robot dynamic behavior. Associated with a model of servo-systems,it can also provide an evaluation of the characteristics of the overall response of the manipulator and its control system (Figure 3.4.1).

The direct dynamic model can also be used for control purpose (for example, in a control mode based on a reference model) in which case a simplified version of it is generally sufficient. Despite the simplifications that are brought to the model, the main limitation stems from the need to integrate the model in real time. Therefore, the use of a direct dynamic model aiming at the synthesis of an appropriate control law is still largely a research subject.

The inverse dynamic model allows to predict the torques needed to reach or maintain a specified geometric configuration. It implies a computer effort which is of an order of magnitude less than the direct model plays therefore a more immediate role in robot design and control. it can be used for two purposes:

- At the design level, in order to evaluate the required mechanical characteristics of actuators and transmission devices and to predict the dynamic behavior of the system;

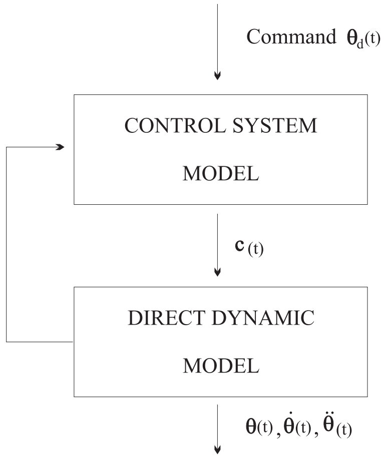


Figure 3.4.1: Computer simulation of robot dynamic behavior using a direct dynamic model

- At the control level, to predict the torques delivered by the motors in view of the synthesis of a dynamic control law.

The principle of a dynamic control law which makes use of an inverse dynamic model is outlined in Fig. 3.4.2. Its main advantage by comparison with classical control lies in its ability to provide adaptation to changes of inertia resulting from variable geometric configuration.

A dynamic model of either direct or inverse type can be constructed according to two techniques which have their respective advantages and drawbacks.

- *The Euler-Newton formalism* proceeds by splitting the system into individual components and writing vector relationships which express dynamic equilibrium of the individual parts. It is well suited to a recursive computational procedure which leads to a minimum of arithmetic operations.
- *The Lagrangian approach*, which is based on the well known Lagrange equations of the classical mechanics, presents mainly the advantage of being systematic and of simple application; it is based on the evaluation of energy quantities such as kinetic energy due to gravity and virtual work associated to applied torques and external loads. It may lead also to recursive computation procedures, but in a not so evident manner than Euler-Newton's formalism.

The following introduction to the concept of dynamic model will be based on the Langrangian approach.

3.4.2 Dynamic model of a two-link manipulator

Let us consider the 2 DOF manipulator of fig. 3.4.3. The masses m_1 and m_2 of members are supposed concentrated at joints in order to simplify the model to a maximum.

The Langragian formalism of classical mechanics stipulates that the kinetic energy $\mathcal{U}$ of a system with n degrees of freedom $q_i, (i = 1, \dots, n)$ can be expressed in the forms

$$\mathcal{T} = \mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) \qquad \text{and} \qquad \mathcal{U} = \mathcal{U}(\boldsymbol{q}, t)$$

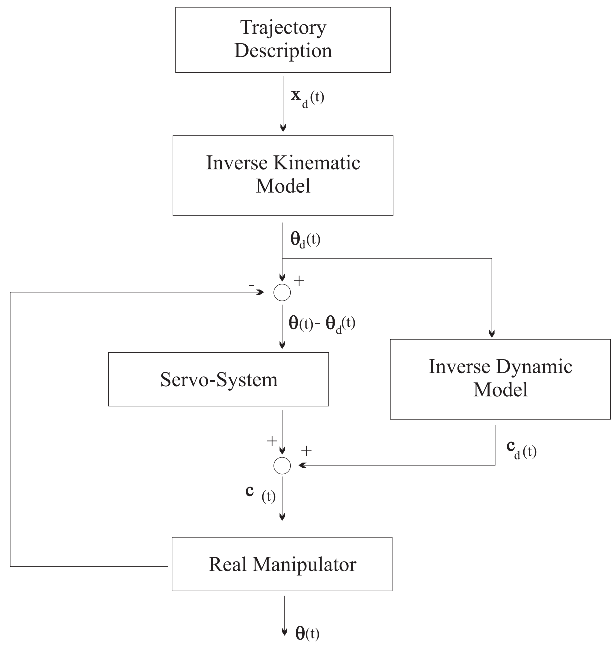


Figure 3.4.2: The principle of dynamic control

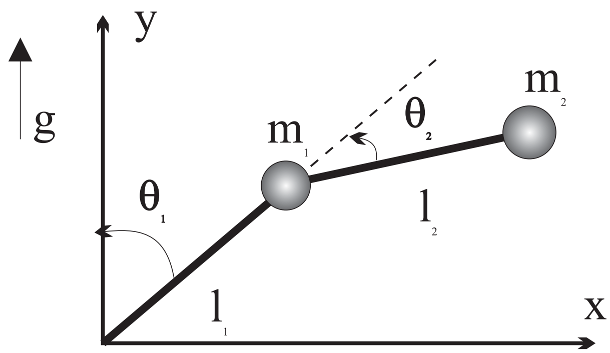


Figure 3.4.3: Model of a two-link manipulator

The dynamic behavior of the system is then governed by the dynamic equilibrium equations written in Lagrange's form

$$Q_i(t) \;=\; \frac{d}{dt}\left(\frac{\partial \mathcal{T}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{T}}{\partial q_i} + \frac{\partial \mathcal{U}}{\partial q_i} \tag{3.4.1}$$

where the Q_i denote the generalized external forces conjugated to the generalized displacements q_i.

Let us decompose the kinetic and potential energies of the system in a sum of member contributions

$$\mathcal{T} \;=\; \mathcal{T}_1 \;+\; \mathcal{T}_2 \qquad \text{and} \qquad \mathcal{U} \;=\; \mathcal{U}_1 \;+\; \mathcal{U}_2 \tag{3.4.2}$$

The kinetic energies of the masses m_1 and m_2 are simply given by

$$\mathcal{T}_1 \;=\; \frac{1}{2} m_1 v_1^2 \qquad \text{and} \qquad \mathcal{T}_2 \;=\; \frac{1}{2} m_2 v_2^2 \tag{3.4.3}$$

where v_1 and v_2 are the absolute velocities of the corresponding points. They are calculated in terms of the generalized degrees of freedom θ_1 and θ_2 by

$$\begin{aligned} v_1^2 &= \dot{x}_1^2 \;+\; \dot{y}_1^2 \\ &= l_1^2 \dot{\theta}_1^2 \\ v_2^2 &= \dot{x}_2^2 \;+\; \dot{y}_2^2 \\ &= l_1^2 \dot{\theta}_1^2 \;+\; l_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \;+\; 2 l_1 l_2 (\dot{\theta}_1 + \dot{\theta}_2) \cos\theta_2 \dot{\theta}_1 \end{aligned} \tag{3.4.4}$$

The potential energies of the masses m_1 and m_2 under the action of gravity are similar given by

$$\begin{aligned} \mathcal{U}_1 &= m_1 g y_1 \\ &= m_1 g l_1 (1 - \cos\theta_1) \\ \mathcal{U}_2 &= m_2 g y_2 \\ &= m_2 g l_1 (1 - \cos\theta_1) \;+\; m_2 g l_2 \big(1 - \cos(\theta_1 + \theta_2)\big) \end{aligned} \tag{3.4.5}$$

The application of the Lagrange equations (3.4.1) provides the algebraic expressions for the generalized forces conjugated to joint displacements θ_1 and θ_2

$$\begin{aligned} c_1(t) &= m_{11}\ddot{\theta}_1 + m_{12}\ddot{\theta}_2 \;+\; b_{111}\dot{\theta}_1^2 \;+\; b_{122}\dot{\theta}_2^2 \;+\; b_{112}\dot{\theta}_1\dot{\theta}_2 + g_1 \\ c_2(t) &= m_{12}\ddot{\theta}_1 + m_{22}\ddot{\theta}_2 \;+\; b_{211}\dot{\theta}_1^2 \;+\; b_{222}\dot{\theta}_2^2 \;+\; b_{221}\dot{\theta}_1\dot{\theta}_2 + g_2 \end{aligned} \tag{3.4.6}$$

The meaning of the different inertia coefficients of the model is the following

- the m_{ii} are the principal inertia coefficients. Their explicit form is

$$\begin{aligned} m_{11} &= m_1 l_1^2 + m_2 (l_1^2 + l_2^2 + 2 l_1 l_2 \cos\theta_2) \\ m_{22} &= m_2 l_2^2 \end{aligned}$$

- the symmetrical term m_{12} and m_{21} are the inertia coupling terms

$$m_{12} \;=\; m_{21} \;=\; m_2 l_2^2 \;+\; m_2 l_1 l_2 \cos\theta_2$$

- the coefficients give rise to centrifugal and Coriolis forces

$$\begin{aligned} b_{111} &= 0 & b_{122} &= -m_2 l_1 l_2 \sin\theta_2 \\ b_{211} &= m_2 l_1 l_2 \sin\theta_2 & b_{222} &= 0 \\ b_{112} &= -2 m_2 l_1 l_2 \sin\theta_2 & b_{221} &= 0 \end{aligned}$$

It can be observed that most of these coefficients are strongly dependent on the instantaneous configuration.

- The terms g_1 and g_2 denote the torques produced by the gravity. One finds explicitly

$$\begin{aligned} g_1 &= (m_1 + m_2) g l_1 \sin\theta_1 \ + \ m_2 g l_2 \sin(\theta_1 + \theta_2) \\ g_2 &= m_2 g l_2 \sin(\theta_1 + \theta_2) \end{aligned}$$

3.5 Dynamic model in the general case

In the general case, the dynamic model of a n-degree of freedom with open chain structure takes a form analogous to (3.4.6)

$$\boldsymbol{M}\,\ddot{\theta} \ + \ \boldsymbol{B}\,\dot{\theta}^2 \ + \ \boldsymbol{g} \ = \ \boldsymbol{c} \tag{3.5.1}$$

where

- $\boldsymbol{M}$ is the inertia matrix with dimension $n \times n$
- $\boldsymbol{B}$ is the matrix of centrifugal and Coriolis coefficients. It has dimension $m \times n$ with $m = n(n+1)/2$
- $\boldsymbol{g}$ is the matrix of gravity terms, with dimension n
- $\boldsymbol{c}(t)$ is the matrix of applied torques, also with dimension n.

All the coefficients of the matrices $\boldsymbol{M}$, $\boldsymbol{B}$ and $\boldsymbol{g}$ are obviously very complex functions of the instantaneous configuration θ.

3.6 Dynamic control according to linear control theory

Our objective is to discuss how to cause a manipulator to actually perform a desired motion.

If one excepts the case of manipulators driven by either stepper or pneumatic motors which can be controlled in an open loop fashion, one may consider that manipulators are powered by actuators which provide a torque or a force at each joint. Some kind of control system is then needed to generate appropriate actuator commands which will realize the prescribed motion.

A manipulator can be regarded as a mechanism with an actuator at each joint to apply a torque between two neighboring links, and instrumented with position (and possibly velocity) sensors to measure the joint angular displacements and velocities.

In order to cause the desired motion of each joint to be followed by the manipulators, we must specify a *control algorithm* which sends torque commands to actuators. Almost always these torques are compared using feedback from the joint sensors.

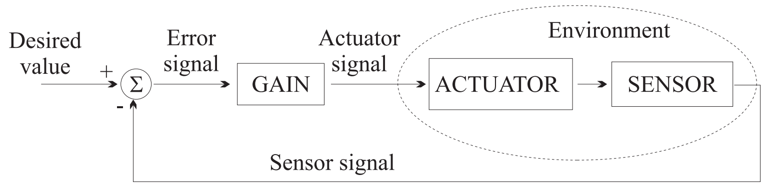


Figure 3.6.1: A typical control system

The control problem for manipulators is inherently non-linear. This means that much of the linear control theory described in this section for the synthesis of control algorithms is not directly applicable.

Nevertheless, in most of the proposed solutions to the non-linear problems, one ends up using methods from linear control theory. This material will be quickly reviewed in the section 6 and applied to multi-variable systems in section 7. The content of the these two sections is essentially based on ref. 5.

3.6.1 The objective of control theory

The objective control theory is to analyze and synthetize like the one illustrated in figure 3.6.1. A typical system consists of

- an actuator which is capable of changing the environment in some way. For the case of a robot joint, the actuator will be the electrical or hydraulic motor driving the joint;
- a sensor of some kind which can measure some aspects of the actuator's effect on the environment. In our case, the sensor(s) will measure position and/ or velocity.

Feedback is used as follows. The actual value measured by the sensor is compared with (i.e subtracted from) the desired value to form an error value. This error value is converted to an actuator control signal through multiplication with a positive gain. In this way, the actuator is commanded such that it always tends to reduce the current value of the error.

3.6.2 Open-loop equations for motion of a physical system

Let us consider the control of a very simple mechanical system such as the system figure 3.6.2. The control variable, x, is the position of the mass. The actuator (not shown) is capable of a force f to drive the mass which experiences also a spring force $-k\,x$ and a viscous damping force $-c\,\dot{x}$. The constants k and c are respectively the spring (stiffness) and damping constants.

The equation describing the motion of the system itself is obtained by summing all the forces acting on the body

$$m\,\ddot{x} \;+\; c\,\dot{x} + k\,x \;=\; f \tag{3.6.1}$$

It is the open-loop equation of motion. Because the equation (3.6.1) is of second order in time, we refer to the mechanical system of figure 6.2 as a second order system.

3.6.3 Closed-loop equation of motion

Let us next assume that we have position and velocity sensors which measure x and $\dot{x}$ for our mechanical system. We could then use this information to compute a value of f to apply to the

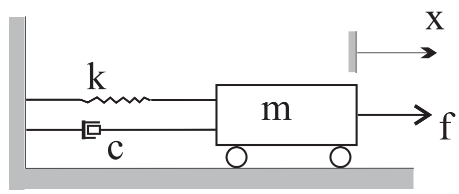


Figure 3.6.2: Mass-spring-dash pot system

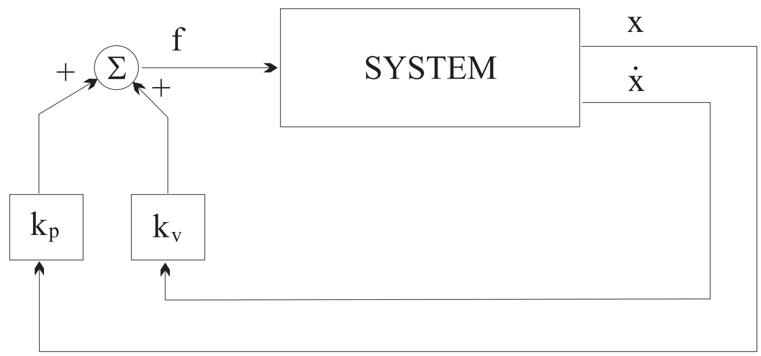


Figure 3.6.3: Block diagram of a position regulating controller

system through the actuator. For example, we could compute the control law

$$f \;=\; -\,k_p x - k_v \dot{x} \tag{3.6.2}$$

In order to analyze how the system would behave when controlled with this law, we write the *closed-loop equation of motion* by combining (3.6.1) and (3.6.2). This leads to

$$m\ddot{x} \;+\; (c + k_v)\dot{x} + (k + k_p)x \;=\; 0 \tag{3.6.3}$$

By comparison with the open-loop equation of motion, the closed loop equation shows us that the system now acts as though the spring has the stiffness value $(k + k_p)$ and as though the damping has the value $(c + k_v)$. Hence *we have used the control effort to change the apparent stiffness and damping of the original system.*

This is perhaps the most basic application of control theory to alter the characteristics of the system under control in order that the system behaves in some desired way.

To build a position-controlling system we would choose a large value for k_p so that the mass would act as if an extremely stiff spring was holding it in position. Such a control system would try to maintain the nominal position of the mass despite various *disturbance forces*. This type of control system is called a *regulator* because it attempts to regulate the value of the control variable to some constant value.

Figure 3.6.3 shows a block of diagram of the control system specified by the control law (3.6.3). The mechanical system is shown only as a black box whose input is the force which we can command with the actuator,and with outputs x and $\dot{x}$ which can be read with the sensors.

3.6.4 Stability, damping and natural frequency of the closed-loop system

Given a control law such as (3.6.2) and the resulting closed loop equation such as (3.6.4), the performance of a system is usually discussed in terms of two central features of the control system, namely

- its ability to respond to changes by the desired output value;
- its ability to suppress disturbances.

As far as performance is concerned, the first requirement of a control system is its stability. A possible definition of stability is the following:

a system is stable if, given an input function (or disturbance function), the output remains bounded.

In order to determine stability, along with other performance measures, we must solve the closed-loop equation. For example, to solve the second-order differential equation with constant coefficients (3.6.3), let us assume a general solution of the form

$$x = ae^{st}$$

The corresponding characteristic equation is

$$ms^2 + (c + k_v)s + (k + k_p) = 0 \tag{3.6.4}$$

It has two roots, s_1 and s_2, which are given by

$$s_{1,2} = -\frac{c + kv}{2m} \pm \frac{\sqrt{(c + k_v)^2 - 4m(k + k_p)}}{2m} \tag{3.6.5}$$

and their nature depends upon the values of the gain k_v and k_p which are introduced in the system. The roots are complex

$$s = -\alpha \pm i\beta \tag{3.6.6}$$

provided that the gains are such that

$$(c + k_v)^2 < 4m(k + k_p) \tag{3.6.7}$$

in which case the solution to (3.6.3) may be conveniently written in the form

$$x = ae^{-\alpha t}\sin(\omega_n\sqrt{1 - \tau^2 t} + \phi) \tag{3.6.8}$$

where

$$\alpha = \frac{c + k_v}{2m}$$

is the *time decay coefficient*, which remains positive as long as $c + k_v > 0$;

$$\omega_n = \frac{\sqrt{k + k_p}}{m} \tag{3.6.9}$$

is the *natural frequency* of the system. It remains real as long the effective stiffness $k + k_p$ is positive, in which case it is related to the magnitude of the roots of the system by

$$\omega_n = \sqrt{\alpha^2 + \beta^2} \tag{3.6.10}$$

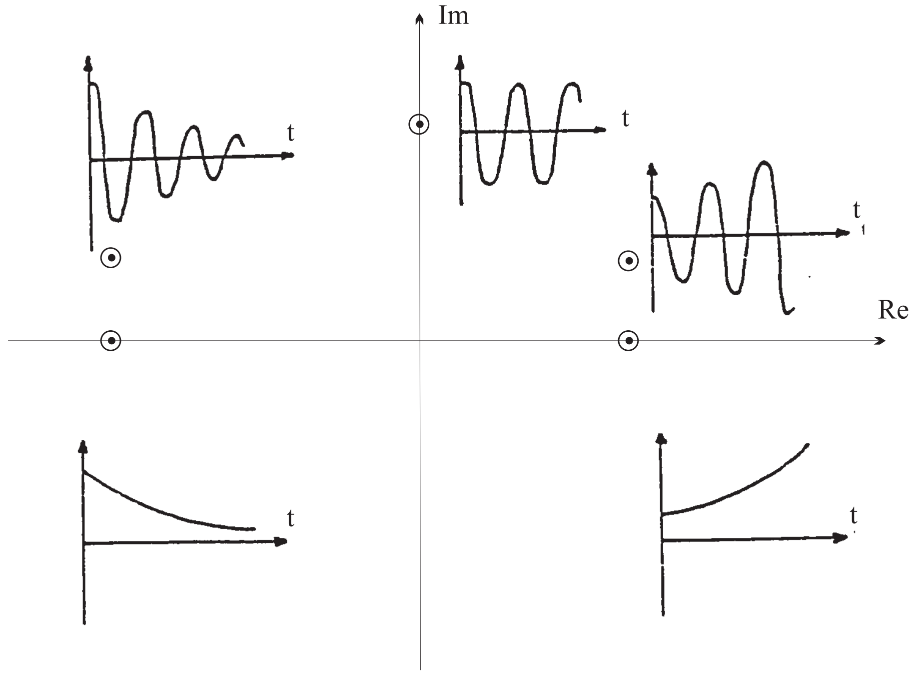


Figure 3.6.4: Response of the 1 DOF system as function of pole location

is the *damping ratio* of the system, and is calculated by

$$\tau = \frac{c + k_v}{2\sqrt{m(k + k_p)}} = \frac{\alpha}{\sqrt{\alpha^2 + \beta^2}} \tag{3.6.11}$$

Different behaviours can be observed depending upon the relative damping of the system.

- If $\tau = 0$, the system has no damping and it will never stop moving once disturbed
- If $0 < \tau < 1$, the system is lightly damped and it returns to its equilibrium position according to an oscillatory motion given by equation.(3.6.8).
- If $\tau > 1$, the system is over-damped and it does not oscillate at all. The motion is of exponential type

  $$x = Ae^{s_1 t} + Be^{s_2 t}$$

  with roots calculated by (3.6.5).
- If $\tau = 1$, the system is critically damped and possesses two equal and real roots

  $$s_{1,2} = -\alpha$$

  Critical damping allows the fastest return to the nominal position without being oscillary.

Figure 3.6.4 shows the behavior of the system according to the location of the roots of the characteristic equation on the real-imaginary plane.
It indicates qualitatively what the shape ,of the response to imposed initial conditions would be. Notice that if the real part of the roof is positive, the response is unstable. If the roots

are complex, the response is oscillatory and if they are real, the response is the sum of two exponentials. For simplicity only one of the roots is displayed in the figure, the second one being its complex conjugate.

3.6.5 Position control

Our control system developed so far has not input; it is simply a regulating system.

Consider again the problem of controlling the system of figure 3.6.2. To add a desired position input, let us use the control law

$$f \;=\; k_p(x_d - x) - k_v\dot{x}$$

where x_d is the target position of the mass. The resulting closed-loop equation is

$$m\ddot{x} + (c + k_v)\dot{x} + (k + k_p)x \;=\; k_p x_d \tag{3.6.12}$$

If the target position x_d is constant with time, position error

$$e \;=\; x_d - x$$

is such that $\dot{e} \;=\; -\dot{x}$ and $\ddot{e} \;=\; -\ddot{x}$. It is thus governed by a similar equation

$$(k + k_p)e \;+\; (c + k_v)\dot{e} \;+\; m\ddot{e} \;=\; kx_d \tag{3.6.13}$$

Assuming that x_d is imposed in a step fashion, the system will asymptotically move its equilibrium position to the steady-state response of (3.6.13)

$$x \;=\; \frac{k_p}{k + k_p}x_d \tag{3.6.14}$$

to which corresponds a steady-state error

$$x \;=\; \frac{k}{k + k_p}x_d \qquad \text{such that} \qquad \lim_{k_p \to \infty} e \;= 0 \tag{3.6.15}$$

3.6.6 Integral correction

A modified control law of type

$$f \;=\; k_p(x_d - x) \;-\; k_v\dot{x} \;+\; k_I\int_0^t (x_d - x)\;dt \tag{3.6.16}$$

includes proportional integral and derivative correction and is called PID controlled law. It will guarantee that any steady state error would cause the integral term to build up until it generates a sufficient force to cause the system to move such as to reduce the steady state error. Time derivation of the corresponding closed-loop equation yields to

$$m\dddot{e} \;+\; (c + k_v)\ddot{e} \;+\; (k + k_p)\dot{e} \;+\; k_I e \;=\; 0 \tag{3.6.17}$$

The performed of the PID controller is thus governed by a third order system. The roots of its characteristic equation may be adjusted so as to match those of the equation

$$(s^2 \;+\; 2\tau\omega_n s + \omega_n^2)(s + \gamma) \;=\; 0 \tag{3.6.18}$$

where γ is spurious root. The gain k_v, k_p and k_I are then related to τ and γ by

$$\begin{aligned} k_I &= \omega_n^2\gamma m \\ k_v &= m(2\tau\omega_n + \gamma) - c \\ k_p &= (\omega_n^2 + 2\tau\gamma)m - k \end{aligned} \tag{3.6.19}$$

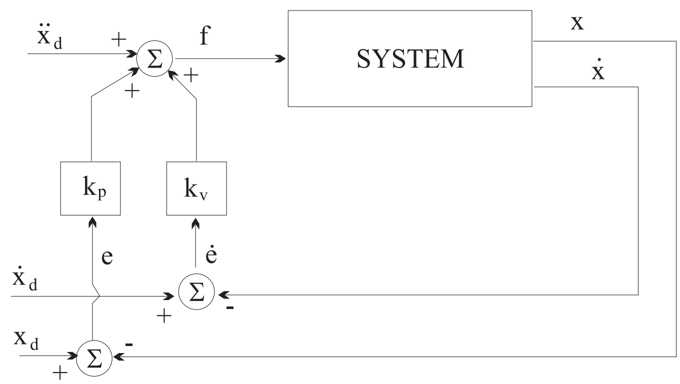


Figure 3.6.5: Control system with complete trajectory input

3.6.7 Trajectory following

Trajectory means, as previously stated, the time history of the position variable and the corresponding velocities and accelerations.
While the control law (3.6.12) was sufficient for a system commanded with step type inputs, for a general position trajectory it is useful to build a control system which has 3 inputs: desired position, velocity and acceleration. If these inputs are available from the trajectory planning, a reasonable choice of the control laws is

$$f \;=\; m\ddot{x}_d \;+\; k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x) \tag{3.6.20}$$

to which an integral correction may eventually be added.

To make this choice clear, let us consider the extremely simple case of a *unit mass* with no elastic restoring force, no damping and driven by the control law (3.6.20). The open-loop equation is simply

$$\ddot{x} \;= f \tag{3.6.21}$$

combining (3.6.20) with the control law (3.6.22), we have

$$\ddot{x} \;=\; \ddot{x}_d + k_v\dot{e} + k_p e \tag{3.6.22}$$

which may then be written as

$$\ddot{e} + k_v\dot{e} + k_p e \;=\; 0 \tag{3.6.23}$$

This form is particularly nice because the *error dynamics* is chosen directly by gain selection. A good choice is to choose gains for critical damping, in which case errors are suppressed in a critically damped fashion. Figure 3.6.5 shows the general structure of such a trajectory following controller. Many times, an integral correction is added to the servo law as described in the previous section.

3.6.8 Control law partitioning

In preparation for designing control laws for more complicated mechanical systems, let us consider again the mass-spring-damper system of figure 6.2 and construct its control law within a certain structure which will be helpful in understanding the control of more complicated systems. We wish decompose the controller of the system into two parts:

- The first part of the control law is *model-based* in that it makes use of the parameters of the particular system under control. It is set up such *it reduces the system so that it appears to be a unit mass*;

- The second part of the control law is *error driven* in that it forms error signals by differenciating desired and actual variables and multiplies these error by gains. The error driven part of the control law is also the *servo portion.*

Since the model based portion of the control law has the effect of making the system to be a unit mass, the design of the servo portion is very simple. Gains are chosen as if we were controlling a unit mass system.

The partitioned control law is thus assumed of the form

$$f = \alpha f^{'} + \beta \tag{3.6.24}$$

where α and β are functions or constants chosen so that when $f^{'}$ is the new input of the system, the system appears to be a unit mass. With this structure of control the system equation is

$$m\,\ddot{x} + c\,\dot{x} + k\,x = \alpha f^{'} + \beta \tag{3.6.25}$$

Clearly, in order to make the system appear as a *unit mass* $f^{'}$ input we should choose α and β as

$$\begin{aligned} \alpha &= m \\ \beta &= c\dot{x} + kx \end{aligned} \tag{3.6.26}$$

in which case the system equation becomes

$$\ddot{x} = f^{'} \tag{3.6.27}$$

The servo portion of the control law is computed next in order to answer to the objective of trajectory following. From equation (3.6.20) we obtain

$$f^{'} = \ddot{x}_d + k_v\dot{e} + k_p e \tag{3.6.28}$$

In this case, k_p and k_v are computed for a system given by equation (3.6.27) and are particularly easy to compute.

Combining next equations (3.6.25) and (3.6.28) we can write the closed loop equation for the system. Making use of the model based part (3.6.25) reduces it to the equation governing the error of the system

$$\ddot{e} + k_v\dot{e} + k_p e = 0 \tag{3.6.29}$$

Its frequency behavior is governed by the parameters

$$\omega_n = \sqrt{k_p} \qquad \frac{k_p}{2\sqrt{k_p}} \tag{3.6.30}$$

and it is critically damped when $k_v^2 = 4k_p$. Figure 6.6 shows the general form of the partitioned servo law with trajectory following inputs.

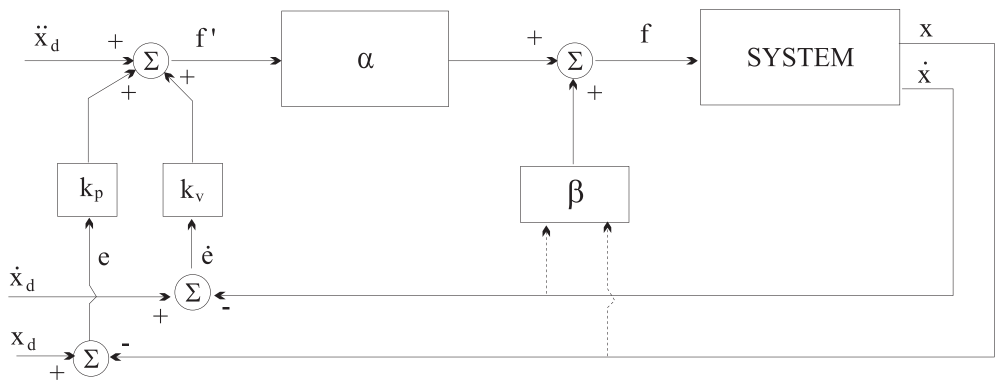


Figure 3.6.6: General form of control law

3.7 Motion Control of Non-Linear and Time-Varying Systems

In the preceding section the mathematical form of the control law arose from the fact that the system under consideration could be modeled as a linear constant coefficient differential equation. For systems whose parameters vary in time or systems which are by nature non linear, efficient solutions are however more difficult to obtain.

In a very broad sense, mainly two approaches may be followed to control systems involving non-linearities:

- Linearization may be used, when non-linearities are not too severe, to derive linear models which are approximations of the nonlinear equations in the neighborhood of an operating point;
- Adaptation of the control law consists to change with time the coefficients of the control law according to variations of systems parameters.

Linearization does not lead to an efficient solution for controlling robot manipulators since they have been shown to have a highly motion dependent dynamics. We will thus concentrate in what follows on an adaptive technique in which a nonlinear control law is applied to the system in such way that it performs linearly.

3.7.1 Design of nonlinear control laws

Various adaptative techniques can be proposed, but the best suited to robot motion control consists probably to construct a linearizing control law as suggested by the control law partitioning method described in the former section. The model based part of the control law is then motion dependent and calculated such that the servo-error of the system performs in the same manner as if the system was linear. Let us consider the case of controlling the more general one-DOF system

$$m\,\ddot{x} + g(x,\dot{x}) \;=\; f \tag{3.7.1}$$

where the function $g(x,\dot{x})$ presents general motion-dependent forces such as

- Elastic and inelastic (plastic, visco-plastic . . .) restoring forces;

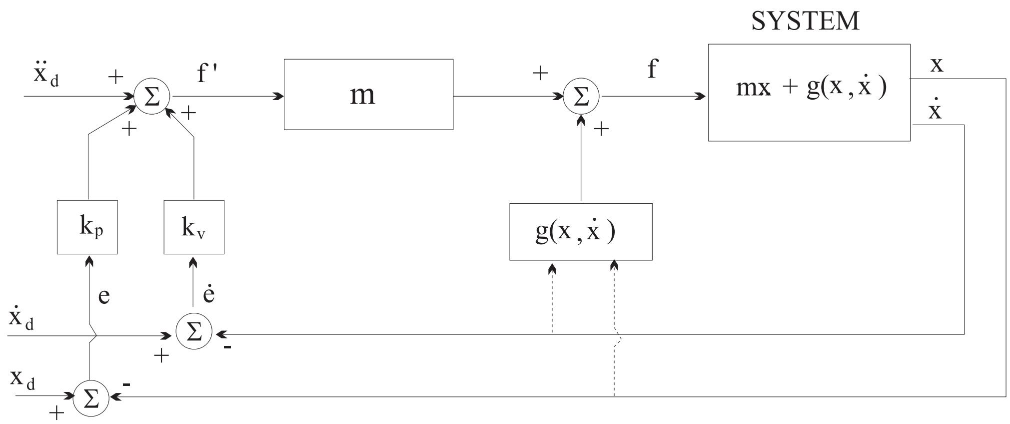


Figure 3.7.1: Nonlinear control system with error performance of linear system

- Linear or nonlinear (dry friction ...) damping forces,
- Gyroscopic forces.

The model based portion of the control law is $f \;=\; \alpha f^{'} + \beta$ where

$$\begin{aligned} \alpha &= m \\ \beta &= g(x, \dot{x}) \end{aligned} \tag{3.7.2}$$

and the servo portion is as always

$$f^{'} \;=\; x_d \;+\; k_v \dot{e} + k_p e \tag{3.7.3}$$

where the values of the gains are calculated from some desired performance specification. Figure 3.7.1 shows a block diagram of this control system.
While the field of nonlinear control theory is quite difficult in general, we see that in specific cases as covered by equation (3.7.1) it is non prohibitively difficult to design a nonlinear control. It can be summarized as follows:

Compute a non-linear model based control law which 'cancels' the non-linearities of the system to be controlled. The system is then reduced to a linear system which can be controlled using a simple linear servo law.

The linearizing control law can be constructed provided that an inverse dynamic model of the system is available. The non-linearies in the system cancels with those of the inverse model, leaving a linear closed loop system. Obviously, to do this cancelling, we must know the parameters and the structure of the non-linear system. This may be the most difficult part when dealing with real nonlinear systems.

3.8 Multi-Variable Control Systems

Controlling a robot manipulator is a multi-input/ multi-output problem. A desired trajectory of the effector can be translated in terms of desired joint positions, velocities and accelerations. The control law must compute a vector of joint actuator signals from the knowledge of the actual configuration and the desired trajectory. Our basic scheme of partitioning the control into a model-based part and a servo part is still applicable, but will appear in the matrix form.

The control law is now the form

$$\boldsymbol{f} \;=\; \boldsymbol{A}\boldsymbol{f}^{'} \;+\; \beta \tag{3.8.1}$$

where, for a n-degree of freedom system,

- $\boldsymbol{f}$, $\boldsymbol{f}^{'}$ are n-dimensional vectors
- $\boldsymbol{A}$ is a nxn matrix.

Note that the matrix $\boldsymbol{A}$ is not necessarily diagonal, but is rather chosen to decouple the n equations of motion. If $\boldsymbol{A}$ and β are correctly chosen, then from the $\boldsymbol{f}^{'}$ input the system appears to be n independent unit masses. For this reason, in the multi-dimensional case the model-based portion of the control law is called the *linearizing* and *decoupling* law. The servo law for a multi-dimensional system becomes

$$\boldsymbol{f}^{'} \;=\; \ddot{\boldsymbol{x}}_d \;+\; \boldsymbol{K}_v\dot{\boldsymbol{e}} \;+\; \boldsymbol{K}_p\boldsymbol{e} \tag{3.8.2}$$

where $\boldsymbol{K}_v$ and $\boldsymbol{K}_p$ are now nxn gain matrices. They are generally chosen to be diagonal with constant gains on the diagonal.

3.9 Multi-variable Problem Manipulators

In the case of manipulator control, we have shown that the equations of motion take the form

$$\boldsymbol{C}(t) \;=\; \boldsymbol{M}(\theta)\,\ddot{\theta} \;+\; \boldsymbol{B}(\dot{\theta})\,\dot{\theta}^2 + \boldsymbol{g}(\theta) \tag{3.9.1}$$

where $\boldsymbol{M}$ is the inertia matrix, $\boldsymbol{B}$ the matrix of Coriolis and centrifugal coefficients and $\boldsymbol{g}$ is the vector of gravity coefficients. All their elements are complex functions of θ, the n-dimensional vector of joint variables. Additionally, we may incorporate a model of friction or other non-rigid body effects in the form

$$\boldsymbol{C}(t) \;=\; \boldsymbol{M}(\theta)\,\ddot{\theta} \;+\; \boldsymbol{B}(\dot{\theta})\,\dot{\theta}^2 + \boldsymbol{g}(\theta) \;+\; \boldsymbol{f}(\theta,\dot{\theta}) \tag{3.9.2}$$

The problem of controlling a system like (3.9.2) can be handled by the partitioned controller scheme that we have introduced throughout this chapter. In this case we have

$$\boldsymbol{c}(t) \;=\; \boldsymbol{A}\,\boldsymbol{c}' \;+\; \beta \tag{3.9.3}$$

and we choose

$$\begin{aligned} \boldsymbol{A} \;&=\; \boldsymbol{M} \\ \beta \;&=\; \boldsymbol{B}(\dot{\theta})\,\dot{\theta}^2 \;+\; \boldsymbol{g}(\theta) \;+\; \boldsymbol{f}(\theta,\dot{\theta}) \end{aligned} \tag{3.9.4}$$

with servo law

$$\boldsymbol{c}' \;=\; \ddot{\theta}_d \;+\; \boldsymbol{K}_v\,\dot{\boldsymbol{e}} + \boldsymbol{K}_p\,\boldsymbol{e} \tag{3.9.5}$$

where the servo error is calculated by

$$\boldsymbol{e} \;=\; \theta_d - \theta \tag{3.9.6}$$

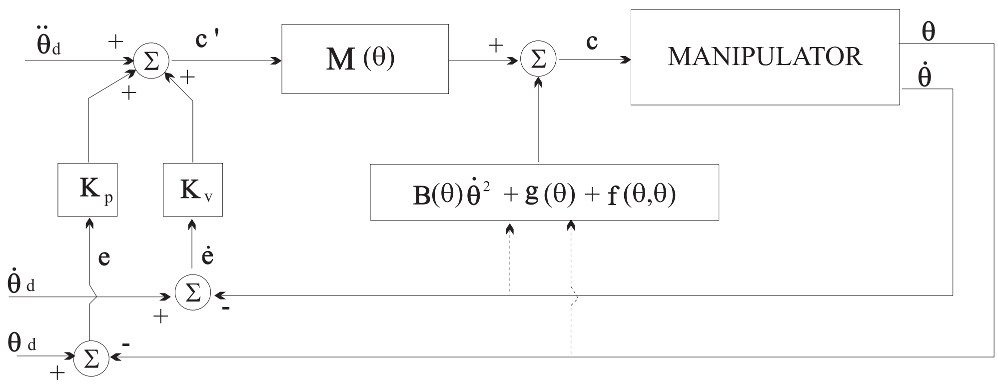


Figure 3.9.1: Dynamic control of a manipulator system

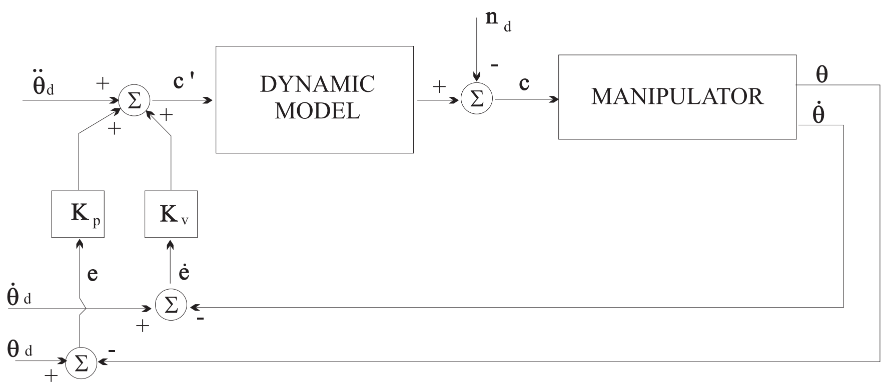


Figure 3.9.2: Manipulator control system with external disturbance included

The resulting control scheme is represented by the figure 3.9.1. The entire inverse dynamic model appears in one box to indicate that the decoupling and linearizing procedure remains the same regardless of how the inverse dynamic model is computed
If we have good knowledge of all the parameters of the manipulator then the control scheme of figure 3.9.1 will provide a performance of the system with the chosen gains.

However, the assumptions that we have made to develop controllers for robot manipulator are rarely available for decoupling and linearizing the control algorithm as described.

One way to analyze the effect of error in the model of the system is to indicate a vector of disturbance torques acting the joints. In figure 3.9.2 we have indicated these disturbances as an input: they arise from any un-modelled effect in the dynamic equations, and they include also the effects of resonance and "noise". Writing the system error equation with the inclusion of unknown disturbances

$$\ddot{\boldsymbol{e}} \;+\; \boldsymbol{K}_v\, \dot{\boldsymbol{e}} \;+\; \boldsymbol{K}_p\, \boldsymbol{e} \;=\; \boldsymbol{M}^{-1}(\theta)\, \boldsymbol{n}_d \tag{3.9.7}$$

where $\boldsymbol{n}_d$ is the vector of disturbance torques at joints. The left-hand side of (3.9.7) is uncoupled, but from the right-hand side we see that a disturbance of any particular joint will introduce errors at all other joints since $\boldsymbol{M}$ is not diagonal.

3.10 Practical Considerations

The assumptions that we have made to develop controllers for robot manipulators are rarely true in practice. The exact models are rarely available for decoupling and linearizing the control algorithm as described. It is thus important to discuss the practical problems faced by the control engineer in the design of control systems and describe the solutions which are adopted industrially to overcome them.

3.10.1 Lack of knowledge of parameters

Even if a model is available, the parameters of the model are often not known accurately:

- Friction effects are extremely difficult to predict with sufficient accuracy;
- The mass properties of the system and the interaction forces with the external world are changing during the task, so that the maintenance of an accurate dynamic model is generally impossible.

Since the model of equation (3.9.2) will never be perfect, let us distinguish between the model of the system and its actual properties. Let us note by an asterisk our model of the manipulator. Then, perfect knowledge of the model would mean

$$\begin{aligned} \boldsymbol{M}^\star(\theta) &= \boldsymbol{M}(\theta) \\ \boldsymbol{B}^\star(\theta,\dot\theta) &= \boldsymbol{B}(\theta,\dot\theta) \\ \boldsymbol{g}^\star(\theta) &= \boldsymbol{g}(\theta) \\ \boldsymbol{f}^\star(\theta,\dot\theta) &= \boldsymbol{f}(\theta,\dot\theta) \end{aligned} \tag{3.10.1}$$

so that, although the manipulator dynamics is given by equation (3.9.1), our control law computes with

$$\begin{aligned} \boldsymbol{c} &= \boldsymbol{A}\boldsymbol{c}' + \beta \\ \boldsymbol{A} &= \boldsymbol{M}^\star \\ \beta &= \boldsymbol{B}^\star(\theta,\dot\theta) + \boldsymbol{g}^\star(\theta) + \boldsymbol{f}^\star(\theta,\dot\theta) \end{aligned} \tag{3.10.2}$$

so that decoupling and linearization are not accomplished perfectly. By writing the closed loop equation one observes that the lack of knowledge of parameters is responsible for a noise input

$$\boldsymbol{n}_d = \boldsymbol{M}^{\star -1}\left[(\boldsymbol{M}-\boldsymbol{M}^\star)\,\ddot\theta + (\boldsymbol{B}-\boldsymbol{B}^\star)\,\dot\theta^2 + (\boldsymbol{g}-\boldsymbol{g}^\star) + (\boldsymbol{f}-\boldsymbol{f}^\star) \right] \tag{3.10.3}$$

which influences the response of the servo equation

$$\ddot{\boldsymbol{e}} + \boldsymbol{K}_v\,\dot{\boldsymbol{e}} + \boldsymbol{K}_p\,\boldsymbol{e} = \boldsymbol{n}_d \tag{3.10.4}$$

3.11 Time Effects in Computing the Model

In all our considerations of control laws based on system modeling we have made the assumption that the entire system was running in continuous time and that the computations in the control laws are made with infinite speed. In practice, it is necessary to take account of time effects such as

- The sampling rate at which sensor information is read and sent to the controller,

- The controller frequency giving the rate at which control signals are computed and sent to the actuators.

To analyze the effects of time delay in computing the model, sampling rate and controller frequency we should use tools from the field of time control, but this is beyond the scope of this presentation.

Let us simply mention that in order to have the confidence that computations are performed quickly enough and that the continuous time approximation is valid we have to take account of the following points:

- **Tracking the reference inputs**: the frequency content of the desired or reference input places an absolute lower bound on the sampling rate. The sampling rate must be at least twice the bandwidth of reference inputs, but this is not usually a limiting factor.
- **Disturbance rejection**: if the sampling period is longer than the correlation time of the disturbance effects (assuming a statistical model for random disturbances) then these disturbances will not be suppressed. In practice, the sample period should be 10 times shorter than the correlation time of the noise.
- **Anti-aliasing**: aliasing occurs when signals coming from analog sensors are converted to digital form. There will be a problem unless the sensor's output is strictly and limited using an anti-aliasing filter. It is also possibly to verify in practical cases that the sampling rate is such that the amount of energy in the aliased signal remains small.
- **Structural resonance effects**: a real mechanism having finite stiffness it will be subject to various kinds of vibrations. The influence of these vibrations on the controller has to be suppressed by choosing a sampling rate which is more than twice the fundamental vibration frequency. A rate 10 times higher is recommended.

3.12 Present Industrial Robot Control Systems

Because of the problems with having accurate knowledge of parameters, it is not clear whether it makes sense to consume time in computing a complicated model based control law for manipulator control. Therefore, the present day manipulators are still controlled with very simple control law which are generally of error driven type. It is still instructive to consider these simpler control schemes within the context of the partitioned controller structure.

3.12.1 Individual joint PID control

Most present industrial robots have a control scheme which could be described in our notation by

$$\boldsymbol{A} = \boldsymbol{I} \qquad \text{and} \qquad \beta = 0 \tag{3.12.1}$$

where $\boldsymbol{I}$ is the nxn identity matrix. The servo portion is

$$\boldsymbol{c}' = \ddot{\theta}_d + \boldsymbol{K}_v \,\dot{\boldsymbol{e}} + \boldsymbol{K}_p \,\boldsymbol{e} + \boldsymbol{K}_I \int_0^t \boldsymbol{e} \, d\tau \tag{3.12.2}$$

where $\boldsymbol{K}_I$, $\boldsymbol{K}_p$ and $\boldsymbol{K}_v$ are constant diagonal gain matrices. In many cases, $\ddot{\theta}_d$ is not available, and this term is then simply set equal to zero. That is, most simpler controllers do not use at all a model based component in their control law.

The advantage of this PID control scheme is that each joint has a separate control system which can be implemented using a separate microprocessor. The servo-error equation is obtained (assuming here $\boldsymbol{K}_I = 0$) by writing

$$\begin{aligned} \boldsymbol{M}^\star &= I \\ \boldsymbol{B}^\star &= \boldsymbol{g}^\star = \boldsymbol{f}^\star = 0 \qquad (3.12.3) \\ \ddot{\boldsymbol{e}} + \boldsymbol{K}_v\dot{\boldsymbol{e}} + \boldsymbol{K}_p\boldsymbol{e} &= (\boldsymbol{M} - \boldsymbol{I})\,\ddot{\theta}_d + \boldsymbol{B}\,\dot{\theta}^2 + \boldsymbol{g} + \boldsymbol{f} \qquad (3.12.4) \end{aligned}$$

Average gains have to be chosen to give approximate critical damping in the center of the robot workspace. In various extreme configurations of the arm the system becomes then either under-damped or over-damped. Maintaining them at a very high value is necessary to minimize the disturbance effect due to the absence of any model in the control law.

3.12.2 Individual joint PID control with effective joint inertia

Some robots have a control scheme in which the inertia properties of the system are modeled to some extent and introduced in the control law by taking

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{M}'(\theta) \\ \beta &= O \qquad (3.12.5) \end{aligned}$$

where $\boldsymbol{M}'$ is a $n \times n$ matrix with functions of configuration on the diagonal. This changing inertia is modeled to try to cancel partially the effective inertia of the system.

Such a model will be easier to keep near critical damping, but will still have to continually suppress disturbances which result from coupling between joints.

3.12.3 Inertial decoupling

A further improvement consists to have a control system in which inertia effects are modeled properly

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{M}^\star(\theta) \\ \beta &= O \qquad (3.12.6) \end{aligned}$$

where $\boldsymbol{M}^\star$ is a $n \times n$ model of the inertia matrix. The closed loop error is then governed by

$$\ddot{\boldsymbol{e}} + \boldsymbol{K}_v\dot{\boldsymbol{e}} + \boldsymbol{K}_p\boldsymbol{e} = {\boldsymbol{M}^\star}^{-1}(\boldsymbol{B}\,\dot{\theta}^2 + \boldsymbol{g} + \boldsymbol{f}) \qquad (3.12.7)$$

The main error to be controlled is due to gravity and friction. Including integral control as suggested by equations (3.6.16) and (3.12.2) is thus essential.

3.13 Cartesian Based Control Systems

In this section we come back to the concept of cartesian based control which we have briefly discussed in the section 1. The concept of cartesian based control is important since accurate trajectory planning requires defining the trajectory into a cartesian space. Therefore, cartesian based control systems will certainly arrive at a fully industrial stage in a near future.

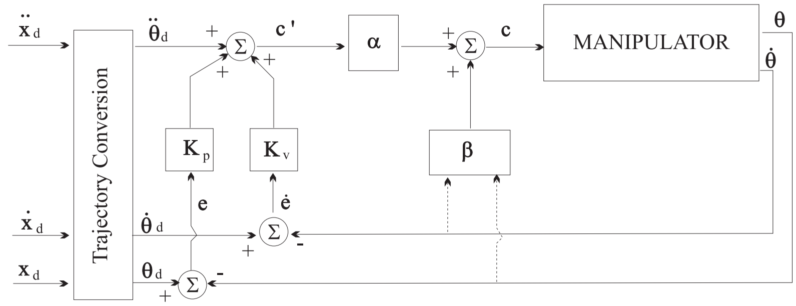


Figure 3.13.1: Joint based control scheme with cartesian trajectory conversion

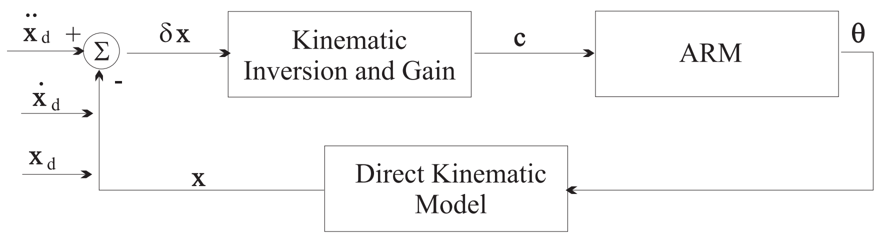


Figure 3.13.2: General structure of cartesian control scheme

3.13.1 Comparison with joint based schemes

So far we have assumed that the desired trajectory is available in terms of joint position, velocity and acceleration. We have then developed joint based control schemes in which we develop trajectory errors in term of desired and actual quantities in joint space.

Very often trajectory planning requires the effector to follow straight lines or other curved paths defined in cartesian coordinates.

It is then theoretically possible to control the trajectory as suggested by figure 3.13.1 by converting first trajectory information into joint space variables, and applying next any of the control laws developed in the previous sections.
The trajectory conversion is a time consuming operation even if done analytically since it involves the three operations

$$\begin{aligned} \theta_d &= INVKIN(\boldsymbol{x}_d) \\ \dot{\theta}_d &= \boldsymbol{J}^{-1}(\boldsymbol{x}_d)\delta\boldsymbol{x}_d \\ \ddot{\theta}_d &= \dot{\boldsymbol{J}}^{-1}\dot{\boldsymbol{x}}_d + \boldsymbol{J}^{-1}\ddot{\boldsymbol{x}}_d \end{aligned} \tag{3.13.1}$$

The solution of the kinematic problem has to be performed explicitly while calculating velocities and accelerations can be done numerically. Numerical differentiation is however a source of noise and lag in the system which have to be avoided.

A alternative approach consists to control directly the cartesian error, in which case the general structure of the controller is that of figure 3.13.2. The output of the system, i.e. the sensed position of the manipulator, has to be converted by means of the kinematic relationships into a cartesian description of position. The current cartesian is then compared to the desired one in order to form errors in cartesian space.

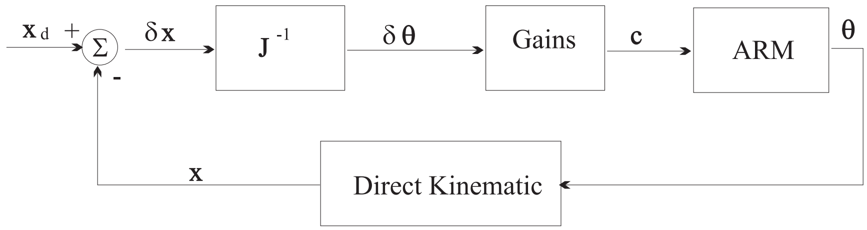


Figure 3.13.3: Inverse Jacobian cartesian control

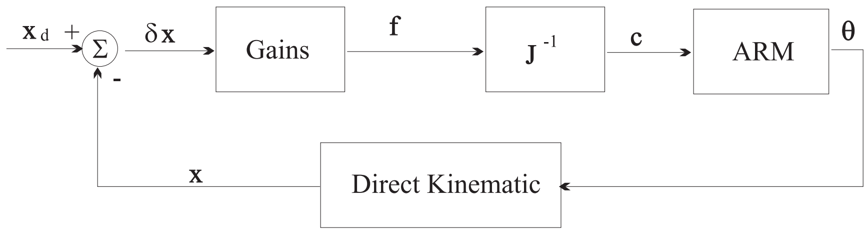


Figure 3.13.4: Transposed Jacobian cartesian control

The most classical implementation of cartesian control and the most intuitive one also is the inverse Jacobian cartesian control. Its general structure is that of figure 3.13.2, but use is made of the differential kinematic model in a first step to convert cartesian errors $\delta \boldsymbol{x}$ into joint space error $\delta\theta$. The resulting errors $\delta\theta$ are then multiplied by gains to compute torques which will tend to reduce these errors. Note that the figure 3.13.3 shows a simplified version of such a controller in the sense that, for clarity, velocity feedback has not been represented. It could be added in a straight forward manner.
Another possible implementation of cartesian control could consist of converting the cartesian errors into forces first, and then transforming these forces into equivalent joint torques which tend to reduce the observed errors. The cartesian scheme so obtained is shown on figure 3.13.4. It presents the advantage that inverting the Jacobian matrix real time is avoided.

The forces $\boldsymbol{f}$ on figure 3.13.4 have the meaning of forces to be applied at the end effector level in order to reduce the cartesian error.
The exact dynamic performance of such cartesian controllers can only be predicted by numerical simulation. One observes that both can be made stable by appropriate gain selection, but their dynamic response varies with arm configuration.

Bibliography

[1] M. BRADY, J.M. HOLLERBACH, T.L. JOHNSON, T. LOZANO-PEREZ, M.T. MASON, *Robot Motion : Planning and Control*, MIT Press, Cambridge, Mass., USA,1982.

[2] Ph. COIFFET, M. CHIROUZE, *An Introduction to Robot Technology*, Kogan Page, London, 1982.

[3] J. J. CRAIG, *Lecture notes in Robotics*, Stanford University, 1983.

[4] B. GORLA, M. RENAUD, *Modèles des robots manipulateurs. Application à leur commande*, Cepadues Editions, Toulouse, 1984.

[5] J. SIMON, *Mathematica, Programmatie en Control ven Industriele Robots*, KUL, Afd. Mech. Konst. en Prod., Leuven, Belgium, 1983.

Chapter 4

KINEMATICS OF THE RIGID BODY

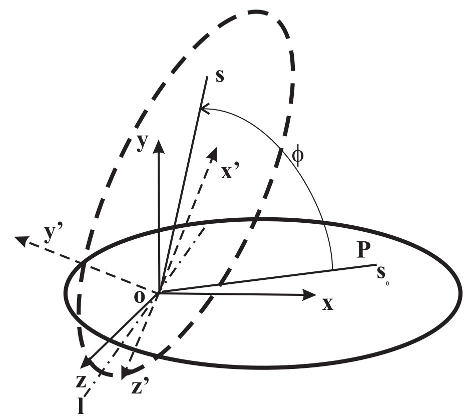


Figure 4.2.1: Rotation of a body and of an arbitrary a point P about an arbitrary direction $\vec{l}$

4.1 Introduction

The introduction on robot technology has learned us that, from a mechanical point of view, a robot manipulator may be regarded as a kinematic chain of rigid bodies articulated together through kinematic pairs, or *joints*. Each body of the system is then subject to translation and rotary motions generated by the relative displacements occurring at joints.

In order to describe the motion of each element in the kinematic chain, from the support to the effector, an appropriate formalism is needed.

One of the crucial aspects to establish such a formalism is the appropriate description of *finite rotations*. The concept of finite rotation will thus be reviewed and revisited in the next sections, keeping in mind the specific needs of robotics.

We will establish next the relationships giving the position, velocity and acceleration of an arbitrary point attached to a rigid body which undergoes arbitrary motion.

Finally, the concept of *homogeneous transformation* will be introduced to describe the kinematics of articulated chains. As it will be shown, homogeneous vector notation allows us to combine in one single matrix transformation both rotation and translation operations. It reduces thus any operation describing arbitrary rigid body motion to a matrix product. As a consequence, with homogeneous notation the kinematics of an open-tree simple structure may be expressed as an ordered sequence of matrix products.

Due to the simplification that it brings into the description of object manipulation, homogeneous notation is a computational tool of fundamental importance both in computer graphics and in robotics.

4.2 The rotation operator

Let $Oxyz$ be a cartesian frame in which the position of a given point P is specified, and let us represent the corresponding position vector $\vec{s}$ in matrix form by the unicolumn matrix $\boldsymbol{s}$ collecting its cartesian components.

Suppose that a body is rotated of an angle ϕ about the specified direction $\vec{l}$. One of its points P characterized by an initial position vector $\vec{s}_0$ is moved to a new position whose coordinates are $\vec{s}$.

The rotation of the body is fully described by frame transformation between the fixed coordinate system $Oxyz$ and the body coordinates $O'x'y'z'$, which was initially coincident with the inertial frame.

In matrix form, the transformation from $\vec{s}_0$ to $\vec{s}$ may be represented by a linear transformation involving a matrix $\boldsymbol{R}$ of dimension (3×3) :

$$\boldsymbol{s} \;=\; \boldsymbol{R}\,\boldsymbol{s}_0 \tag{4.2.1}$$

4.2.1 Properties of rotation

Equation (4.2.1) will effectively represent a rotation operation if the length of $\vec{s}$ is preserved. This condition may be given in explicit matrix form

$$\begin{aligned} \boldsymbol{s}^T\;\boldsymbol{s} &= \boldsymbol{s}_0^T\;\boldsymbol{R}^T\;\boldsymbol{R}\,\boldsymbol{s}_0 \\ &= \boldsymbol{s}_0^T\;\boldsymbol{s}_0 \end{aligned}$$

and implies thus

$$\boldsymbol{R}^T\,\boldsymbol{R} \;=\; \boldsymbol{I}. \tag{4.2.2}$$

Therefore, the matrix product of an arbitrary position vector $\boldsymbol{s}$ by an orthogonal matrix $\boldsymbol{R}$

$$\boldsymbol{s} \;=\; \boldsymbol{R}\,\boldsymbol{s}_0 \qquad \text{with} \qquad \boldsymbol{R}^{-1} \;=\; \boldsymbol{R}^T$$

expresses the rotation of any vector $\vec{s}$ about a specified direction.

4.2.2 Remark

The position vector of P is initially given by $\vec{s}_0$ in both coordinate systems. After rotation, point P has reached a position that is still given by $\vec{s}_0$ in the body coordinate system, which has rotated with P, while the coordinates of P are now $\vec{s}$ when seen from the initial frame.

In the following, we will decide to by $\vec{p}$ and column matrix $\boldsymbol{p}$ the absolute coordinates of point P in frame $Oxyz$ after rotation , whereas we will note $\vec{p}'$ and $\boldsymbol{p}'$ the coordinates of the same point in the relative coordinate $Ox'y'z'$ attached to the body (or body coordinate system).

After rotation, it comes:

$$\boxed{\boldsymbol{p} \;=\; \boldsymbol{R}\,\boldsymbol{p}' \qquad \text{with} \qquad \boldsymbol{R}^{-1} \;=\; \boldsymbol{R}^T} \tag{4.2.3}$$

4.3 Position and orientation of a rigid body

Let us consider the rigid body V of figure 4.3.1 and adopt the following definitions

- O is the origin of the absolute cartesian frame $Oxyz$
- O' is the reference point attached to body V

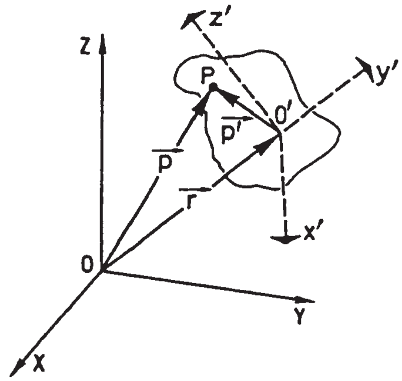


Figure 4.3.1: Position and orientation of a rigid body

- $O'x'y'z'$ is a cartesian frame attached to V and centered in O'
- $\vec{r}$ is the position vector of O' relatively to O
- $\vec{p}$ is the position vector of P relatively to O
- $\vec{p'}$ is the position vector of P relatively to O'

In vector form, the position vector at point P can be decomposed in the form

$$\vec{p} = \vec{r} + \vec{p'} \tag{4.3.1}$$

In order to express equation (4.3.1) in matrix notation, let us represent respectively by

- $\boldsymbol{p}$ and $\boldsymbol{r}$ the cartesian components of $\vec{p}$ and $\vec{r}$ in the absolute reference frame $Oxyz$
- $\boldsymbol{p}'$ the cartesian components of $\vec{p'}$ in the moving reference frame $O'x'y'z'$

The frame transformation law giving the cartesian components of P in the absolute frame $Oxyz$ takes then the form

$$\boldsymbol{p} = \boldsymbol{r} + \boldsymbol{R}\,\boldsymbol{p}' \tag{4.3.2}$$

where $\boldsymbol{R}$ is the orthogonal matrix describing the finite rotation from frame $O'x'y'z'$ to $Oxyz$. the first term in (4.3.2) represents the translation from O to O', and the second one expresses the rotation of the relative position vector $\vec{p}'$.

The relationship (4.3.2) provides the basis of the matrix formalism to describe the kinematics of a rigid body.

4.4 Algebraic expression of the rotation operator

Various techniques are available to represent the rotation operator $\boldsymbol{R}$ in matrix form. It can be expressed in terms of various sets of parameters such as direction cosines, Euler angles, Bryant angles, Euler parameters, etc.

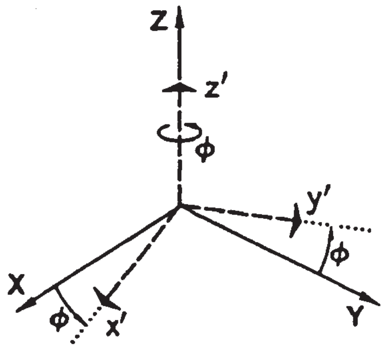


Figure 4.5.1: Rotation in the Oxy plane

It is easy to show that an arbitrary rotation may be expressed in terms of a set of *three independent parameters.* Indeed $\boldsymbol{R}$ is a 3×3 matrix containing thus 9 terms, and it can be decomposed into column-vectors

$$\boldsymbol{R} \;=\; [\,\boldsymbol{r}_1\;\boldsymbol{r}_2\;\boldsymbol{r}_3\,] \tag{4.4.1}$$

The orthonormality property (4.2.3) has for consequence that the vectors $\boldsymbol{r}_j$ are linked by the 6 constraints.

$$\boldsymbol{r}_i^T\;\boldsymbol{r}_j \;=\; \delta_{ij} \qquad (i,j \;=\; 1,2,3)$$

where δ_{ij} represents the Kronecker symbol

$$\delta_{ij} \;=\; \begin{cases} 1 & \text{if} \qquad \text{i=j} \\ 0 & \text{otherwise.} \end{cases}$$

so that one can write

$$\boldsymbol{R} \;=\; \boldsymbol{R}(\alpha_1,\alpha_2,\alpha_3) \tag{4.4.2}$$

where α_1, α_2, α_3 are three independent parameters retained to describe the rotation. Some of the possible choices will be described below.

4.5 The plane rotation operator

Let us consider the simplest rotation operation corresponding to finite rotation about a coordinate axis. For example, figure 4.5.1 represents the case where a rotation ϕ is performed about the z coordinate axis.

For a vector $\boldsymbol{r}$ with components $\boldsymbol{r} \;=\; [\,x\;y\;z\,]$ one obtains the change of coordinates

$$\begin{aligned} x &= x'\cos\phi \;-\; y'\sin\phi \\ y &= x'\sin\phi \;+\; y'\cos\phi \\ z &= z' \end{aligned} \tag{4.5.1}$$

or, in matrix form

$$\boldsymbol{r} \;=\; \boldsymbol{R}\,\boldsymbol{r}' \tag{4.5.2}$$

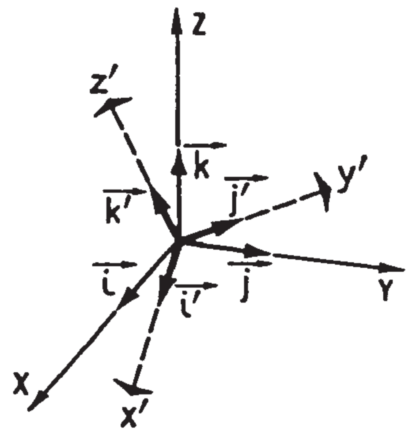


Figure 4.6.1: Finite rotation in terms of direction cosines

with the rotation operator

$$\boldsymbol{R}(\boldsymbol{z},\phi) \;=\; \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.5.3}$$

Similarly, for a rotation θ about the y axis and a rotation ψ about the x axis, one would obtain the rotation operators

$$\boldsymbol{R}(\boldsymbol{y},\theta) \;=\; \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \tag{4.5.4}$$

and

$$\boldsymbol{R}(\boldsymbol{x},\psi) \;=\; \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} \tag{4.5.5}$$

4.6 Finite rotation in terms of direction cosines

The most obvious expression for a rotation operator performing an arbitrary rotation is the one obtained in terms of direction cosines. To that purpose, let us note (Figure 4.6.1) by $(\vec{i},\vec{j},\vec{k})$ and $(\vec{i}',\vec{j}',\vec{k}')$ the unit vectors spanning the cartesian frames $Oxyz$ and $O'x'y'z'$. An arbitrary position vector $\vec{r}$ may be expressed indifferently in the $Oxyz$ and $O'x'y'z'$ frames

$$\begin{aligned} \boldsymbol{r} &= x\vec{i} \;+\; y\vec{j} \;+ z\vec{k} \\ &= x'\vec{i}' \;+\; y'\vec{j}' \;+ z'\vec{k}' \end{aligned}$$

Since both sets of unit vectors are orthonormal, a component such as x may be calculated by

$$\begin{aligned} x &= \vec{i}\cdot\boldsymbol{r} \\ &= \vec{i}\cdot\vec{i}'\;x' \;+\; \vec{i}\cdot\vec{j}'\;y' \;+\; \vec{i}\cdot\vec{k}'\;z' \end{aligned}$$

and similarly

$$\begin{aligned} y &= \vec{j} \cdot \boldsymbol{r} \\ &= \vec{j} \cdot \vec{i}' \; x' \; + \; \vec{j} \cdot \vec{j}' \; y' \; + \; \vec{j} \cdot \vec{k}' \; z' \\ z &= \vec{k} \cdot \boldsymbol{r} \\ &= \vec{k} \cdot \vec{i}' \; x' \; + \; \vec{k} \cdot \vec{j}' \; y' \; + \; \vec{k} \cdot \vec{k}' \; z' \end{aligned}$$

giving the expression of the rotation operator in terms of direction cosines

$$\boldsymbol{R} \; = \; \begin{bmatrix} \cos(\vec{i}, \vec{i}') & \cos(\vec{i}, \vec{j}') & \cos(\vec{i}, \vec{k}') \\ \cos(\vec{j}, \vec{i}') & \cos(\vec{j}, \vec{j}') & \cos(\vec{j}, \vec{k}') \\ \cos(\vec{k}, \vec{i}') & \cos(\vec{k}, \vec{j}') & \cos(\vec{k}, \vec{k}') \end{bmatrix} \tag{4.6.1}$$

Let us note that when using this representation

- The dependence of the operator with respect to only 3 parameters is not immediately apparent;
- On the other hand, the orthonormality property is obvious since the inverse transformation

$$\boldsymbol{r}' \; = \; \boldsymbol{R}^{-1} \; \boldsymbol{r} \tag{4.6.2}$$

is obtained in a similar manner

$$\begin{aligned} x' &= (\vec{i}' \cdot \vec{i}) \; x \; + \; (\vec{i}' \cdot \vec{j}) \; y \; + \; (\vec{i}' \cdot \vec{k}) \; z \\ y' &= (\vec{j}' \cdot \vec{i}) \; x \; + \; (\vec{j}' \cdot \vec{j}) \; y \; + \; (\vec{j}' \cdot \vec{k}) \; z \\ z' &= (\vec{k}' \cdot \vec{i}) \; x \; + \; (\vec{k}' \cdot \vec{j}) \; y \; + \; (\vec{k}' \cdot \vec{k}) \; z \end{aligned}$$

The scalar product being commutative, one easily verifies that

$$\boldsymbol{R}^{-1} \; = \; \boldsymbol{R}^{T} \tag{4.6.3}$$

4.7 Finite rotation in terms of dyadic products

A complementary result to (4.6.1) consists to observe that, using matrix notation, any finite rotation can be written in the form

$$\boldsymbol{R} \; = \; \boldsymbol{l} \, \boldsymbol{l}'^{T} \; + \; \boldsymbol{m} \, \boldsymbol{m}'^{T} \; + \; \boldsymbol{n} \, \boldsymbol{n}'^{T} \tag{4.7.1}$$

where $(\boldsymbol{l}, \boldsymbol{m}, \boldsymbol{n})$ and $(\boldsymbol{l}', \boldsymbol{m}', \boldsymbol{n}')$ are two sets of orthonormal vectors defining two orthogonal bases.

The proof can be established by noticing at first that:

$$\boldsymbol{I} \; = \; \boldsymbol{l}' \boldsymbol{l}'^{T} \; + \; \boldsymbol{m}' \, \boldsymbol{m}'^{T} \; + \; \boldsymbol{n}' \, \boldsymbol{n}'^{T} \tag{4.7.2}$$

This can be easily demonstrated by showing that the following relation

$$\boldsymbol{I} \, \boldsymbol{a} \; = \; \boldsymbol{a}$$

holds for any vector $\boldsymbol{a} = x'\boldsymbol{l'} + y'\boldsymbol{m'} + z'\boldsymbol{n'}$.

Since

$$\begin{aligned} \boldsymbol{R}^T\ \boldsymbol{l} &= \boldsymbol{l'} \\ \boldsymbol{R}^T\ \boldsymbol{m} &= \boldsymbol{m'} \\ \boldsymbol{R}^T\ \boldsymbol{n} &= \boldsymbol{n'} \end{aligned}$$

the lemma (4.7.2) can be written as follows:

$$\begin{aligned} \boldsymbol{I}\ &= (\boldsymbol{R}^T\boldsymbol{l})\ \boldsymbol{l'}^T\ +\ (\boldsymbol{R}^T\boldsymbol{m})\ \boldsymbol{m'}^T\ +\ (\boldsymbol{R}^T\boldsymbol{l})\ \boldsymbol{n'}^T \\ \boldsymbol{I}\ &= \boldsymbol{R}^T\left(\boldsymbol{l}\ \boldsymbol{l'}^T + \ \boldsymbol{m}\ \boldsymbol{m'}^T + \ \boldsymbol{n}\ \boldsymbol{n'}^T\right) \end{aligned}$$

which leads to the result of equation (4.7.1) because $\boldsymbol{R}^T = \boldsymbol{R}^{-1}$.

4.8 Composition of Finite Rotations

4.8.1 Composition rule of rotations

Let's consider two successive rotations:

- Rotation 1 is the rotation that brings frame $Oxyz$ onto frame $Ox_1y_1z_1$,
- Rotation 2 is the rotation that brings frame $Ox_1y_1z_1$ onto frame $Ox_2y_2z_2$

The coordinates in the different frame systems are related by the following relations:

$$\boldsymbol{x}\ =\ \boldsymbol{R}_1\ \boldsymbol{x}_1 \tag{4.8.1}$$

$$\boldsymbol{x}_1\ =\ \boldsymbol{R}_2\ \boldsymbol{x}_2 \tag{4.8.2}$$

The overall rotations that transforms frame $Oxyz$ into frame $Ox_2y_2z_2$ is given by:

$$\boldsymbol{x}\ =\ \boldsymbol{R}_1\ \boldsymbol{x}_1\ =\ \boldsymbol{R}_1\ (\boldsymbol{R}_2\ \boldsymbol{x}_2) =\ (\boldsymbol{R}_1\ \boldsymbol{R}_2)\ \boldsymbol{x}_2$$

because of the associative property of matrix multiplications. Fin ally it comes that

$$\boldsymbol{x}\ =\ \boldsymbol{R}\ \boldsymbol{x}_2 \quad \text{with} \quad \boldsymbol{R}\ =\ \boldsymbol{R}_1\ \boldsymbol{R}_2 \tag{4.8.3}$$

4.8.2 Non commutative character of finite rotations

As an example, let us consider an object (Figure 4.8.1) undergoing two successive rotations $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$ of 90° about axes $\boldsymbol{z}$ and $\boldsymbol{y}$ respectively

$$\begin{aligned} \boldsymbol{R}_1\ &=\ \boldsymbol{R}(\boldsymbol{z}, 90^\circ) \\ \boldsymbol{R}_2\ &=\ \boldsymbol{R}(\boldsymbol{y}, 90^\circ) \end{aligned}$$

One knows that the matrix product is a non−commutative operation

$$\boldsymbol{R}_1\ \boldsymbol{R}_2\ \neq\ \boldsymbol{R}_2\ \boldsymbol{R}_1$$

Geometrically, the *non−commutativity* of finite rotations expresses the fact that reversing the order of two successive rotations generates different geometric configurations.

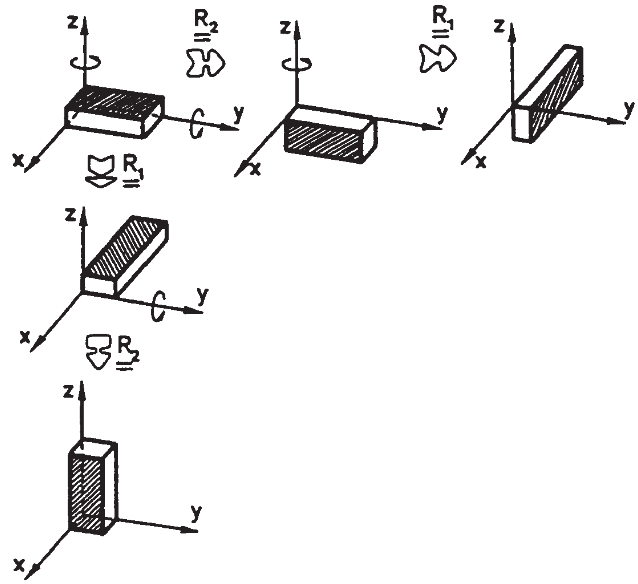


Figure 4.8.1: Non−commutativity of finite rotations

4.9 Euler angles

Euler angles are the most classical set of 3 independent parameters which allow to express in a unique manner the rotation of a rigid body about an arbitrary axis. They are specially well adapted to the kinematic description of spinning systems such as tops, gyroscopes, etc.

The Euler angle formalism consists to decompose the rotation from $Oxyz$ to $Ox'y'z'$ into three elementary rotations expressed in body axes (Figure 4.9.1):

- a rotation ϕ about Oz: $\boldsymbol{R}(\boldsymbol{z}, \phi)$
- a rotation θ about Ox_1: $\boldsymbol{R}(\boldsymbol{x}_1, \theta)$
- a rotation ψ about Oz_2: $\boldsymbol{R}(\boldsymbol{z}_2, \psi)$

By combining the 3 successive rotations, the frame transformation can be written

$$\boldsymbol{r} \;=\; \boldsymbol{R}(\boldsymbol{z}, \phi)\ \boldsymbol{R}(\boldsymbol{x}_1, \theta)\ \boldsymbol{R}(\boldsymbol{z}_2, \psi)\ \boldsymbol{r}' \;=\; \boldsymbol{R}\,\boldsymbol{r}'$$

with the rotation operator expression

$$\boldsymbol{R} \;=\; \boldsymbol{R}(\boldsymbol{z}, \phi)\ \boldsymbol{R}(\boldsymbol{x}_1, \theta)\ \boldsymbol{R}(\boldsymbol{z}_2, \psi)$$

and where the elementary rotations about $\boldsymbol{z}$ and $\boldsymbol{x}$ axes are given by equations (4.5.3) and (4.5.5). One obtains explicitly

$$\boldsymbol{R} \;=\; \begin{bmatrix} \cos\phi\cos\psi - \sin\phi\cos\theta\sin\psi & -\cos\phi\sin\psi - \sin\phi\cos\theta\cos\psi & \sin\phi\sin\theta \\ \sin\phi\cos\psi + \cos\phi\cos\theta\sin\psi & -\sin\phi\sin\psi + \cos\phi\cos\theta\cos\psi & -\cos\phi\sin\theta \\ \sin\theta\sin\psi & \sin\theta\cos\psi & \cos\theta \end{bmatrix} \qquad (4.9.1)$$

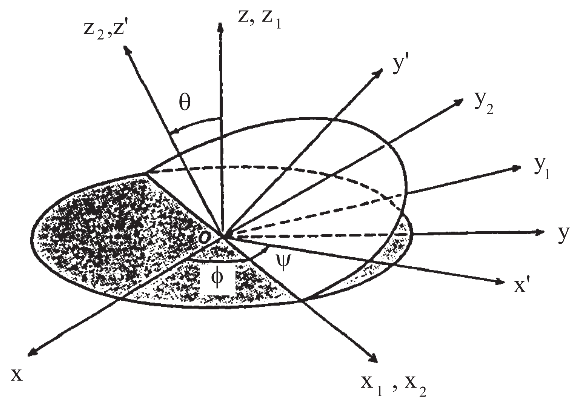


Figure 4.9.1: Euler angles

4.9.1 Singular values

From the very definition of Euler angles it follows that the matrix (4.9.1) becomes singular when $\theta = 0$ or π, since both rotation axes along $\boldsymbol{z}$ become collinear. The rotation reduces then to a single rotation $(\phi \pm \psi)$ about $\boldsymbol{z}$.

4.9.2 Inversion

Solving the inverse problem consists to deduce the Euler angles (ψ, θ, ϕ) from the numerical values of $\boldsymbol{R}$ given :

$$\boldsymbol{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \boldsymbol{R}(\psi, \theta, \phi) \quad (4.9.2)$$

let us determine the corresponding values (ψ, θ, ϕ) of Euler angles. Comparing expressions (4.6.3) and (4.7.1) shows that a crude solution would consist of calculating

$$\theta = \cos^{-1}(r_{33}), \qquad \phi = -\cos^{-1}(\frac{r_{23}}{\sin\theta}), \qquad \psi = \cos^{-1}(\frac{r_{32}}{\sin\theta}) \quad (4.9.3)$$

However, this method of solution is not satisfactory for the reasons that an indeterminacy about signs of angles remains and that it becomes very unaccurate in the vicinity of singular values.

In practice an accurate method of solution consists of making a systematic use of the function $ATAN2$ from FORTRAN programming language. The FORTRAN function $\psi = ATAN2(x, y)$ gives $\psi = tan^{-1}(\frac{x}{y})$ and uses the signs of x and y to determine the quadrant in which the solution lies. Thus one computes first

$$\psi = ATAN2(r_{31}, r_{32}) \quad (4.9.4)$$

It is then possible to calculate θ and ϕ without ambiguity if $\sin\theta \neq 0$:

$$\theta = ATAN2(\sqrt{r_{31}^2 + r_{32}^2}, r_{33}) \quad (4.9.5)$$

$$\phi = ATAN2(r_{13}, -r_{23}) \quad (4.9.6)$$

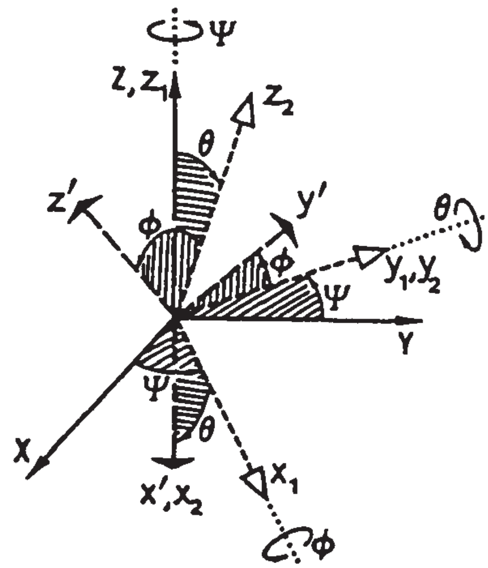


Figure 4.10.1: Bryant angles

Although a second solution exists by using the negative square root in (4.9.5), we always compute in this way the solution for which $0 \leq \theta \leq \pi$.

If the solution degenerates ($\theta = 0$ or π), it is then possible to choose conveniently $\theta = \phi = 0$ and calculate ψ by

$$\psi \;=\; ATAN2(r_{21}, r_{11}) \tag{4.9.7}$$

4.10 Finite rotations in terms of Bryant angles

In order to represent the orientation of some mechanical systems such as a flying object (airplane, robot effector) or devices such as Cardan joints, it is better adapted to define the finite rotation operator in terms of three elementary rotations about distinct axes, called *roll, pitch and yaw* (RPY) axes. The use of RPY axes is quite common for task description in robotics.

Let us decompose the total rotation $\boldsymbol{R}$ into 3 elementary rotations (4.10.1)

- a ψ rotation about O_z: $\boldsymbol{R}(\boldsymbol{z}, \psi)$
- a θ rotation about Oy_1: $\boldsymbol{R}(\boldsymbol{y}_1, \theta)$
- a ϕ rotation about Ox_2: $\boldsymbol{R}(\boldsymbol{x}_2, \phi)$

In terms of these 3 successive rotations, the frame transformation can be written

$$\boldsymbol{r} \;=\; \boldsymbol{R}(\boldsymbol{z}, \psi)\; \boldsymbol{R}(\boldsymbol{y}_1, \theta)\; \boldsymbol{R}(\boldsymbol{x}_2, \phi) \;=\; \boldsymbol{R}\,\boldsymbol{r}' \tag{4.10.1}$$

with

$$\boldsymbol{R} \;=\; \boldsymbol{R}(\boldsymbol{z}, \psi)\; \boldsymbol{R}(\boldsymbol{y}_1, \theta)\; \boldsymbol{R}(\boldsymbol{x}_2, \phi) \tag{4.10.2}$$

where the elementary rotations about $\boldsymbol{z}$, $\boldsymbol{y}_1$ and $\boldsymbol{x}_2$ are given by equations (4.5.3), (4.5.4) and (4.5.5). One obtains explicitly

$$\boldsymbol{R} = \begin{bmatrix} \cos\theta\cos\psi & \sin\theta\sin\phi\cos\psi - \sin\psi\cos\phi & \sin\theta\cos\phi\cos\psi + \sin\psi\sin\phi \\ \cos\theta\sin\psi & \sin\theta\sin\phi\sin\psi + \cos\psi\cos\phi & \sin\theta\cos\phi\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \quad (4.10.3)$$

4.10.1 Singularities

A singularity of (4.10.3) occurs when $\theta = \pm\frac{\pi}{2}$, since axes $\boldsymbol{z}$ and $\boldsymbol{x}_2$ become then collinear.

4.10.2 Inversion

The inversion procedure is the same as the one used for Euler angles

$$\begin{aligned} \psi &= ATAN2(r_{21}, r_{11}) \\ \theta &= ATAN2(-r_{31}, \sqrt{r_{11}^2 + r_{21}^2}) \\ \phi &= ATAN2(r_{32}, r_{33}) \end{aligned} \quad (4.10.4)$$

Although a second solution exists when taking the negative square root, equation (4.10.1) provides a value of θ such that

$$-\frac{\pi}{2} \le \theta \le \frac{\pi}{2} \quad (4.10.5)$$

If $\theta = \pm\frac{\pi}{2}$, we have already indicated that the solution degenerates. Then, we can conveniently take $\phi = 0$ and

$$\begin{cases} \psi = ATAN2(r_{12}, r_{22}) & \text{if} \quad \theta = \frac{\pi}{2}, \\ \psi = ATAN2(-r_{12}, r_{22}) & \text{if} \quad \theta = -\frac{\pi}{2}. \end{cases} \quad (4.10.6)$$

4.11 Unique rotation about an arbitrary axis

Euler's theorem on finite rotations states that any finite rotation can be expressed as a unique rotation of angle ϕ about an appropriate axis $\boldsymbol{e}$ (Figure 4.11.1).
One numbers then 4 parameters to describe the rotation

$$l_x,\ l_y,\ l_z, \quad \text{and} \quad \phi \quad (4.11.1)$$

but they are linked by the constraint

$$|\boldsymbol{l}| = \sqrt{l_x^2 + l_y^2 + l_z^2} = 1, \qquad \phi \in [0, \pi]$$

The most intuitive procedure to construct the rotation operator in equivalent angle-axis form consists to decompose the transformation in three phases

i) 2 elementary successive rotations (Figure 4.11.2)

$$\boldsymbol{R}(\boldsymbol{z}, -\alpha) \qquad \text{and} \qquad \boldsymbol{R}(\boldsymbol{y}, +\beta)$$

bring the $\boldsymbol{e}$ axis into coincidence with the principal direction Ox; they can be combined in a unique rotation

$$\boldsymbol{C} = \boldsymbol{R}(\boldsymbol{y}, +\beta)\boldsymbol{R}(\boldsymbol{z}, -\alpha) \quad (4.11.2)$$

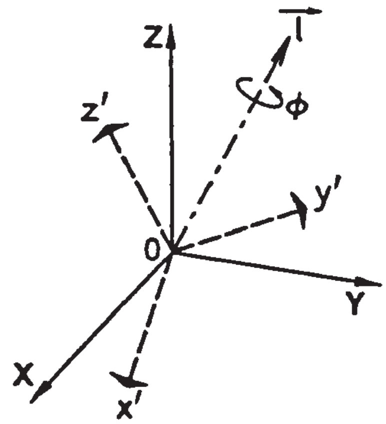


Figure 4.11.1: Expression of a finite rotation as a unique rotation ϕ about an axis $\boldsymbol{e}$

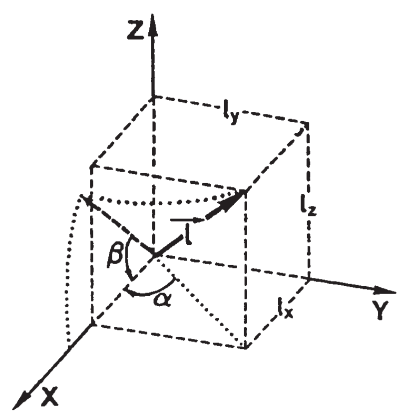


Figure 4.11.2: Superposition of two vectors through two successive rotations

ii) the rotation ϕ about the Ox axis is described by the elementary rotation $\boldsymbol{R}(\boldsymbol{x}, \phi)$;

iii) the $\boldsymbol{e}$ axis is placed back into position by the inverse transformation $\boldsymbol{C}^{-1} = \boldsymbol{C}^T$

The resulting operation can be expressed in the form

$$\boldsymbol{R}(\boldsymbol{l}, \phi) \;=\; \boldsymbol{C}^T \; \boldsymbol{R}(\boldsymbol{x}, \phi) \; \boldsymbol{C} \tag{4.11.3}$$

with the matrix

$$\boldsymbol{C} \;=\; \begin{bmatrix} \cos\alpha\cos\beta & \sin\alpha\cos\beta & \sin\beta \\ -\sin\alpha & \cos\alpha & 0 \\ -\cos\alpha\sin\beta & -\sin\alpha\sin\beta & \cos\beta \end{bmatrix} \tag{4.11.4}$$

If one notes further that

$$\begin{aligned} \sin\beta = l_z \qquad & \cos\beta \;=\; \sqrt{l_x^2 \,+\, l_y^2} \\ \sin\alpha = \frac{l_y}{\sqrt{l_x^2 \,+\, l_y^2}} \qquad & \cos\alpha \;=\; \frac{l_x}{\sqrt{l_x^2 \,+\, l_y^2}} \end{aligned} \tag{4.11.5}$$

we have

$$\boldsymbol{C} \;=\; \begin{bmatrix} l_x & l_y & l_z \\ -\frac{l_y}{d} & \frac{l_x}{d} & 0 \\ -\frac{l_x l_z}{d} & -\frac{l_y l_z}{d} & d \end{bmatrix} \qquad \text{with} \qquad d \;=\; \sqrt{l_x^2 \,+\, l_y^2} \tag{4.11.6}$$

and one obtains the explicit expression for the rotation operator

$$\boldsymbol{R} \;=\; \begin{bmatrix} l_x^2 \; V\phi \,+\, C\phi & l_x \; l_y \; V\phi \,-\, l_z \; S\phi & l_x \; l_z \; V\phi \,+\, l_y \; S\phi \\ l_x \; l_y \; V\phi \,+\, l_z \; S\phi & l_y^2 \; V\phi \,+\, C\phi & l_y \; l_z \; V\phi \,-\, l_x \; S\phi \\ l_x \; l_z \; V\phi \,-\, l_y \; S\phi & l_y \; l_z \; V\phi \,+\, l_x \; S\phi & l_z^2 \; V\phi \,+\, C\phi \end{bmatrix} \tag{4.11.7}$$

with the notations

$$C\phi \;=\; \cos\phi, \qquad S\phi \;=\; \sin\phi, \qquad V\phi \;=\; vers(\phi) \;=\; 1 - \cos\phi \tag{4.11.8}$$

An alternate proof of this result can be obtained by starting from equation (4.7.1)

$$\boldsymbol{R} \;=\; \left[\boldsymbol{l} \; \boldsymbol{l}'^T \;+\; \boldsymbol{m} \; \boldsymbol{m}'^T \;+\; \boldsymbol{n} \; \boldsymbol{n}'^T\right] \tag{4.11.9}$$

as follows. Suppose that the rotation ϕ is performed around the $\boldsymbol{l}$ axis. Then, vectors $\boldsymbol{m}'$ and $\boldsymbol{n}'$ are transformed according to

$$\begin{aligned} \boldsymbol{l}' &= \boldsymbol{l} \\ \boldsymbol{m}' &= \; \boldsymbol{R}^T \, \boldsymbol{m} \;=\; \boldsymbol{m} \; \cos\phi \;-\; \boldsymbol{n} \; \sin\phi \\ \boldsymbol{n}' &= \; \boldsymbol{R}^T \, \boldsymbol{n} \;= \boldsymbol{m} \; \sin\phi \;+\; \boldsymbol{n} \; \cos\phi \end{aligned} \tag{4.11.10}$$

and substitution of (4.11.10) into (4.11.9) yields to

$$\boldsymbol{R} \;=\; \left[\boldsymbol{l} \; \boldsymbol{l}^T \;+\; \cos\phi \; (\boldsymbol{m} \; \boldsymbol{m}^T + \boldsymbol{n} \; \boldsymbol{n}^T) \;+\; \sin\phi \; (-\boldsymbol{m} \; \boldsymbol{n}^T + \boldsymbol{n} \; \boldsymbol{m}^T)\right] \tag{4.11.11}$$

If we further note that since $(\boldsymbol{l}, \boldsymbol{m}, \boldsymbol{n})$ form an orthonormal basis, the last term represents the matrix form of the cross product $\boldsymbol{m} \times \boldsymbol{n} = \boldsymbol{l}$ and can thus be written in the form

$$\boldsymbol{n} \; \boldsymbol{m}^T - \boldsymbol{m} \; \boldsymbol{n}^T \;=\; \tilde{\boldsymbol{l}} \tag{4.11.12}$$

where $\tilde{\boldsymbol{l}}$ is the skew-symmetric matrix

$$\tilde{\boldsymbol{l}} = \begin{bmatrix} 0 & -l_z & l_y \\ l_z & 0 & -l_x \\ -l_y & l_x & 0 \end{bmatrix} \tag{4.11.13}$$

The resulting rotation operator is then equivalent to equation (4.11.7), written in matrix form

$$\boldsymbol{R} = \left[\boldsymbol{l}\,\boldsymbol{l}^T + \cos\phi\,(\boldsymbol{m}\,\boldsymbol{m}^T + \boldsymbol{n}\,\boldsymbol{n}^T) + \sin\phi\,\tilde{\boldsymbol{l}})\right] \tag{4.11.14}$$

Taking into account that

$$\boldsymbol{l}\boldsymbol{l}^T + \boldsymbol{m}\boldsymbol{m}^T + \boldsymbol{n}\boldsymbol{n}^T = \boldsymbol{I}$$

one gets finally

$$\boldsymbol{R} = \left[\cos\phi\,\boldsymbol{I} + (1-\cos\phi)\,\boldsymbol{l}\,\boldsymbol{l}^T + \sin\phi\,\tilde{\boldsymbol{l}}\right] \tag{4.11.15}$$

4.11.1 inversion

In order to invert (4.11.6), let us evaluate the sum of the diagonal terms

$$\begin{aligned} trace(\boldsymbol{R}) &= r_{11} + r_{22} + r_{33} \\ &= (l_x^2 + l_y^2 + l_z^2)(1-\cos\phi) + 3\cos\phi \\ &= 1 + 2\cos\phi \end{aligned} \tag{4.11.16}$$

Similarly, to obtain the sine of ϕ let us define the *vector part* of $\boldsymbol{R}$.

If ϵ_{ijk} is the permutation symbol such that:

$$\begin{aligned} &\epsilon_{123} = \epsilon_{231} = \epsilon_{312} = 1 \\ &\epsilon_{132} = \epsilon_{321} = \epsilon_{213} = -1 \\ &\epsilon_{ijk} = 0 \quad \text{otherwise} \end{aligned}$$

Then

$$vec(\boldsymbol{R}) = -\epsilon_{ijk}\,r_{jk} = \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix} = 2\,\boldsymbol{l}\,\sin\phi \tag{4.11.17}$$

It is then possible to calculate the angle ϕ and the direction of the rotation by

$$\phi = ATAN2(|vec(\boldsymbol{R})|, tr(\boldsymbol{R}) - 1) \tag{4.11.18}$$

and

$$\boldsymbol{l} = \frac{vec(\boldsymbol{R})}{|vec(\boldsymbol{R})|} \tag{4.11.19}$$

One has also

$$trace(\boldsymbol{R}) = 1 + 2\cos\phi$$

giving

$$\cos\phi = \frac{1}{2}[trace(\boldsymbol{R}) - 1] \tag{4.11.20}$$

4.12 Finite rotations in terms of Euler parameters

In order to put equations (4.11.6) and (4.11.8) in purely algebraic form, let us define the so-called Euler parameters

$$\begin{aligned} e_0 &= \cos\frac{\phi}{2} \\ e_1 &= l_x \sin\frac{\phi}{2} \\ e_2 &= l_y \sin\frac{\phi}{2} \\ e_3 &= l_z \sin\frac{\phi}{2} \end{aligned} \tag{4.12.1}$$

The rotation operator may then be rewritten in the form

$$\boldsymbol{R} = \begin{bmatrix} 1-2(e_2^2+e_3^2) & 2(e_1e_2-e_0e_3) & 2(e_1e_3+e_0e_2) \\ 2(e_1e_2+e_0e_3) & 1-2(e_1^2+e_3^2) & 2(e_2e_3-e_0e_1) \\ 2(e_1e_3-e_0e_2) & 2(e_2e_3+e_0e_1) & 1-2(e_1^2+e_2^2) \end{bmatrix} \tag{4.12.2}$$

or, in a more compact form

$$\boldsymbol{R} = (2{e_0}^2-1)\boldsymbol{I} + 2\boldsymbol{e}\boldsymbol{e}^T + 2e_0\tilde{\boldsymbol{e}} \tag{4.12.3}$$

where the four parameters introduced are algebraic quantities which play equal roles. They are linked by the constraint

$$e_0^2+e_1^2+e_2^2+e_3^2 \;=\; 1 \tag{4.12.4}$$

Under this form, the problem of kinematic inversion does not give produce singularities. It is best performed by observing that the 4×4 symmetric matrix $\boldsymbol{S}$ obtained from the elements of $\boldsymbol{R}$

$$\boldsymbol{S} = \begin{bmatrix} 1+r_{11}+r_{22}+r_{33} & r_{32}-r_{23} & r_{13}-r_{31} & r_{21}-r_{12} \\ r_{32}-r_{23} & 1+r_{11}-r_{22}-r_{33} & r_{12}+r_{21} & r_{13}+r_{31} \\ r_{13}-r_{31} & r_{12}+r_{21} & 1-r_{11}+r_{22}-r_{33} & r_{32}+r_{23} \\ r_{21}-r_{12} & r_{13}+r_{31} & r_{32}+r_{23} & 1-r_{11}-r_{22}+r_{33} \end{bmatrix} \tag{4.12.5}$$

is a quadratic form of Euler parameters

$$\boldsymbol{S} = 4\begin{bmatrix} e_0^2 & e_0e_1 & e_0e_2 & e_0e_3 \\ e_0e_1 & e_1^2 & e_1e_2 & e_1e_3 \\ e_0e_2 & e_1e_2 & e_2^2 & e_2e_3 \\ e_0e_3 & e_1e_3 & e_2e_3 & e_3^2 \end{bmatrix} \tag{4.12.6}$$

The knowledge of one row of $\boldsymbol{S}$ allows us to calculate the parameters e_0 and $\boldsymbol{e}$. In practice, the algorithm is the following:

- determine k such that $s_{kk} = \max_j(s_{jj})$
- compute $e_k = \dfrac{1}{2}\sqrt{s_{kk}}$
- compute $e_i = \dfrac{s_{ik}}{4e_k} \qquad i = 0,1,2,3$

As it has been demonstrated, Euler parameters form a set of four dependent parameters: from a computational point of view, the redundancy contained in their definition is at first sight a drawback. However, Euler parameters possess also a certain number of properties which make their use very attractive.

- As it has been seen, the associated inversion procedure never gives rise to singular values;
- Euler parameters obey to quaternion algebra: as a consequence, successive finite rotations may be composed according to the quaternion multiplication rule (see appendix B);
- Euler parameters are purely algebraic quantities: finite rotations and their numerical inversion do not imply transcendental functions.

It is customary to use Euler parameters to describe the orientation of a robot effector. Therefore, they play an important role in trajectory generation methods.

A more detailed presentation of their properties is given in appendix B.

4.13 Rodrigues' parameters

Let us start again from the fact that the finite rotation of a vector about the origin may be described by the operation

$$\boldsymbol{s} \;=\; \boldsymbol{R}\,\boldsymbol{s}' \tag{4.13.1}$$

with $\boldsymbol{R}$ orthogonal, and that the length of the original vector is conserved

$$\boldsymbol{s}^T\boldsymbol{s} - \boldsymbol{s}'^T\boldsymbol{s}' = 0 \tag{4.13.2}$$

or

$$(\boldsymbol{s} - \boldsymbol{s}')^T(\boldsymbol{s} + \boldsymbol{s}') = 0 \tag{4.13.3}$$

Equation (4.13.3) means that the vectors $\boldsymbol{f}$ and $\boldsymbol{g}$ defined by

$$\begin{aligned} \boldsymbol{f} &= \boldsymbol{s} - \boldsymbol{s}' = (\boldsymbol{R} - \boldsymbol{I})\boldsymbol{s}' \\ \boldsymbol{g} &= \boldsymbol{s} + \boldsymbol{s}' = (\boldsymbol{R} + \boldsymbol{I})\boldsymbol{s}' \end{aligned} \tag{4.13.4}$$

are orthogonal together

$$\boldsymbol{f}^T\boldsymbol{g} = 0 \tag{4.13.5}$$

Let us next eliminate $\boldsymbol{s}'$ between both equations (4.13.4). One obtains

$$\boldsymbol{f} = (\boldsymbol{R} - \boldsymbol{I})(\boldsymbol{R} + \boldsymbol{I})^{-1}\boldsymbol{g} = \boldsymbol{B}\boldsymbol{g} \tag{4.13.6}$$

where $\boldsymbol{B}$ is necessarily of antisymmetric type since (4.13.6) yields to

$$\boldsymbol{g}^T\boldsymbol{B}\boldsymbol{g} = 0 \tag{4.13.7}$$

Let us express this property in terms of the vector $\boldsymbol{b}^T = [b_1\ b_2\ b_3]$ collecting the components of the matrix

$$\boldsymbol{B} = \tilde{\boldsymbol{b}} = \begin{bmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{bmatrix} \tag{4.13.8}$$

Equation (4.13.6) shows that the rotation operator is such that

$$\boldsymbol{R} - \boldsymbol{I} = \tilde{\boldsymbol{b}}(\boldsymbol{R} + \boldsymbol{I}) \tag{4.13.9}$$

or, if we solve with respect to $\boldsymbol{R}$

$$\boldsymbol{R} = (\boldsymbol{I} - \tilde{\boldsymbol{b}})^{-1}\ (\boldsymbol{I} + \tilde{\boldsymbol{b}}) \tag{4.13.10}$$

which corresponds to a specific choice of the rotation parameters.

Let us calculate explicitly

$$\Delta = dtm(\boldsymbol{I} - \tilde{\boldsymbol{b}}) = 1 + \boldsymbol{b}^T\boldsymbol{b} \tag{4.13.11}$$

and

$$(\boldsymbol{I} - \tilde{\boldsymbol{b}})^{-1} \ = \ \Delta^{-1}\ [I + \boldsymbol{b}\boldsymbol{b}^T + \tilde{\boldsymbol{b}}] \tag{4.13.12}$$

Performing then the product (4.13.9) and using the fact that

$$\tilde{\boldsymbol{b}}\ \tilde{\boldsymbol{b}} \ = \ \boldsymbol{b}\boldsymbol{b}^T \ - \ \boldsymbol{b}^T\boldsymbol{b}\ \boldsymbol{I} \tag{4.13.13}$$

provides the final expression

$$\boldsymbol{R} = \boldsymbol{I} + \frac{2}{1 + \boldsymbol{b}^T\boldsymbol{b}}(\tilde{\boldsymbol{b}} + \tilde{\boldsymbol{b}}\tilde{\boldsymbol{b}}) \tag{4.13.14}$$

Comparing the results (4.12.3) and (4.13.14), it is easy to check that they are equivalent, provided that one takes

$$b_1 = \frac{e_1}{e_0} = l_x \tan\frac{\phi}{2}$$
$$b_2 = \frac{e_2}{e_0} = l_y \tan\frac{\phi}{2}$$
$$b_3 = \frac{e_3}{e_0} = l_z \tan\frac{\phi}{2}$$

The corresponding vector has the norm

$$|\boldsymbol{b}| = \tan\frac{\phi}{2} \tag{4.13.15}$$

Taking Rodrigues' parameters in place of Euler parameters offers the advantage of using 3 independent quantities out of 4 linked by a normality condition. However, they give rise to a singularity when $\phi = \pm\pi$ and have thus to be used with caution.

4.14 Translation and angular velocities

Let $\boldsymbol{V}$ be a rigid body undergoing motion (Figure 4.3.1), and let us calculate the velocity of an arbitrary point P attached to it. In the absolute frame $Oxyz$, its position vector has the cartesian coordinates

$$\boldsymbol{p} \ = \ \boldsymbol{r} \ + \ \boldsymbol{R}\boldsymbol{p}' \tag{4.14.1}$$

Where the rotation matrix $\boldsymbol{R}$ may be described using any of the parameter sets just described above.

The components of the velocity vector at point P are obtained through time differentiation

$$\dot{\boldsymbol{p}} \ = \ \dot{\boldsymbol{r}} + \dot{\boldsymbol{R}}\boldsymbol{p}' + \boldsymbol{R}\dot{\boldsymbol{p}}' \tag{4.14.2}$$

where

- $\dot{\boldsymbol{r}}$ is the velocity vector of the reference point O',
- $\dot{\boldsymbol{p}}' = 0$: body $\boldsymbol{V}$ is assumed rigid.

The velocity vector is thus given by

$$\dot{\boldsymbol{p}} \;=\; \dot{\boldsymbol{r}} + \dot{\boldsymbol{R}}\boldsymbol{p}' \tag{4.14.3}$$

and its expression may still be modified by writing it in terms of absolute quantities. Let us invert (4.14.1) in the form

$$\boldsymbol{p}' \;=\; \boldsymbol{R}^T(\boldsymbol{p} - \boldsymbol{r}) \tag{4.14.4}$$

Equation (4.14.3) may then be rewritten in the form

$$\dot{\boldsymbol{p}} \;=\; \dot{\boldsymbol{r}} + \dot{\boldsymbol{R}}\boldsymbol{R}^T(\boldsymbol{p} - \boldsymbol{r}) \tag{4.14.5}$$

It is easy to observe that (4.14.5) is the matrix equivalent of the vector equation for velocities

$$\frac{d\vec{p}}{dt} \;=\; \frac{d\vec{r}}{dt} + \vec{\omega} \times (\vec{p} - \vec{r}) \tag{4.14.6}$$

where ω is the angular velocity of the frame $O'x'y'z'$ relatively to $Oxyz$.

Indeed, the orthonormality property $\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$ implies

$$\dot{\boldsymbol{R}}\boldsymbol{R}^T + \boldsymbol{R}\dot{\boldsymbol{R}}^T \;=\; \boldsymbol{O} \tag{4.14.7}$$

from which one deduces that the matrix

$$\dot{\boldsymbol{R}}\boldsymbol{R}^T \;=\; -\,\boldsymbol{R}\dot{\boldsymbol{R}}^T \tag{4.14.8}$$

is skew symmetric.

One deduces thus *the angular velocity matrix*

$$\tilde{\omega} \;=\; \dot{\boldsymbol{R}}\boldsymbol{R}^T \;=\; \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{4.14.9}$$

where ω_x, ω_y and ω_z are the cartesian components of ω in the frame $Oxyz$.

Equation (4.14.5) takes thus the final form

$$\dot{\boldsymbol{p}} \;=\; \dot{\boldsymbol{r}} + \tilde{\omega}(\boldsymbol{p} - \boldsymbol{r}) \tag{4.14.10}$$

which is obviously the matrix analog of (4.14.6).

4.15 Explicit expression for angular velocities

To the skew-symmetric matrix (4.14.9) let us associate the uni-column matrix ω of angular velocities

$$\omega^T \;=\; [\,\omega_x\ \omega_y\ \omega_z\ \,] \tag{4.15.1}$$

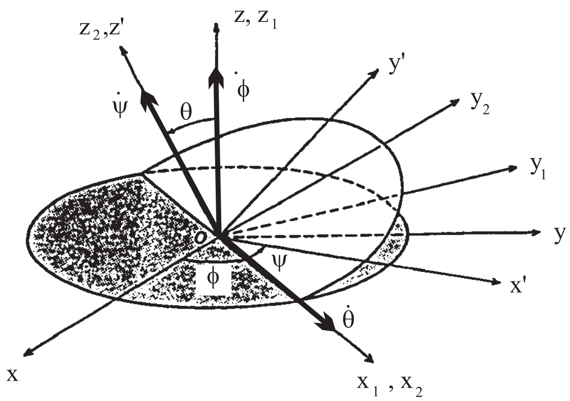


Figure 4.15.1: Angular velocities in terms of time derivatives of Euler angles

4.15.1 In terms of Euler Angles

In terms of Euler angles, the angular velovity vector results from the superposition of three angular velocities (Figure 4.15.1)

- $\dot{\phi}$ about the O_z axis,
- $\dot{\theta}$ about the Ox_1 axis,
- $\dot{\psi}$ about the Oz_2 axis.

The resulting angular velocity is thus given by

$$\omega \; = \; \begin{bmatrix} 0 & 0 & \dot{\phi} \end{bmatrix}^T \; + \; \boldsymbol{R}(z,\phi) \; \begin{bmatrix} \dot{\theta} & 0 & 0 \end{bmatrix}^T \; + \; \boldsymbol{R}(z,\phi) \; \boldsymbol{R}(x,\theta) \; \begin{bmatrix} 0 & 0 & \dot{\psi} \end{bmatrix}^T$$

The resulting expression of angular velocities is

$$\omega \; = \; \begin{bmatrix} \dot{\theta}\,\cos\phi + \dot{\psi}\,\sin\phi\,\sin\theta \\ \dot{\theta}\,\sin\phi - \dot{\psi}\,\cos\phi\,\sin\theta \\ \dot{\phi} + \dot{\psi}\,\cos\theta \end{bmatrix} \tag{4.15.2}$$

4.15.2 In terms of Bryant angles

A similar reasoning leads to a decomposition of the angular velocity into

- $\dot{\psi}$ about the O_z axis,
- $\dot{\theta}$ about the Oy_1 axis,
- $\dot{\phi}$ about the Ox_2 axis.

The resulting angular velocity is

$$\omega \; = \; \begin{bmatrix} -\dot{\theta}\,\sin\psi + \dot{\phi}\,\cos\psi\,\cos\theta \\ \dot{\theta}\,\cos\psi + \dot{\phi}\,\sin\psi\,\cos\theta \\ \dot{\psi} - \dot{\phi}\,\sin\theta \end{bmatrix} \tag{4.15.3}$$

4.15.3 In terms of Euler parameters

Proof of the result is given in appendix. It presents the remarkable property to provide a bilinear expression in Euler parameters and their time derivatives:

$$\omega = 2 \begin{bmatrix} -e_1 & e_0 & -e_3 & e_2 \\ -e_2 & e_3 & e_0 & -e_1 \\ -e_3 & -e_2 & e_1 & e_0 \end{bmatrix} \begin{bmatrix} \dot{e}_0 \\ \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} \tag{4.15.4}$$

4.16 Infinitesimal displacement

The concept of infinitesimal displacement allows us to evaluate small position changes at a point P (Figure 4.3.1) due to small variations in position $\boldsymbol{r}$ of reference point and orientation $\boldsymbol{R}$ of the body frame $O'x'y'z'$.

The variation of (4.14.1) provides the expression

$$\delta \boldsymbol{p} = \delta \boldsymbol{r} + \delta \boldsymbol{R}\, \boldsymbol{p}' + \boldsymbol{R}\, \delta \boldsymbol{p}' \tag{4.16.1}$$

The rigid body assumption allows us to omit the last term of (4.16.1), and taking account of (4.14.5) provides the expression

$$\delta \boldsymbol{p} = \delta \boldsymbol{r} + \delta \boldsymbol{R}\boldsymbol{R}^T (\boldsymbol{p} - \boldsymbol{r})$$

Invoking again the orthonormality property of $\boldsymbol{R}$

$$\delta \boldsymbol{R}\boldsymbol{R}^T + \boldsymbol{R}\delta \boldsymbol{R}^T = \boldsymbol{O}$$

tells us that the skew-symmetric matrix

$$\delta\tilde{\alpha} = \delta \boldsymbol{R}\, \boldsymbol{R}^T = \begin{bmatrix} 0 & -\delta\alpha_z & \delta\alpha_y \\ \delta\alpha_z & 0 & -\delta\alpha_x \\ -\delta\alpha_y & \delta\alpha_x & 0 \end{bmatrix} \tag{4.16.2}$$

describes the infinitesimal rotation of $O'x'y'z'$ related to $Oxyz$.

The infinitesimal displacement $\delta \boldsymbol{p}$ may thus be written in a form analog to velocities

$$\delta \boldsymbol{p} = \delta \boldsymbol{r} + \delta\tilde{\alpha}\, (\boldsymbol{p} - \boldsymbol{r}) \tag{4.16.3}$$

4.17 Accelerations

Expressing the dynamics of a manipulator and planning its trajectory involve necessarily to compute the accelerations resulting from the motion.

In order to express the acceleration at an arbitrary point P of a body $\boldsymbol{V}$ undergoing rigid body motion, let us derive (4.13.13) twice with respect to time

$$\ddot{\boldsymbol{p}} = \ddot{\boldsymbol{r}} + \ddot{\boldsymbol{R}}\boldsymbol{p}' \tag{4.17.1}$$

or, if use is made of (4.14.4)

$$\ddot{\boldsymbol{p}} = \ddot{\boldsymbol{r}} + \ddot{\boldsymbol{R}}\boldsymbol{R}^T (\boldsymbol{p} - \boldsymbol{r}) \tag{4.17.2}$$

It is straightforward to obtain the meaning of matrix $\ddot{\boldsymbol{R}}\boldsymbol{R}^T$ by deriving the expression (4.14.9) of the angular velocity matrix ω :

$$\begin{aligned}\dot{\tilde{\omega}} &= \frac{d}{dt}(\tilde{\omega}) = \ddot{\boldsymbol{R}}\boldsymbol{R}^T + \dot{\boldsymbol{R}}\dot{\boldsymbol{R}}^T \\ &= \ddot{\boldsymbol{R}}\boldsymbol{R}^T + \tilde{\omega}\tilde{\omega}^T\end{aligned}$$

or

$$\ddot{\boldsymbol{R}}\boldsymbol{R}^T = \dot{\tilde{\omega}} - \tilde{\omega}\,\tilde{\omega}^T \tag{4.17.3}$$

where the first term represents the matrix of angular accelerations

$$\dot{\tilde{\omega}} = \begin{bmatrix} 0 & -\dot{\omega}_z & \dot{\omega}_y \\ \dot{\omega}_z & 0 & -\dot{\omega}_x \\ -\dot{\omega}_y & \dot{\omega}_x & 0 \end{bmatrix} \tag{4.17.4}$$

and the second one, the matrix of centrifugal accelerations

$$\begin{aligned}\tilde{\omega}\;\tilde{\omega}^T &= \omega^2\,\boldsymbol{I} - \omega\;\omega^T \\ &= \begin{bmatrix} \omega_y^2 + \omega_z^2 & -\omega_x\omega_y & -\omega_x\omega_z \\ -\omega_x\omega_y & \omega_x^2 + \omega_z^2 & -\omega_y\omega_z \\ -\omega_x\omega_z & \omega_y\omega_z & \omega_x^2\omega_y^2 \end{bmatrix}\end{aligned} \tag{4.17.5}$$

Substitution of (4.17.3) into (4.17.2) provides the expression of accelerations

$$\ddot{\boldsymbol{p}} = \ddot{\boldsymbol{r}} + (\dot{\tilde{\omega}} - \tilde{\omega}\tilde{\omega}^T)(\boldsymbol{p} - \boldsymbol{r}) \tag{4.17.6}$$

4.18 Screw or helicoidal motion

It is worthwhile noticing that the differential motion of a rigid body as described by equation (4.14.5) may be interpreted as *screw motion*, i.e. the combination of a translation and a rotation *about a same axis* $\boldsymbol{s}$ with arbitrary orientation in space.

To this purpose, let us determine the locus of points P having a velocity vector $\dot{\boldsymbol{p}}$ that is colinear to the angular velocity ω vector. The problem is to find a scalar σ such that

$$\dot{\boldsymbol{p}} = \sigma\,\omega = \tilde{\omega}\,(\boldsymbol{p} - \boldsymbol{d}) + \dot{\boldsymbol{d}} \tag{4.18.1}$$

Premultiplying by ω^T provides the expression

$$\sigma = \frac{1}{\omega^2}\omega^T\dot{\boldsymbol{d}} \tag{4.18.2}$$

and the locus itself is obtained by solving (4.18.1) with respect to $\boldsymbol{p}$. To this purpose, let us examine the properties of the solution of the linear solutions of

$$\tilde{\boldsymbol{a}}\,\boldsymbol{x} = \boldsymbol{b} \tag{4.18.3}$$

It is obvious that a solution to (4.18.3) exists if and only if

$$\boldsymbol{a}^T\,\boldsymbol{b} = 0$$

particular solution is obtained through premultiplication of (4.18.3) by the system matrix $\tilde{\boldsymbol{a}}$

$$\tilde{\boldsymbol{a}}\,\tilde{\boldsymbol{a}}\,\boldsymbol{x} \;=\; (\boldsymbol{a}\,\boldsymbol{a}^T - ||a||^2\boldsymbol{I})\,\boldsymbol{x} \;=\; \tilde{\boldsymbol{a}}\,\boldsymbol{b} \tag{4.18.4}$$

Let us try a solution of the form

$$\boldsymbol{x} \;=\; k\,\tilde{\boldsymbol{a}}\,\boldsymbol{b} + \mu\boldsymbol{a}$$

Its direct substitution into (4.18.4) gives

$$-k||a||^2\,\tilde{\boldsymbol{a}}\boldsymbol{b} \;=\; \tilde{\boldsymbol{a}}\boldsymbol{b}$$

or $k \;=\; \frac{-1}{||a||^2}$ and the general solution to (4.18.3) is thus

$$\boldsymbol{x} \;=\; \frac{-1}{||a||^2}\,\tilde{\boldsymbol{a}}\boldsymbol{b} \;+\; \mu\boldsymbol{a} \tag{4.18.5}$$

Let us apply the result of (4.18.5) to (4.18.1) rewritten in the form

$$\tilde{\omega}\,(\boldsymbol{p} - \boldsymbol{d}) \;=\; \dot{\boldsymbol{p}} - \dot{\boldsymbol{d}} \tag{4.18.6}$$

Its solution is

$$\boldsymbol{p} \;=\; \boldsymbol{d} - \frac{1}{\omega^2}\,\tilde{\omega}\,(\dot{\boldsymbol{p}} - \dot{\boldsymbol{d}}) + \mu\,\omega$$

or, after noticing that (4.18.1) involves $\tilde{\omega}\,\dot{\boldsymbol{p}} \;=\; \boldsymbol{O}$, one has

$$\boldsymbol{p} \;=\; \boldsymbol{d} \;+\; \frac{1}{\omega^2}\tilde{\omega}\dot{\boldsymbol{d}} \;+\; \mu\,\omega \tag{4.18.7}$$

Equation (4.18.7) describes the locus of points having a velocity vector parallel to ω. It is a straight line $\boldsymbol{s}$ with direction ω defined by the parameter μ, as it can be very simply observed by noticing that two distinct points P_1 and P_2 are such that

$$\boldsymbol{p}_2 - \boldsymbol{p}_1 \;=\; (\mu_2 - \mu_1)\omega \tag{4.18.8}$$

Henceforth we express the locus in the form

$$\boldsymbol{s} \;=\; \boldsymbol{d} \;+\; \frac{1}{\omega^2}\,\tilde{\omega}\,\dot{\boldsymbol{d}} \;+\; \mu\omega \tag{4.18.9}$$

By premultiplying (4.18.9) by $\tilde{\omega}$, we deduce that

$$\begin{aligned}\tilde{\omega}(\boldsymbol{s} - \boldsymbol{d}) \;&=\; \frac{1}{\omega^2}\,\tilde{\omega}\,\tilde{\omega}\,\dot{\boldsymbol{d}} \\ &=\; \frac{1}{\omega^2}\,(\omega\,\omega^T - \omega^2\;\boldsymbol{I})\;\dot{\boldsymbol{d}}\end{aligned} \tag{4.18.10}$$

or equivalently

$$\dot{\boldsymbol{d}} - \tilde{\omega}\boldsymbol{d} \;=\; \sigma\omega - \tilde{\omega}\boldsymbol{s} \tag{4.18.11}$$

Let us finally substitute (4.18.11) into (4.18.1), it provides the following expression for the *velocity at an arbitrary point P*

$$\dot{\boldsymbol{p}} \;=\; \sigma\,\omega + \tilde{\omega}\;(\boldsymbol{p} - \boldsymbol{s}) \tag{4.18.12}$$

It is easily seen that it corresponds to a screw motion characterized by

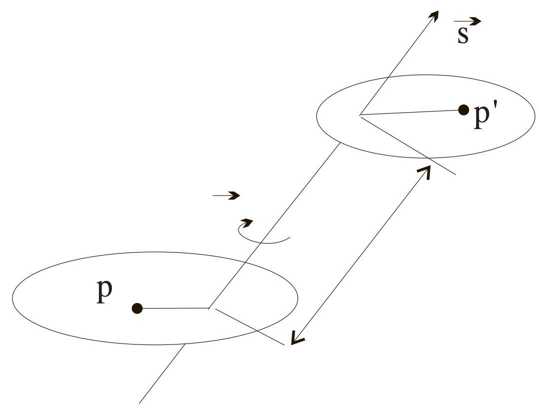


Figure 4.18.1: Screw motion

- A first component of translational velocity parallel to the rotation axis ω with a *pitch velocity* given by (4.18.1);
- A second component resulting from angular motion with rotational velocity ω around the straight line $\boldsymbol{s}$ which can thus be interpreted as the *screw axis* of the *helicoidal motion* (Figure 4.18.1).

The helicoidal representation of motion is the generalization to 3-D motion of the well known concept of instantaneous center of rotation: the center of rotation is replaced by the rotation axis, and the rotation is complemented by a translation along the rotation axis.

4.19 Homogeneous representation of vectors

The homogeneous components of a vector $\boldsymbol{r}$ are the four scalar quantities obtained by adding to the three cartesian components $[x\ y\ z]$ a scaling factor.

In homogeneous form, vectors are thus represented in a 4-D space; the 3-D representation may be restored by an particular projection.

Two cases have to be distinguished : the homogeneous representation of a *bound vector* such as the position vector of a point, and that of a *free vector* such as the velocity or the displacement at a given point.

As an example, let us consider the position vector $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ of two point P_1 and P_2 (Figure 4.19.1).
In homogeneous coordinates, a bound vector is presented by the 4-components row matrix

$$\boldsymbol{p}^T = [\, x^\star\ y^\star\ z^\star\ w^\star \,] \tag{4.19.1}$$

where $x^\star$, $y^\star$, $z^\star$ and $w^\star$ are related to the cartesian components (x, y, z) of the 3-D representation by

$$x = \frac{x^\star}{w^\star} \quad , \quad y = \frac{y^\star}{w^\star} \quad , \quad z = \frac{z^\star}{w^\star} \tag{4.19.2}$$

The parameter $w^\star$ is a scaling factor. One notes in particular that

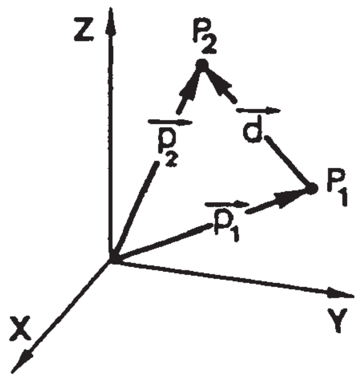


Figure 4.19.1: Position and displacement vectors

- the homogeneous representation is not affected by a change in scale

$$\boldsymbol{p} \;=\; \begin{bmatrix} x^\star \\ y^\star \\ z^\star \\ w^\star \end{bmatrix} \;=\; \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

  and it is thus customary to represent a bound vector in homogeneous notation by just *adding a fourth component equal to 1*, leaving then the three first components unchanged;

- the origin of the coordinate system is represented by the vector

$$\boldsymbol{u}^T \;=\; [\,0\;0\;0\;1\,] \qquad (4.19.3)$$

Let us calculate next the free vector representing the displacement from P_1 to P_2

$$\boldsymbol{d} \;=\; \boldsymbol{p}_2 - \boldsymbol{p}_1 \;=\; \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \\ 0 \end{bmatrix} \qquad (4.19.4)$$

One notes thus that a *free vector* is characterized by the fact that its *fourth component is zero*. In terms of homogeneous coordinates, it represents a point at infinity in the $\boldsymbol{d}$ direction.

4.20 Homogeneous representation of frame transformations

The advantage of using homogeneous representation of vectors both in computer-aided design and robotics lies in its ability to represent any frame transformation as an ordinary matrix product.

With homogeneous notation, the translation and rotation terms of the frame transformation rule

$$\boldsymbol{p} \;=\; \boldsymbol{r} + \boldsymbol{R}\boldsymbol{p}'$$

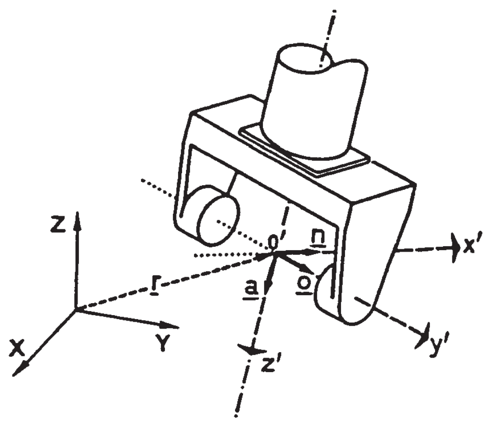


Figure 4.20.1: Tool frame transformation $\boldsymbol{A}$

may be gathered in a simple matrix transformation

$$\boldsymbol{p} = \boldsymbol{A}\boldsymbol{p}' \tag{4.20.1}$$

with the 4×4 matrix

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{r} \\ \boldsymbol{o}^T & 1 \end{bmatrix} \tag{4.20.2}$$

Its specific form expresses the fact that it generates rigid body motion. The last row collects necessarily three zeros and

- the upper 3×3 block represents the rotation, and is such that $\boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{I}$.

- the vector $\boldsymbol{r} = [r_x \; r_y \; r_z]^T$ in the last column represents the origin translation.

Although it is made of 16 terms, the transformation $\boldsymbol{A}$ depends on only six parameters :

- the three translations components r_x, r_y and r_z;

- the three rotation parameters contained in $\boldsymbol{R}$ (Euler parameters, Bryant angles, etc.).

Let us note that in robotics, it is convenient to build the transformation $\boldsymbol{A}$ describing the tool location and orientation in a given reference frame using the four following vectors (Figure 4.20.1):

- $\boldsymbol{r}$ describes the position of the reference point O' attached to the effector;
- the approach vector $\boldsymbol{a}$ is adopted as the local $\boldsymbol{z}$ axis;
- the orientation vector $\boldsymbol{o}$ is the local $\boldsymbol{y}$ axis;
- the normal vector $\boldsymbol{n}$, obtained by right-hand rule, gives the local $\boldsymbol{x}$ direction.

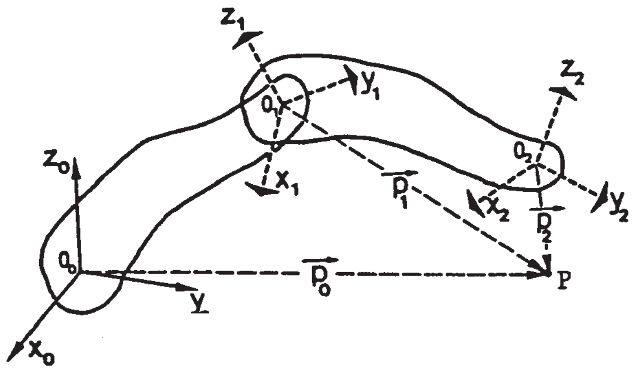


Figure 4.21.1: Successive homogeneous transformations

It is expressed in the form

$$\boldsymbol{A} = \begin{bmatrix} n_x & o_x & a_x & r_x \\ n_y & o_y & a_y & r_y \\ n_z & o_z & a_z & r_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.20.3}$$

In which the first three column are made of the unit vectors $\boldsymbol{n}$, $\boldsymbol{o}$ and $\boldsymbol{a}$ along the local axes $O'x'$, $O'y'$ and $O'z'$ and the last column describes the position of the effector in the $Oxyz$ frame.

In term of the vectors $\boldsymbol{n}$, $\boldsymbol{o}$ and $\boldsymbol{a}$, the orthonormality property of $\boldsymbol{R}$ can be expressed by the six scalar constraints :

$$\begin{aligned} \boldsymbol{a}\boldsymbol{a}^T &= \boldsymbol{n}^T\boldsymbol{n} = \boldsymbol{o}^T\boldsymbol{o} = 1 \\ \boldsymbol{a}^T\boldsymbol{o} &= \boldsymbol{n}^T\boldsymbol{o} = \boldsymbol{o}^T\boldsymbol{a} = 0 \end{aligned} \tag{4.20.4}$$

or alternatively, in terms of cross products

$$\tilde{\boldsymbol{a}}\,\boldsymbol{n} = \boldsymbol{o} \quad , \quad \tilde{\boldsymbol{n}}\,\boldsymbol{o} = \boldsymbol{a} \quad , \quad \tilde{\boldsymbol{o}}\,\boldsymbol{a} = \boldsymbol{n} \tag{4.20.5}$$

4.21 Successive homogeneous transformations

Let $\boldsymbol{p}_0$, $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ the position vector of a same point P in successive frames $O_0x_0y_0z_0$, $O_1x_1y_1z_1$, and $O_2x_2y_2z_2$ attached to different bodies of an articulated chain.

In a general manner, let us define

$$^{i}\boldsymbol{A}_j \tag{4.21.1}$$

as the homogeneous transformation matrix providing the cartesian components in frame $O_ix_iy_iz_i$ in terms of its components in $O_jx_jy_jz_j$.

In the example described above, we may thus write

$$\boldsymbol{p}_1 = {}^1\boldsymbol{A}_2\,\boldsymbol{p}_2 \tag{4.21.2}$$

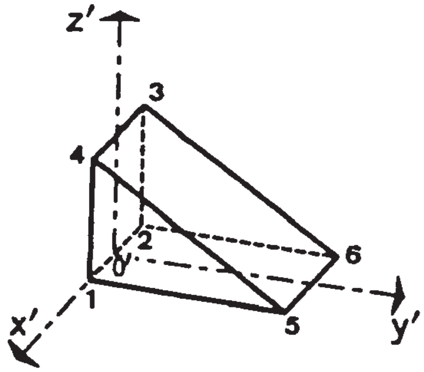


Figure 4.22.1: Geometric representation of a body $\boldsymbol{B}'$ in body coordinates

and similarly

$$\boldsymbol{p}_0 = {}^0\boldsymbol{A}_1\, \boldsymbol{p}_1 = {}^0\boldsymbol{A}_1\, {}^1\boldsymbol{A}_2\, \boldsymbol{p}_2 \tag{4.21.3}$$

The combined transformation

$$ {}^0\boldsymbol{A}_2 = {}^0\boldsymbol{A}_1 {}^1\boldsymbol{A}_2 \tag{4.21.4}$$

expresses thus the transformation from frame 2 to frame 0, and is obtained simply through matrix multiplication.

The same reasoning may be applied in a recursive manner to express that the successive transformations ${}^0\boldsymbol{A}_1$, ${}^1\boldsymbol{A}_2 \ldots {}^{n-1}\boldsymbol{A}_n$ generate the combined transformation

$$ {}^0\boldsymbol{A}_n = {}^0\boldsymbol{A}_1 . {}^1\boldsymbol{A}_2 \ldots {}^{n-1}\boldsymbol{A}_n \tag{4.21.5}$$

Equation (4.21.5) provides the combination law for successive homogeneous transformations.

In kinematic analysis of simply-connected open-tree structures, when the successive frame $O_i x_i y_i z_i$ are attached the successive links of the system, equation (4.21.5) provides an extremely compact representation of the geometrical model of the system.

4.22 Object manipulation in space

Let us represent the geometry of an object by the coordinates of a set of points given in a body frame. As an example, the prism of figure 4.22.1 may be described geometrically by an array $\boldsymbol{B}'$ containing the homogeneous coordinates of the corners (numbered from 1 to 6), expressed in a local frame

$$\boldsymbol{B}' = \begin{bmatrix} 1 & -1 & -1 & 1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 4 & 4 \\ 0 & 0 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{4.22.1}$$

Supposed that this object is located in the robot environment or has to be grasped by the tool. Any manipulation of this object or its description in another coordinate system can be simply described by a sequence of homogeneous transformations.

For example, suppose that it undergoes the following displacements:

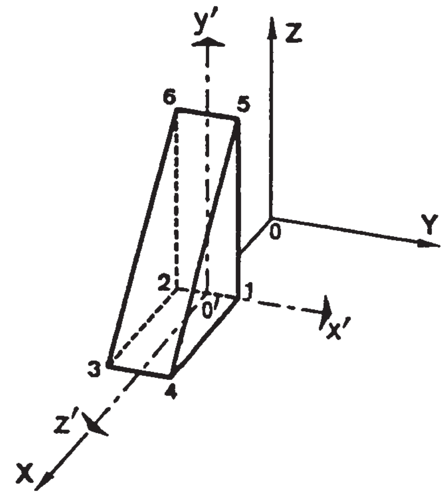


Figure 4.22.2: Object displacement expressed in global frame

- a translation of 4 units along the Ox reference axis;
- a rotation of 90° about Oy;
- a rotation of 90° about the new Oz axis.

If one notes

$$\boldsymbol{T}(r_x, r_y, r_z) \quad \text{and} \quad \boldsymbol{R}(\boldsymbol{n}, \phi) \tag{4.22.2}$$

the homogeneous transformation representing translation and rotation, the resulting transformation is :

$$\begin{aligned}\boldsymbol{A} &= \begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 1 & 4 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\end{aligned} \tag{4.22.3}$$

After displacement, body $\boldsymbol{B}'$ is geometrically described by the array

$$\boldsymbol{B} = \boldsymbol{A}\boldsymbol{B}' \tag{4.22.4}$$

or explicitly

$$\boldsymbol{B} = \begin{bmatrix} 4 & 4 & 6 & 6 & 6 & 4 \\ 1 & -1 & -1 & 1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 4 & 4 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

and its geometric configuration is that of figure 4.22.2. It is easy to verify that the new location is the result of the imposed motion.

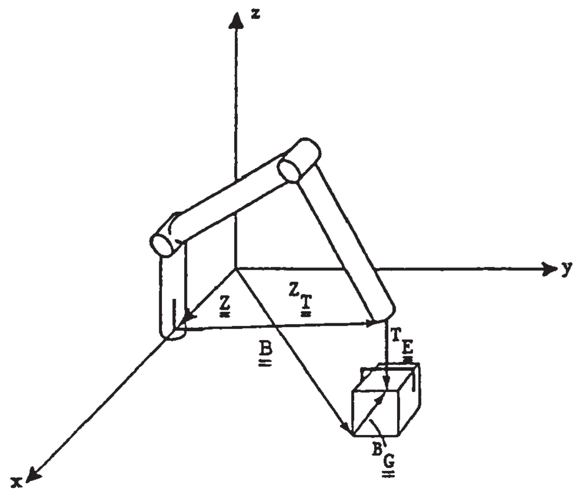


Figure 4.24.1: Closed loop of homogeneous transformations and task description

4.23 Inversion of homogeneous transformation

The inverse transformation to (4.20.1)

$$\boldsymbol{p}' = \boldsymbol{A}^{-1}\boldsymbol{p} \tag{4.23.1}$$

represents also a change in position and orientation without shape alteration. Therefore, it is also of the form (4.20.2)

$$\boldsymbol{A}^{-1} = \begin{bmatrix} \boldsymbol{R}' & \boldsymbol{r}' \\ \boldsymbol{o}^T & 1 \end{bmatrix} \tag{4.23.2}$$

and has to verify the relationship

$$\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I}$$

or explicitly

$$\begin{bmatrix} \boldsymbol{R} & \boldsymbol{r} \\ \boldsymbol{o}^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{R}' & \boldsymbol{r}' \\ \boldsymbol{o}^T & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}\boldsymbol{R}' & \boldsymbol{R}\boldsymbol{r}' + \boldsymbol{r} \\ \boldsymbol{o}^T & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{o} \\ \boldsymbol{o}^T & 1 \end{bmatrix}$$

Simple identification provides the results.

$$\begin{aligned} \boldsymbol{R}' &= \boldsymbol{R}^T \\ \boldsymbol{r}' &= -\boldsymbol{R}^T\,\boldsymbol{r} \end{aligned} \tag{4.23.3}$$

Its rotation part is the inverse of the rotation contained in $\boldsymbol{A}$, while the displacement has its sign reversed and is expressed in frame $O'x'y'z'$.

4.24 Closed loop of homogeneous transformation

The situation described by Figure 4.24.1 is a usual one in robotics.
Let us adopt $Oxyz$ as an absolute reference frame and define the following relative transformations:

- $\boldsymbol{Z}$: position and orientation of the manipulator support in $Oxyz$ frame;
- ${}^{Z}\boldsymbol{T}$: position and orientation of the wrist in frame $\boldsymbol{Z}$;
- ${}^{T}\boldsymbol{E}$: frame attached to end of tool relatively to wrist attachement;
- $\boldsymbol{B}$: location and orientation of body $\boldsymbol{B}$ in $Oxyz$ frame;
- ${}^{B}\boldsymbol{G}$: grasping transformation, describing location of grasping point and grasping orientation.

Figure 4.24.1 shows that two different points of view may be adopted to describe the grasping task:

- from the "robot" point of view, the grasping transformation is

$$\boldsymbol{E} = \boldsymbol{Z}.{}^{Z}\boldsymbol{T}.{}^{T}\boldsymbol{E} \tag{4.24.1}$$

- from the "world" point of view, the task to perform is

$$\boldsymbol{E} = \boldsymbol{B}.{}^{B}\boldsymbol{G} \tag{4.24.2}$$

Identifying the two points of view gives rise to the matrix identity

$$\boldsymbol{Z}.{}^{Z}\boldsymbol{T}.{}^{T}\boldsymbol{E} = \boldsymbol{B}.{}^{B}\boldsymbol{G} \tag{4.24.3}$$

Equation (4.24.3) can than be resolved with respect to any of its components. For example, given the task description and the location of the manipulator, the robot configuration may be computed from

$${}^{Z}\boldsymbol{T} = \boldsymbol{Z}^{-1}\,.\,\boldsymbol{B}\,.\,{}^{B}\boldsymbol{G}\,.\,\boldsymbol{E}^{-1} \tag{4.24.4}$$

Equivalent form of closed-loop equations such as (4.24.3) and (4.24.4) are all contained in a *transformation graph* (Figure 4.24.2) where

- each node represents a specific frame;
- each branch represents a direct transformation if traveled in the positive direction and its inverse if traveled in the negative direction.

On the diagram of Figure 4.24.2, one easily verifies

- equation (4.24.3), when traveling from the reference frame to the tool frame along two distinct paths;
- equation (4.24.4), when proceeding in the same manner from the manipulator base frame to the wrist frame.

The transformations $\boldsymbol{T}$ and $\boldsymbol{E}$ added on the diagram represent the wrist and the tool frames in absolute coordinates.

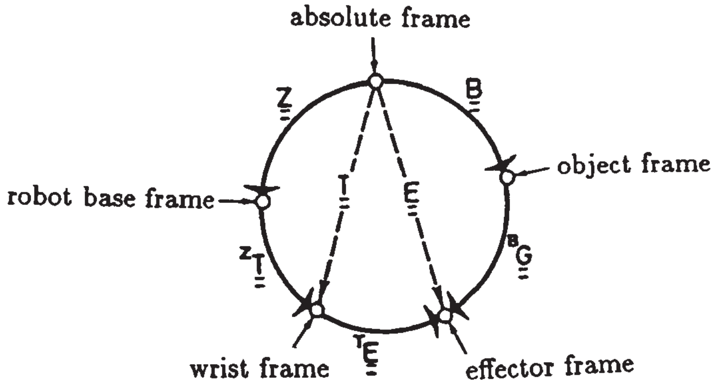


Figure 4.24.2: Graph representation of frame transformations

4.25 Homogeneous representation of velocities

It is possible, through time differentiation of equation (4.19.1), to express transformation and angular velocities of a given point P in a single matrix transformation. Let us suppose that P is attached to a rigid body so that its velocity $\dot{\boldsymbol{p}}'$ relatively to $O'x'y'z'$ is zero. Differentiation provides the free vector

$$\dot{\boldsymbol{p}} \;=\; [\,\dot{p}_x\;\dot{p}_y\;\dot{p}_z\;0\,]^T \;=\; \dot{\boldsymbol{A}}\,\boldsymbol{p}' \tag{4.25.1}$$

where

$$\dot{\boldsymbol{A}} \;=\; \begin{bmatrix} \dot{\boldsymbol{R}} & \dot{\boldsymbol{r}} \\ \boldsymbol{o}^T & 0 \end{bmatrix} \tag{4.25.2}$$

is the matrix of angular velocities.

It may still be expressed in the form

$$\dot{\boldsymbol{A}} \;=\; \dot{\boldsymbol{\Delta}}\,\boldsymbol{A} \tag{4.25.3}$$

where $\dot{\boldsymbol{\Delta}}$ is the *velocity differential operator* such that

$$\dot{\boldsymbol{p}} \;=\; \dot{\boldsymbol{\Delta}}\,\boldsymbol{p} \tag{4.25.4}$$

Equation (4.25.4) is the analog of (4.14.5) written in homogeneous coordinates. One finds explicitly

$$\dot{\boldsymbol{\Delta}} \;=\; \dot{\boldsymbol{A}}\,\boldsymbol{A}^{-1} \;=\; \begin{bmatrix} \tilde{\omega} & \boldsymbol{v} \\ \boldsymbol{o}^T & 0 \end{bmatrix} \tag{4.25.5}$$

where $\tilde{\omega}$ is the angular velocity matrix (4.14.6) and $\boldsymbol{v}$ represents the velocity of a point coinciding instantaneously with the origin of $Oxyz$: that is $\boldsymbol{p} = \mathbf{o}$. The velocity vector is thus

$$\boldsymbol{v} \;=\; \dot{\boldsymbol{r}} - \tilde{\omega}\,\boldsymbol{r}$$

4.26 Homogeneous representation of accelerations

A second differentiation of equation (4.23.3) provides the free vector of accelerations

$$\ddot{\boldsymbol{p}} \;=\; [\, \ddot{p}_x \; \ddot{p}_y \; \ddot{p}_z \; 0 \,]^T \;=\; \ddot{\boldsymbol{A}}\, \boldsymbol{p}' \tag{4.26.1}$$

where

$$\ddot{\boldsymbol{A}} \;=\; \begin{bmatrix} \ddot{\boldsymbol{R}} & \ddot{\boldsymbol{r}} \\ \boldsymbol{o}^T & 0 \end{bmatrix} \tag{4.26.2}$$

Matrix (4.26.2) may also be put in the form

$$\ddot{\boldsymbol{A}} \;=\; \ddot{\boldsymbol{\Delta}}\, \boldsymbol{A} \tag{4.26.3}$$

where the *acceleration differential operator* $\ddot{\boldsymbol{\Delta}}$ is such that

$$\ddot{\boldsymbol{p}} \;=\; \ddot{\boldsymbol{\Delta}} \boldsymbol{p} \tag{4.26.4}$$

One finds explicitly

$$\ddot{\boldsymbol{\Delta}} \;=\; \begin{bmatrix} \boldsymbol{B} & \boldsymbol{a} \\ \boldsymbol{o}^T & 0 \end{bmatrix} \tag{4.26.5}$$

where

$$\boldsymbol{B} \;=\; \ddot{\boldsymbol{R}}\boldsymbol{R}^T \;=\; \dot{\tilde{\omega}} - \tilde{\omega}\, \tilde{\omega}^T \tag{4.26.6}$$

collects angular and centrifugal accelerations and

$$\boldsymbol{a} \;=\; \ddot{\boldsymbol{r}} - (\dot{\tilde{\omega}} - \tilde{\omega}\, \tilde{\omega}^T)\; \boldsymbol{r} \tag{4.26.7}$$

represents the acceleration of a point attached to the body which coincides instantaneously to the origin of the $Oxyz$ frame.

Chapter 5

KINEMATICS OF SIMPLY CONNECTED OPEN-TREE STRUCTURES

5.1 Introduction

The kinematic model concept has already been introduced for planar kinematic structures. It expresses the algebraic relationship between *configuration space*, defined by the m joint variables $\boldsymbol{q}(t) = [q_1 \dots q_m]^T$ and *task space*, defined by a set of parameters $\boldsymbol{x}(t) = [x_1 \dots x_n]^T$ describing position, orientation and status of the effector.

The bilateral relationship

$$\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{q}) \tag{5.1.1}$$

existing between configuration space and task space is verified in a trivial manner when taken in the direct sense, but solving the inverse problem raises a certain number of difficulties such as condition of existence, its multiplicity and the effectiveness of the available methods to solve it.

The goal of this chapter is to proceed to the construction of the kinematic model for a $3-D$ manipulator with open-tree, simply connected mechanical structure. The type of joints considered are all of class 5 (one DOF) since only these are used in open-tree kinematic architectures.

The homogeneous transformation formalism is adopted as representation tool since it provides an easy expression of the successive transformations occurring in a kinematic chain.

At an elementary level, one will consider two distinct methods to express the geometric transformation introduced by one link within a kinematic chain.

- The most classical one is the *Hartenberg-Denavit* (DH) representation. It presents the advantage to describe most situations encountered in robot architectures while using a very limited number of geometric and displacement parameters;
- The *Sheth* method is much more general, but is not so flexible to use because of its large number of parameters.

We will describe both formalisms in order to indicate their respective limitations, but later on we will limit ourselves to the use of the DH notation.

It will next be shown how the homogeneous transformation formalism can be applied to solve the inverse kinematic problem in closed form, provided that the uncoupling condition between effector translation and orientation displacements is verified.

The rest of the chapter will be devoted to the construction of the differential model and its use.

5.2 Link description by Hartenberg-Denavit method

Let us consider a binary link of an articulated mechanism such as represented in Figure 5.2.1. It establishes a rigid connection between two successive joints numbered n and $n+1$, and their axes are supposed skew. Its *geometry* can be described very simply in terms of only two parameters:

- The distance a_n, measured along the common perpendicular to both axes;
- The twist angle α_n, defined as the angle between both joint axes.

If on the other hand, relative motion is restrained to joints of revolute, prismatic, cylindrical and skew types, the relative displacement occurring at joint n may also be described in terms of two parameters:

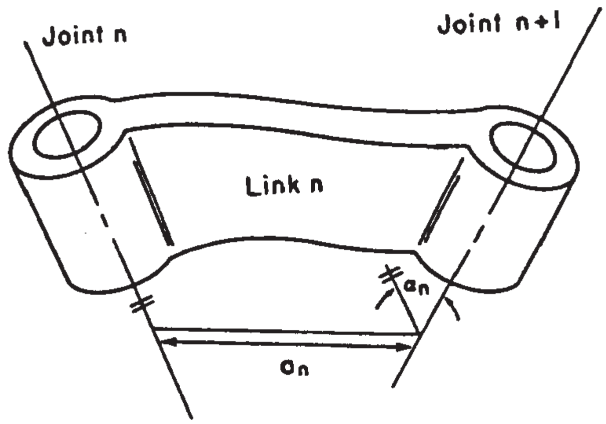


Figure 5.2.1: Hartenberg-Denavit representation of a binary link

- The rotation θ_n about the joint axis;
- The displacement d_n along the same axis.

The geometric parameters of the member

A link n is connected to, at most, two other links (i.e. link $n-1$ and link $n+1$). Thus two joint axes are established at both ends of the connection. The significance of links, from a kinematic perspective, is that they maintain a fixed configuration between the joints which can be characterized by two parameters a_n and α_n, which determine the structure of the link. They are defined as following:

- a_n, the length of the member, is the shortest distance measured along the common normal between the joint axes i.e. $\boldsymbol{z}_{n-1}$ and $\boldsymbol{z}_n$ axes for joint $n-1$ and n,
- α_n, the twist angle, is the angle between the joint axes measured in a plane perpendicular to common perpendicular.

The displacement parameters of the member

A joint axis establishes the connection between two links. This joint axis will have two normals connected to it, one for each link. The relative position of two such connected links is given by d_n which is the distance measured along the joint axis $\boldsymbol{z}_{n-1}$ between the normals. The joint angle θ_n between the normals is measured in a plane normal to the joint axis. Hence, the parameters d_n and θ_n are called distance and angle between adjacent links. They determine the relative position of neighboring links.

The different types of joints which can be represented correspond then to the following cases:

- *Revolute joint:* $\theta_n = \theta_n(t)$, while d_n is a fixed length.
- *Prismatic joint:* $d_n = d_n(t)$, while the angle θ_n remains constant.
- *Cylindrical joint:* $\theta_n = \theta_n(t)$ and $d_n = d_n(t)$ may vary independently.

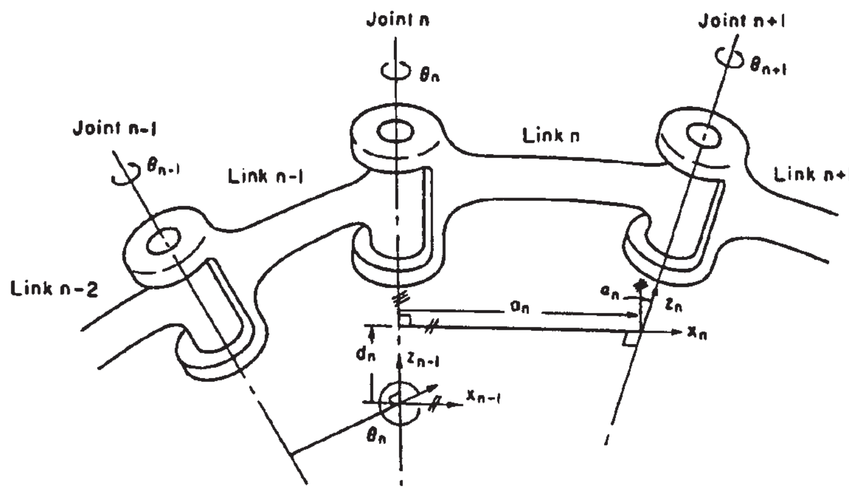


Figure 5.2.2: Hartenberg-Denavit frames on a kinematic chain

- *Screw joint:* $\theta_n = \theta_n(t)$ and $d_n = k_n\theta_n(t)$, where k_n is the constant pitch of the screw.

Considering next an articulated chain of such members, let us define a frame $Ox_ny_nz_n$ attached to link n as follows (Figure 5.2.2):

- Its origin O_n is located at the intersection of joint axis L_{n+1} with the common perpendicular to L_n and L_{n+1};
- The x_n axis is taken along the common perpendicular;
- The z_n axis coincides with joint axis L_{n+1};
- The y_n axis is defined by the cross product $\boldsymbol{y}_n = \boldsymbol{z}_n \times \boldsymbol{x}_n$.

The frame transformation ${}^{n-1}\boldsymbol{A}_n$ describing the finite motion from link n to link $n-1$ may then be expressed as the following sequence of elementary transformations, starting from link $n-1$:

1. A rotation θ_n about $\boldsymbol{z}_{n-1}$;
2. A translation d_n along the same axis $\boldsymbol{z}_{n-1}$;
3. A translation a_n along the $\boldsymbol{x}_n$ axis;
4. A rotation α_n about $\boldsymbol{x}_n$.

Steps 1 and 2 describe the relative displacement at joint n, while steps 3 and 4 describe the frame transformation associated to link geometry.

In matrix form

$$ {}^{n-1}\boldsymbol{A}_n = \boldsymbol{R}ot(\boldsymbol{z}_{n-1}, \theta_n) \cdot \boldsymbol{T}rans(\boldsymbol{z}_{n-1}, d_n) \cdot \boldsymbol{T}rans(\boldsymbol{x}_n, a_n) \cdot \boldsymbol{R}ot(\boldsymbol{x}_n, \alpha) \tag{5.2.1} $$

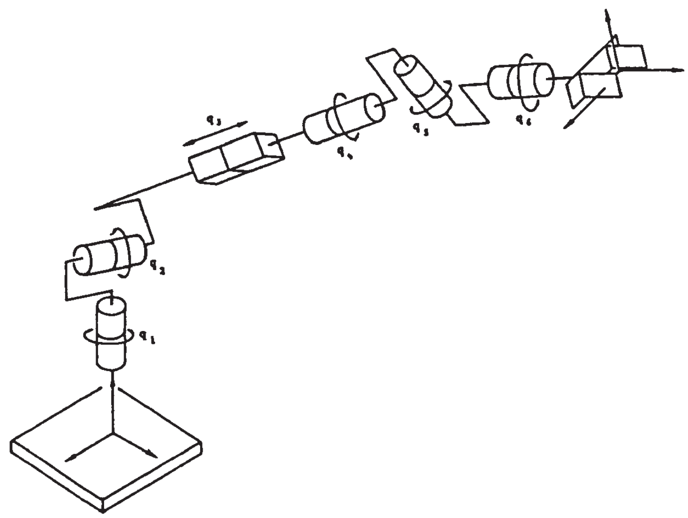


Figure 5.3.1: Open-tree simply connected structure

and its explicit calculation provides the expression

$$^{n-1}\boldsymbol{A}_n = \begin{bmatrix} \cos\theta_n & -\sin\theta_n\cos\alpha_n & \sin\theta_n\sin\alpha_n & a_n\cos\theta_n \\ \sin\theta_n & \cos\theta_n\cos\alpha_n & -\cos\theta_n\sin\alpha_n & a_n\sin\theta_n \\ 0 & \sin\alpha_n & \cos\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.2.2}$$

If the description of articulated structures is restricted to open-tree simply connected structures with joints of class 5, then the local frame transformation (5.2.2) may be expressed as a function of one single DOF

$$^{n-1}\boldsymbol{A}_n =^{n-1}\boldsymbol{A}_n(q_n) \tag{5.2.3}$$

5.3 Kinematic description of an open-tree simply connected structure

Let us denote by $\boldsymbol{T}$ the homogeneous transformation describing the position and orientation of the tool for an open-tree simply connected structure as represented in Figure 5.3.1.

On one hand, it results from the successive transformations introduced by the individual links. Each transformation involving only one DOF, one may write

$$\boldsymbol{T} =^0\boldsymbol{A}_1 \cdot^1\boldsymbol{A}_2 \ldots^{m-1}\boldsymbol{A}_m = \begin{bmatrix} t_{11}(q_1,\ldots q_m) & \ldots & \ldots & t_{14}(q_1,\ldots q_m) \\ t_{21}(q_1,\ldots q_m) & \ldots & \ldots & t_{24}(q_1,\ldots q_m) \\ t_{31}(q_1,\ldots q_m) & \ldots & \ldots & t_{34}(q_1,\ldots q_m) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.3.1}$$

On the other hand, the expression of $\boldsymbol{T}$ in task space is

$$\boldsymbol{T} = \begin{bmatrix} n_x & o_x & a_x & r_x \\ n_y & o_y & a_y & r_y \\ n_z & o_z & a_z & r_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{r} \\ \mathbf{o}^T & 1 \end{bmatrix} \tag{5.3.2}$$

where $\boldsymbol{n}$, $\boldsymbol{o}$, and $\boldsymbol{a}$ represent respectively the normal, approach and orientation vectors of the tool and $\boldsymbol{r}$, the location of its center.

In order to apply Hartenberg-Denavit's method to a simply connected open-tree structure, let us identify first the joint axes L_n and attach to them a variable σ_n describing their type: since one is limited to R and P joints, let us write for each joint axis L_n

$$\sigma_n = \begin{cases} 0 & \text{for a revolute joint} \\ 1 & \text{for a prismatic joint} \end{cases} \tag{5.3.3}$$

in which case the associated DOF is

$$q_n = (1 - \sigma_n)\theta_n + \sigma_n d_n \tag{5.3.4}$$

5.3.1 Link coordinate system assignment

The geometric kinematic model may then be obtained by applying the following recursive procedure.

For each link, when traveling from base to effector of the manipulator, the intermediate transformations ${}^{n-1}\boldsymbol{A}_n$ can be identified in four steps:

1. The origin O_n of frame $O_n x_n y_n z_n$ is the intersection of the common perpendicular to L_n and L_{n+1} with axis L_{n+1}. In the case of occurrence of two successive axes which are parallel or confused, the location of the common perpendicular has however to be chosen arbitrarily.

2. Axis $\boldsymbol{z}_n$ is defined by the unit vector along L_{n+1}: it is arbitrarily oriented.

3. Axis $\boldsymbol{x}_n$ is the unit vector along the common perpendicular to L_n and L_{n+1}, and directed towards L_{n+1}. When axes L_n and L_{n+1} intersect, then the choice of the $\boldsymbol{x}_n$ can not be determined in that way. From the definition of the Denavith-Hartenberg transformation, $\boldsymbol{z}_{n-1}$ comes into coincidence with $\boldsymbol{z}_n$ by a rotation around $\boldsymbol{x}_n$. So $\boldsymbol{x}_n$ is the common perpendicular direction to L_n and L_{n+1} and it comes that $\boldsymbol{x}_n$ must be always be chosen as the cross product $\boldsymbol{z}_{n-1} \times \boldsymbol{z}_n$.

4. Axis $\boldsymbol{y}_n$ is defined by the cross product rule $\boldsymbol{y}_n = \boldsymbol{z}_n \times \boldsymbol{x}_n$.

Remarks

- At the effector, the vector $\boldsymbol{z}_n$ and the joint axis L are oriented along the approach vector $\boldsymbol{a}$. However for planar manipulators, it is generally much simpler to create a virtual joint axis at the effector that is out of the plane of the manipulator than in the approach direction as for space manipulators.

- One is free to choose the location of the frame $O_0 x_0 y_0 z_0$ anywhere in the supporting base, as long as $\boldsymbol{z}_0$ axis lies along the axis of motion of the first joint.

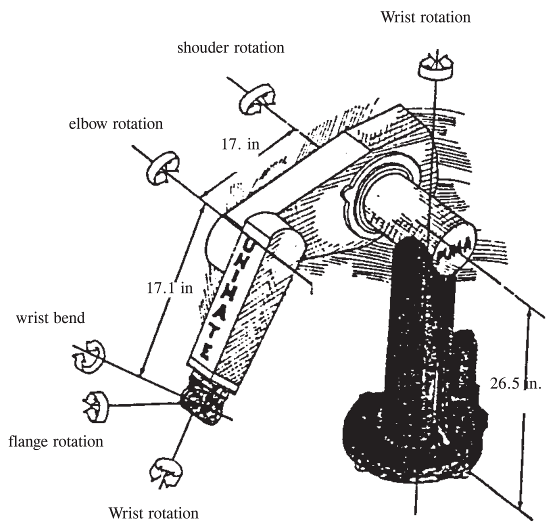


Figure 5.4.1: kinematic architecture of $6R$ type

- When joint axes of L_n and L_{n+1} are parallel, it is better to take O_n in such a way that the distance d_n is minimum (often zero) for the link for which the origin is defined unambiguously.
- At effector level (link n), the origin of the coordinate system is generally chosen identical to the origin of the link $n-1$. So ${}^0\boldsymbol{A}_6$ specifies the position and orientation of the end-point of the manipulator with respect to the base coordinate system.
- If the manipulator is related to a reference system by a transformation $\boldsymbol{B}$ and has a tool attached to its last joint's mounting plate described by $\boldsymbol{H}$, then the end-point of the tool can be related to the reference coordinates frame by

$$ {}^{ref}\boldsymbol{T}_{tool} = \boldsymbol{B}\ {}^0\boldsymbol{A}_6\ \boldsymbol{H} \tag{5.3.5} $$

5.4 Geometric model of the PUMA 560

Let us consider the structure of the PUMA robot described by Figure 5.4.1: it has an open-tree architecture of type $6R$.

Its kinematic architecture is represented schematically in Figure 5.4.2 in a specific configuration which we use as a reference. The figure displays

- The definition of the joint axes L_1 to L_7, L_7 being a supplementary axis defined at the end of link 6 and chosen coincident with the approach vector;

- The points O_i at frame origins which can be defined without ambiguity.

Figure 5.4.3 represents the frame vectors $\boldsymbol{x}_n$ and $\boldsymbol{z}_n$ which can be defined without ambiguity, except for their direction. It is easy to complete the frame definition by choosing the frame centers O_0, O_2 and O_6 and making an appropriate choice for the directions $\boldsymbol{x}_0$, $\boldsymbol{x}_4$ and $\boldsymbol{x}_6$ (Figure 5.4.4).

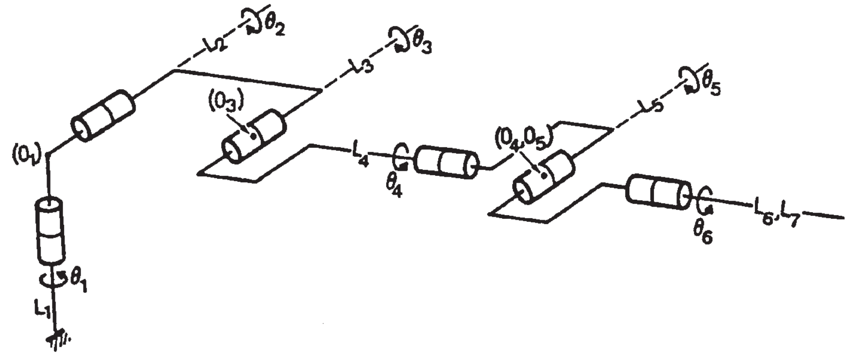


Figure 5.4.2: Definition of joint axes L_n and frame origins O_n

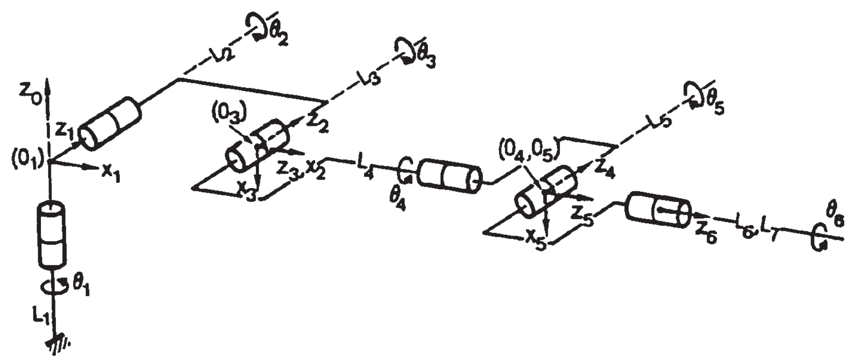


Figure 5.4.3: Location of axes $\boldsymbol{x}_n$ and $\boldsymbol{z}_n$

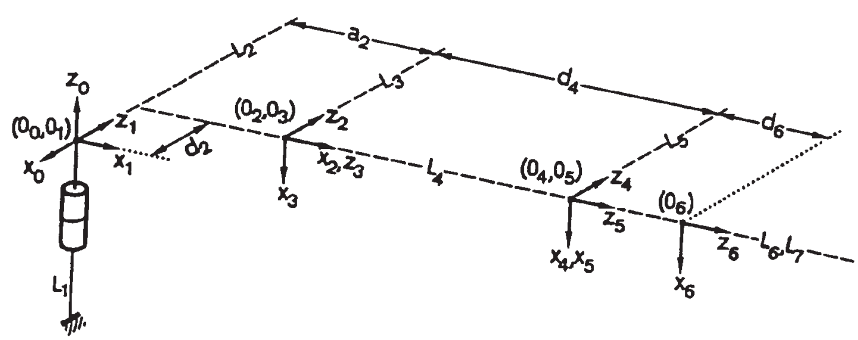


Figure 5.4.4: Complete definition of frames $O_nx_ny_nz_n$

Parameter	Link 1	Link 2	Link 3	Link 4	Link 5	Link 6
σ_n	0	0	0	0	0	0
α_n	$-\frac{\pi}{2}$	0	$\frac{\pi}{2}$	$-\frac{\pi}{2}$	$\frac{\pi}{2}$	0
a_n	0	$432mm$	0	0	0	0
θ_n	q_1	q_2	q_3	q_4	q_5	q_6
d_n	0	$149.5mm$	0	$432mm$	0	$56.5mm$
q_n value	$\frac{\pi}{2}$	0	$\frac{\pi}{2}$	0	0	0
Variation	$-160°$ $160°$	$-225°$ $45°$	$-45°$ $225°$	$-110°$ $170°$	$-100°$ $100°$	$-266°$ $266°$

Table 5.1: Hartenberg-Denavit parameters for the PUMA 560

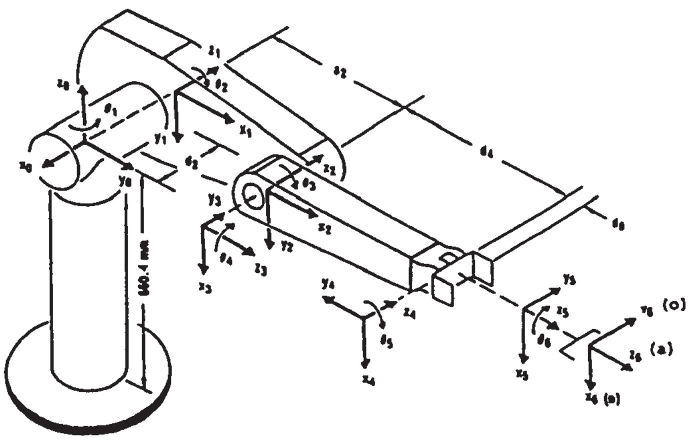


Figure 5.4.5: Hartenberg-Denavit representation of the PUMA 560

The $\boldsymbol{y}_n$ vectors are such that $\boldsymbol{y}_n = \boldsymbol{z}_n \times \boldsymbol{x}_n$ and are not represented in the figure. The corresponding parameters in Hartenberg-Denavit notation are given for the PUMA 560 by the Table 5.1.

The successive transformations have the following expressions

$$
{}^0\boldsymbol{A}_1 = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^1\boldsymbol{A}_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2C_2 \\ S_2 & C_2 & 0 & a_2S_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$
$$
{}^2\boldsymbol{A}_3 = \begin{bmatrix} C_3 & 0 & S_3 & 0 \\ S_3 & 0 & -C_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^3\boldsymbol{A}_4 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (5.4.1)
$$
$$
{}^4\boldsymbol{A}_5 = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^5\boldsymbol{A}_6 = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

To express the complete transformation

$$\boldsymbol{T} = {}^0\boldsymbol{A}_1 \cdot {}^1\boldsymbol{A}_2 \cdot {}^2\boldsymbol{A}_3 \cdot {}^3\boldsymbol{A}_4 \cdot {}^4\boldsymbol{A}_5 \cdot {}^5\boldsymbol{A}_6 \tag{5.4.2}$$

describing the effector configuration, we use the notations

$$\begin{aligned} C_{ij} &= \cos(\theta_i + \theta_j + \ldots) \\ S_{ij} &= \sin(\theta_i + \theta_j + \ldots) \end{aligned} \tag{5.4.3}$$

and we express the successive matrix products, starting from the effector, to obtain the intermediate matrices

$${}^4\boldsymbol{A}_6 = {}^4\boldsymbol{A}_5 \cdot {}^5\boldsymbol{A}_6 = \begin{bmatrix} C_5C_6 & -C_5S_6 & S_5 & S_5d_6 \\ S_5C_6 & -S_5S_6 & C_5 & C_5d_6 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.4.4}$$

$${}^3\boldsymbol{A}_6 = {}^3\boldsymbol{A}_4 \cdot {}^4\boldsymbol{A}_6 = \begin{bmatrix} C_4C_5C_6 - S_4S_6 & -C_4C_5S_6 - S_4C_6 & C_4S_5 & C_4S_5d_6 \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & S_4S_5 & S_4S_5d_6 \\ -S_5C_6 & S_5S_6 & C_5 & d_4 + C_5d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.4.5}$$

$$\begin{aligned} {}^2\boldsymbol{A}_6 &= {}^2\boldsymbol{A}_3 \cdot {}^3\boldsymbol{A}_6 \\ &= \begin{bmatrix} \begin{array}{c} C_3(C_4C_5C_6 - S_4S_6) \\ -S_3S_5C_6 \end{array} & \begin{array}{c} -C_3(C_4C_5S_6 + S_4C_6) \\ +S_3S_5S_6 \end{array} & C_3C_4S_5 + S_3C_5 & \begin{array}{c} C_3C_4S_5d_6 \\ +S_3(d_4 + C_5d_6) \end{array} \\ \begin{array}{c} S_3(C_4C_5C_6 - S_4S_6) \\ +C_3S_5C_6 \end{array} & \begin{array}{c} -S_3(C_4C_5S_6 + S_4C_6) \\ -C_3S_5S_6 \end{array} & S_3C_4S_5 - C_3C_5 & \begin{array}{c} S_3C_4S_5d_6 \\ -C_3(d_4 + C_5d_6) \end{array} \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & S_4S_5 & S_4S_5d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{5.4.6}$$

$$\begin{aligned} {}^1\boldsymbol{A}_6 &= {}^1\boldsymbol{A}_2 \cdot {}^2\boldsymbol{A}_6 \\ &= \begin{bmatrix} \begin{array}{c} C_{23}(C_4C_5C_6 - S_4S_6) \\ -S_{23}S_5C_6 \end{array} & \begin{array}{c} -C_{23}(C_4C_5S_6 + S_4C_6) \\ +S_{23}S_5S_6 \end{array} & \begin{array}{c} C_{23}C_4S_5 \\ +S_{23}C_5 \end{array} & \begin{array}{c} C_{23}C_4S_5d_6 \\ +S_{23}(d_4 + C_5d_6) \\ +a_2C_2 \end{array} \\ \begin{array}{c} S_{23}(C_4C_5C_6 - S_4S_6) \\ +C_{23}S_5C_6 \end{array} & \begin{array}{c} -S_{23}(C_4C_5S_6 + S_4C_6) \\ -C_{23}S_5S_6 \end{array} & \begin{array}{c} S_{23}C_4S_5 \\ -C_{23}C_5 \end{array} & \begin{array}{c} S_{23}C_4S_5d_6 + a_2S_2 \\ -C_{23}(d_4 + C_5d_6) \end{array} \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & S_4S_5 & S_4S_5d_6 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{5.4.7}$$

Let us note that in (5.4.7) the angles θ_2 and θ_3 are simply added since they correspond to rotations about parallel axes.

The final result is

$$\boldsymbol{T} = {}^0\boldsymbol{A}_1 \cdot {}^1\boldsymbol{A}_6 = \begin{bmatrix} n_x & o_x & a_x & r_x \\ n_y & o_y & a_y & r_y \\ n_z & o_z & a_z & r_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.4.8}$$

with the components

$$
\begin{aligned}
n_x &= C_1[C_{23}(C_4C_5C_6 - S_4S_6) - S_{23}S_5C_6] - S_1(S_4C_5C_6 + C_4S_6)\\
n_y &= S_1[C_{23}(C_4C_5C_6 - S_4S_6) - S_{23}S_5C_6] + C_1(S_4C_5C_6 + C_4S_6)\\
n_z &= -S_{23}(C_4C_5C_6 - S_4S_6) - C_{23}S_5C_6\\
o_x &= C_1[-C_{23}(C_4C_5S_6 + S_4C_6) + S_{23}S_5S_6] + S_1(S_4C_5S_6 - C_4C_6)\\
o_y &= S_1[-C_{23}(C_4C_5S_6 + S_4C_6) + S_{23}S_5S_6] - C_1(S_4C_5S_6 - C_4C_6)\\
o_z &= S_{23}(C_4C_5S_6 + S_4C_6) + C_{23}S_5S_6\\
a_x &= C_1(C_{23}C_4S_5 + S_{23}C_5) - S_1S_4S_5\\
a_y &= S_1(C_{23}C_4S_5 + S_{23}C_5) + C_1S_4S_5\\
a_z &= -S_{23}C_4S_5 + C_{23}C_5\\
r_x &= C_1[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] - S_1(S_4S_5d_6 + d_2)\\
r_y &= S_1[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] + C_1(S_4S_5d_6 + d_2)\\
r_z &= (-S_{23}C_4S_5 + C_{23}C_5)d_6 + C_{23}d_4 - a_2S_2
\end{aligned}
\tag{5.4.9}
$$

Note that in the introduction on robot technology, we have mentioned that a particular architecture where the wrist is made of three revolute joints with intersecting axes and orthogonal two by two provides decoupling between effector location and orientation. The point of intersection of the three joints at wrist level behaves like a virtual spherical joint and its location is determined by the only three displacements θ_1, θ_2 and θ_3.

The PUMA 560 architecture takes advantage of this interesting property. The virtual spherical joint may then be defined as the specific point obtained by setting a zero effector length

$$d_6 = 0 \tag{5.4.10}$$

in which case the formulas (5.4.9) describing the effector position simplify into

$$
\begin{aligned}
r'_x &= C_1(S_{23}d_4 + a_2C_2) - S_1d_2\\
r'_y &= S_1(S_{23}d_4 + a_2C_2) + C_1d_2\\
r'_z &= C_{23}d_4 - a_2S_2
\end{aligned}
\tag{5.4.11}
$$

where $\boldsymbol{r}'$ is the modified position vector

$$\boldsymbol{r}' = \boldsymbol{r} - d_6\ \boldsymbol{a} \tag{5.4.12}$$

The geometric model splits then into two sets of equations which we may formally write in the form

$$
\begin{aligned}
\boldsymbol{r}' &= \boldsymbol{f}\ (\theta_1, \theta_2, \theta_3)\\
\boldsymbol{R} &= \boldsymbol{R}\ (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6)
\end{aligned}
\tag{5.4.13}
$$

5.5 Link description using the Sheth method

The Hartenberg-Denavit method that we have just introduced to describe kinematic chains suffers from several drawbacks:

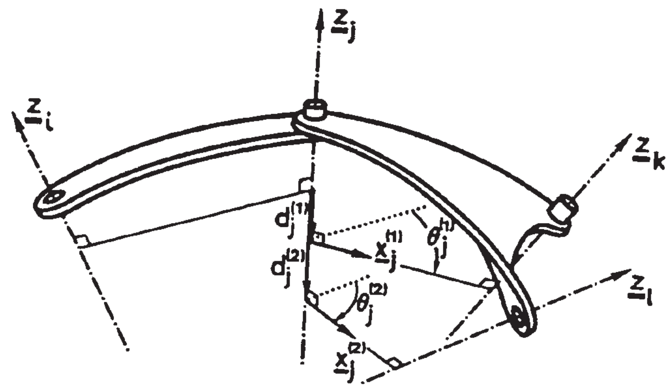


Figure 5.5.1: Description of ternary link by Hartenberg-Denavit's method

1. The successive coordinate axes are necessarily defined in such a way that the reference point O_n and the axis $\boldsymbol{x}_n$ are defined on the common perpendicular to adjacent link axes. From the point of view of data generation, this may render difficult the task of specifying the geometry of the link and its structural data.
2. The DH transformation mixes two types of information: α_n and a_n describe the link geometry while θ_n and d_n are motion-related quantities.
3. The DH notation cannot be extended to ternary links.

Let us take the example of a multiply-connected system, which involves then one or more ternary links (Figure 5.5.1). With the DH notation, the displacement parameters θ_j and d_j of the link cannot be defined in a unique manner: according to Figure 5.5.1 two distinct displacement parameters have to be introduced such that

$$\theta_j^{(2)} = \theta_j^{(1)} + \phi_j \tag{5.5.1}$$

$$d_j^{(2)} = d_j^{(1)} + \psi_j \tag{5.5.2}$$

where ϕ_j and ψ_j are additional geometric parameters describing the link. On the other hand, the geometric parameters α_j and a_j characterize the previous link. They are thus not modified.

Sheth's method overcomes the limitations due higher order links by introducing a number of frames equal to the number of joints on the link. At the same time, it provides more flexibility to specify the link geometry.

With Sheth's method, the geometry of the link (Figure 5.5.2) is specified by the position and orientation of frames attached to each joint. Figure 5.5.2 shows the case of a binary link where a first frame $\boldsymbol{u}_j\boldsymbol{v}_j\boldsymbol{w}_j$ is attached at the 'origin' of the link (joint j) and a second one, $\boldsymbol{x}_k\boldsymbol{y}_k\boldsymbol{z}_k$, to the 'end' (joint k). The expressions 'origin' and 'end' result simply from the traveling path in the kinematic chain.

To describe the geometry, one locates first the common perpendicular to both joint axes $\boldsymbol{w}_j$ and $\boldsymbol{z}_k$. One chooses then arbitrarily on it a positive direction with unit axis $\boldsymbol{t}_{jk}$.

Specifying the link geometry requires no less than 6 parameters which are however easily found as follows (Figure 5.5.2):

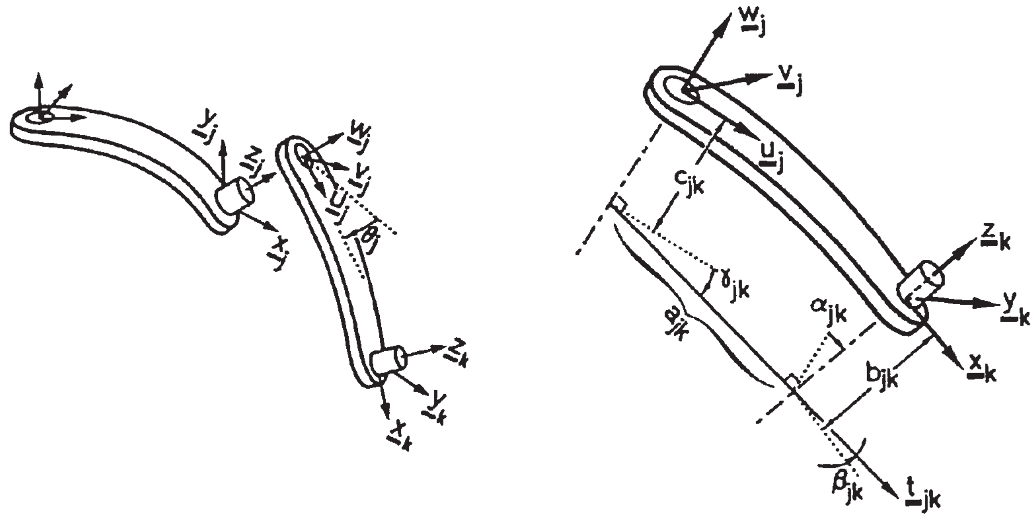


Figure 5.5.2: Sheth's description of a binary link: (a) definition of frames attached to the link; (b) definition of geometric parameters

- a_{jk} is the distance from $\boldsymbol{w}_j$ to $\boldsymbol{z}_k$ measured along $\boldsymbol{t}_{jk}$.
- b_{jk} is the distance from $\boldsymbol{t}_{jk}$ to $\boldsymbol{x}_k$ measured along $\boldsymbol{z}_k$.
- c_{jk} is the distance from $\boldsymbol{u}_j$ to $\boldsymbol{t}_{jk}$ measured along $\boldsymbol{w}_j$.
- α_{jk} is the angle made by axes $\boldsymbol{w}_j$ and $\boldsymbol{z}_k$, measured positively from $\boldsymbol{w}_j$ to $\boldsymbol{z}_k$ about $\boldsymbol{t}_{jk}$.
- β_{jk} is the angle made by axes $\boldsymbol{t}_{jk}$ and $\boldsymbol{x}_k$, measured positively from $\boldsymbol{t}_{jk}$ to $\boldsymbol{x}_k$ about the $\boldsymbol{z}_k$ axis.
- γ_{jk} is the angle made by axes $\boldsymbol{u}_j$ and $\boldsymbol{t}_{jk}$, measured positively from $\boldsymbol{u}_j$ to $\boldsymbol{t}_{jk}$ about $\boldsymbol{w}_j$.

They generate a geometric transformation which can be noted

$$\boldsymbol{G}_{jk} = \boldsymbol{G}(a_{jk}, b_{jk}, c_{jk}, \alpha_{jk}, \beta_{jk}, \gamma_{jk}) \tag{5.5.3}$$

It is such that

$$\begin{bmatrix} u_j \\ v_j \\ w_j \\ 1 \end{bmatrix} = \boldsymbol{G}_{jk} \begin{bmatrix} x_k \\ y_k \\ z_k \\ 1 \end{bmatrix} \tag{5.5.4}$$

The relative motion produced by the joint axis j is treated independently. It generates a displacement transformation which is supposed to depend on one single joint DOF q_j

$$\boldsymbol{D}_j = \boldsymbol{D}_j(q_j) \tag{5.5.5}$$

and such that

$$\begin{bmatrix} x_j \\ y_j \\ z_j \\ 1 \end{bmatrix} = \boldsymbol{D}_j \begin{bmatrix} u_j \\ v_j \\ w_j \\ 1 \end{bmatrix} \tag{5.5.6}$$

For a ternary link, several geometric transformations (5.5.4) have to be generated but the displacement part (5.5.5) of the transformation remains unique.

The resulting frame transformation from frame $\boldsymbol{x}_k\boldsymbol{y}_k\boldsymbol{z}_k$ to frame $\boldsymbol{x}_j\boldsymbol{y}_j\boldsymbol{z}_j$ is then calculated by

$$\boldsymbol{A}_{jk} = \boldsymbol{D}_j\boldsymbol{G}_{jk} \tag{5.5.7}$$

The advantages of this notation where geometric and relative motion transformations are treated separately are the following:

1. For ternary links connecting joint j to both joints k and l, two geometric transformations $\boldsymbol{G}_{jk}$ and $\boldsymbol{G}_{jl}$ have to be introduced but the displacement part remains common. It generates two transformations

$$\boldsymbol{A}_{jk} = \boldsymbol{D}_j\boldsymbol{G}_{jk} \qquad \text{and} \qquad \boldsymbol{A}_{jl} = \boldsymbol{D}_j\boldsymbol{G}_{jl} \tag{5.5.8}$$

2. Both coordinate systems $(u,\ v,\ w)$ and $(x,\ y,\ z)$ attached to extremities of the link are defined quite arbitrarily. An adequate choice often simplifies a lot the geometric description of the link, and may also make the description of the kinematic model easier.

3. The geometric transformation is specific of the link under consideration, while with the DH method it involves quantities attached to the previous link.

Sheth's notation allows to describe very simply a large number of joints such as not only revolute and prismatic pairs, but also cylindrical, screw, spherical, gear pairs ...

5.6 Sheth's geometric transformation

According to the notations adopted, and omitting the subscripts j and k for sake of clarity, the geometric transformation from origin to end of the link is

$$\begin{aligned}\begin{bmatrix} u \\ v \\ w \\ 1 \end{bmatrix} =& \boldsymbol{T}rans(0,0,c)\cdot \boldsymbol{R}ot(\boldsymbol{z},\gamma)\cdot \\ & \boldsymbol{T}rans(a,0,0)\cdot \boldsymbol{R}ot(\boldsymbol{x},\alpha)\cdot \boldsymbol{T}rans(0,0,b)\cdot \boldsymbol{R}ot(\boldsymbol{z},\beta)\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}\end{aligned} \tag{5.6.1}$$

One obtains the explicit expression of the transformation matrix

$$\boldsymbol{G} = \begin{bmatrix} \cos\beta\cos\gamma & -\sin\beta\cos\gamma & \sin\alpha\sin\gamma & a\cos\gamma \\ -\cos\alpha\sin\beta\sin\gamma & -\cos\alpha\cos\beta\sin\gamma & & +b\sin\alpha\sin\gamma \\ \cos\beta\sin\gamma & -\sin\beta\sin\gamma & -\sin\alpha\cos\gamma & a\sin\gamma \\ +\cos\alpha\sin\beta\cos\gamma & +\cos\alpha\cos\beta\cos\gamma & & -b\sin\alpha\cos\gamma \\ \sin\alpha\sin\beta & \sin\alpha\cos\beta & \cos\alpha & c+b\cos\alpha \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.6.2}$$

One can show that in the general case, equation (5.6.2) is easily evaluated from the knowledge of the coordinates of six points attached to the link.

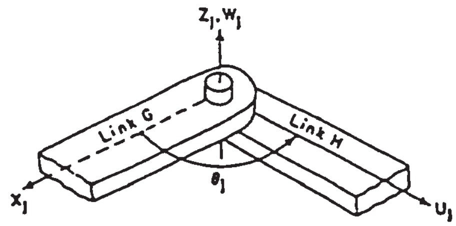


Figure 5.7.1: The revolute pair

5.7 Relative motion transformations

The explicit expression of matrix (5.6.2) depends upon the type of kinematic pair considered.

5.7.1 The revolute pair

The frame definition is such that

- axes $\boldsymbol{z}_j$ and $\boldsymbol{w}_j$ coincide with the rotation axis,

-both frames have same origin.

The displacement matrix is then

$$\boldsymbol{D}_j(\theta_j) = \begin{bmatrix} \cos\theta_j & -\sin\theta_j & 0 & 0 \\ \sin\theta_j & \cos\theta_j & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.7.1}$$

5.7.2 The prismatic pair

The only variable is the displacement d_j along the joint axis, and the displacement transformation is expressed according to the following conventions:

- axes $\boldsymbol{w}_j$ and $\boldsymbol{z}_j$ coincide with the translation direction;

- axes $\boldsymbol{u}_j$ and $\boldsymbol{x}_j$ are chosen parallel and noted positively in the same direction.

The transformation matrix is then

$$\boldsymbol{D}_j(d_j) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.7.2}$$

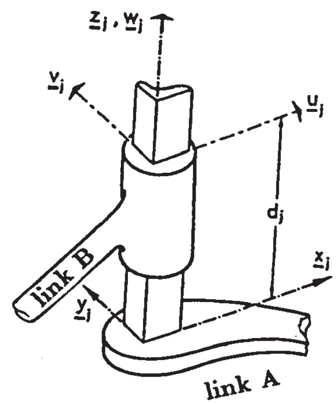


Figure 5.7.2: The prismatic pair

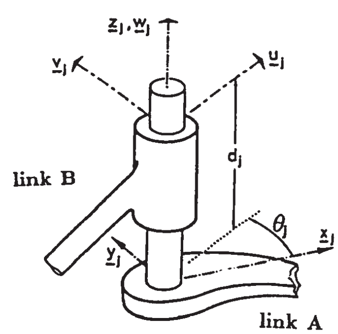


Figure 5.7.3: The cylindrical pair

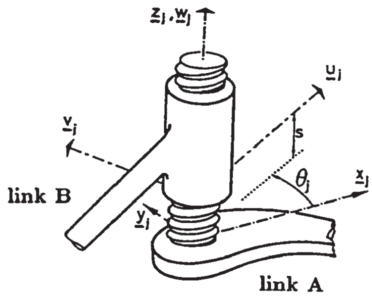


Figure 5.7.4: The screw pair

5.7.3 The cylindrical pair

In this case, the joint numbers 2 DOF: rotation and translation about the same axis. It can be described by a transformation combining one revolute and one prismatic joint. The transformation matrix is

$$\boldsymbol{D}_j(\theta_j, d_j) = \begin{bmatrix} \cos\theta_j & -\sin\theta_j & 0 & 0 \\ \sin\theta_j & \cos\theta_j & 0 & 0 \\ 0 & 0 & 1 & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.7.3}$$

The only condition for a correct use of (5.7.3) is that axes $\boldsymbol{z}_j$ and $\boldsymbol{w}_j$ have to be aligned along the joint axis.

5.7.4 The screw joint

When the displacements θ_j and d_j of a cylindrical joint become proportional, the cylindrical pair degenerates into a screw pair. The relationship between θ_j and d_j reduces to one the number of DOF of the pair; it is expressed in terms of the lead of the screw, defined by

$$L_j = \frac{2\pi d_j}{\theta_j} \tag{5.7.4}$$

The transformation may be indifferently expressed in terms of the relative rotation θ_j or displacement d_j

$$\boldsymbol{D}_j(\theta_j) = \begin{bmatrix} \cos\theta_j & -\sin\theta_j & 0 & 0 \\ \sin\theta_j & \cos\theta_j & 0 & 0 \\ 0 & 0 & 1 & \frac{L_j\theta_j}{2\pi} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.7.5}$$

or

$$\boldsymbol{D}_j(d_j) = \begin{bmatrix} \cos\left(\frac{2\pi d_j}{L_j}\right) & -\sin\left(\frac{2\pi d_j}{L_j}\right) & 0 & 0 \\ \sin\left(\frac{2\pi d_j}{L_j}\right) & \cos\left(\frac{2\pi d_j}{L_j}\right) & 0 & 0 \\ 0 & 0 & 1 & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.7.6}$$

Both transformations can be used under the following restrictive conditions:

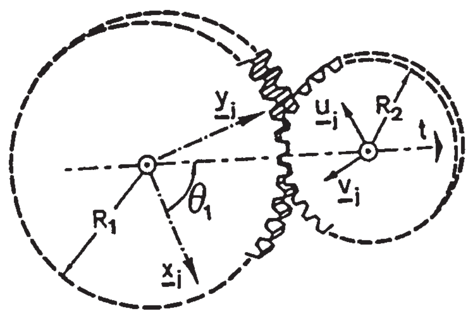


Figure 5.7.5: The gear pair

- Axes $\boldsymbol{w}_j$ and $\boldsymbol{z}_j$ have to be aligned along the screw axis;

- Frames $\boldsymbol{u}_j\boldsymbol{v}_j\boldsymbol{w}_j$ and $\boldsymbol{x}_j\boldsymbol{y}_j\boldsymbol{z}_j$ have to be chosen in such a way that the axes $\boldsymbol{u}_j$ and $\boldsymbol{x}_j$ coincide in a given reference configuration.

5.7.5 The gear pair

The gear pair is chosen as an example of how the extended notation may be used for higher pairs. The frames $\boldsymbol{u}_j\boldsymbol{v}_j\boldsymbol{w}_j$ and $\boldsymbol{x}_j\boldsymbol{y}_j\boldsymbol{z}_j$ are attached respectively to links G and H in their rotation axis.

The common perpendicular to axes $\boldsymbol{z}_j$ and $\boldsymbol{w}_j$ is the $\boldsymbol{t}$ axis; then, the general transformation $\boldsymbol{G}$ of equation (5.6.2) may be used to describe the displacement rotation introduced in the gear pair.

With the choice of parameters

$$a = R_1 + R_2, \qquad \alpha = b = c = 0, \qquad \gamma = \theta_1, \qquad \beta = \theta_2 \tag{5.7.7}$$

and the definition of the gear ratio

$$\tau = \frac{R_1}{R_2} \tag{5.7.8}$$

substitution into (5.6.2) provides the explicit expression

$$\boldsymbol{D}_j(\theta_1) = \begin{bmatrix} \cos[(1+\tau)\theta_1] & -\sin[(1+\tau)\theta_1] & 0 & (1+\tau)R_2\cos\theta_1 \\ \sin[(1+\tau)\theta_1] & \cos[(1+\tau)\theta_1] & 0 & (1+\tau)R_2\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.7.9}$$

5.8 Sheth's description of the PUMA 560

The application of Sheth's formulation to describe the kinematics of an open-tree simply connected structure is relatively straightforward. As an example, consider the PUMA 560 structure of section 5.4.

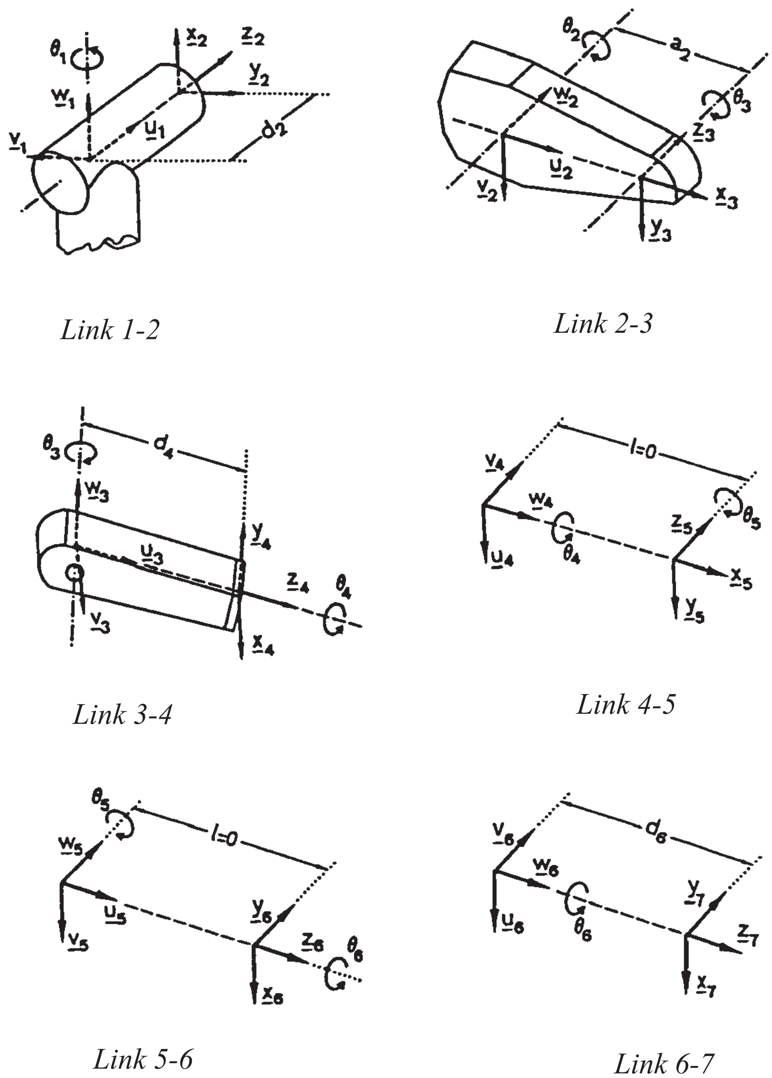


Figure 5.8.1: Successive frames in Sheth's representation of the PUMA 560

link	a	b	c	α	β	γ
1–2	0	d_2	0	$+\frac{\pi}{2}$	$+\frac{\pi}{2}$	$+\frac{\pi}{2}$
2–3	a_2	0	0	0	0	0
3–4	0	d_4	0	$+\frac{\pi}{2}$	0	$+\frac{\pi}{2}$
4–5	0	0	0	$+\frac{\pi}{2}$	$+\frac{\pi}{2}$	π
5–6	0	0	0	$+\frac{\pi}{2}$	0	$+\frac{\pi}{2}$
6–7	0	d_6	0	0	0	0

Table 5.2: Geometric parameters for PUMA robot 560

The first step consists of breaking the structure into individual components and attaching to each of them origin and end frames $\boldsymbol{u}_i\boldsymbol{v}_i\boldsymbol{w}_i$ and $\boldsymbol{x}_k\boldsymbol{y}_k\boldsymbol{z}_k$. The axes $\boldsymbol{w}_j$ and $\boldsymbol{z}_k$ are each time taken into coincidence with joint axes.

It is easy to verify that the geometric parameters describing each of the members are those given in the table of Table 5.2

The resulting transformations are

$$\boldsymbol{G}_{12} = \begin{bmatrix} 0 & 0 & 1 & d_2 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{G}_{23} = \begin{bmatrix} 1 & 0 & 0 & a_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\boldsymbol{G}_{34} = \begin{bmatrix} 0 & 0 & 1 & d_4 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{G}_{45} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.8.1}$$
$$\boldsymbol{G}_{56} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{G}_{67} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

5.9 Inversion of geometric kinematic model

One has already mentioned that the problem of solving the kinematic equations of a manipulator is a highly nonlinear problem. These equations express, as depicted by figure 5.9.1, a correspondence between *joint (or configuration) space* and *task (or operational) space.*

The number of variables in joint space is equal to the number m of DOF occurring at the joints of the system. In particular, for a simply-connected open-tree structure will all joints of class 5, this number equals the number of links.

The number of variables in task space is equal to the number of DOF of the task. Arbitrary location and orientation of the tool implies the definition of 6 parameters is task space. Specific tasks such as vertical insertion, manipulation of cylindrical objects, use of rotating tools (grinders, screw drivers ...) can be defined with a smaller number of DOF.

Given the joint variables $(q_1, \ldots q_m)$, the equation

$$\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{q}) \tag{5.9.1}$$

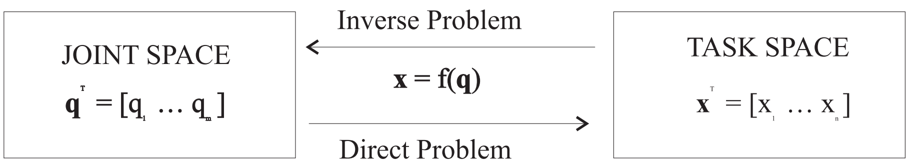


Figure 5.9.1: Nonlinear mapping by the geometric kinematic model

has always one and only one solution since the functions $f^T = [f_1 \dots f_m]$ may always be expressed in an explicit manner. The direct problem is thus trivial.

The problem of solvability arises when considering the inverse problem: given the position and orientation of the tool in task space, which set of joint displacements actually generates such a configuration?

Since the functions $(f_1, \dots f_m)$ are of transcendental type and are thus highly nonlinear, we must concern ourselves with the existence of solutions, their multiplicity and the methods available to reach them.

Existence of solutions

The question of whether solutions exist or not raises first the question of *manipulator workspace*.

When examining the kinematic behavior of planar structures, we have already defined workspace as the space which the end-effector or the manipulator can reach. The definition may be improved further by distinguishing between

- *dextrous workspace*, as the volume of space that the end-effector can reach with all orientations.
- *position workspace*, as the volume of space that it can reach in at least one orientation.

Workspace is limited by the geometry of the system on one hand, and by the mobility allowed to the joints on the other hand.

The existence of solutions depends also on the number of degrees of freedom of the system.

- The normal case is when the number of joints equals 6. Then, provided that no DOF are redundant and that the goal assigned to the manipulator lies within the workspace, solutions normally exist in finite number. The different solutions correspond then to different possible configurations to reach the same point. The multiplicity of the solution depends upon the number of joints of the manipulator and their type. The fact that a manipulator has multiple solutions may cause problems since the system has to be able to select one of them. The criteria upon which to base a decision may vary, but a very reasonable choice consists to choose the 'closest solution to the current configuration.
- When the number of joints m is less than 6, no solution exists unless freedom is reduced in the same time in task space, for example by constraining the tool orientation to certain directions. Mathematically, it is always possible, given an arbitrary position and orientation of the tool, to find a robot configuration which brings the tool in the 'closest configuration to it. This resulting configuration will then depend on the criterion adopted to measure proximity.

- When the number of joints m exceeds 6, the structure becomes redundant and an infinite number of solutions exists then to reach the same point within the manipulator workspace. Redundancy of the robot architecture is an interesting feature for systems installed in a highly constrained environment. From the kinematic point of view, the difficulty lies in formulating the environment constraints in mathematical form in order to insure the uniqueness of the solution to the inverse kinematic problem.

5.10 Closed form inversion of PUMA 560 robot

5.10.1 Decoupling between position and orientation

The very first step in order to obtain a closed form solution to the geometric kinematic model of the PUMA robot displayed in figure 5.4.1 is to define the position of the end-tool frame in such a way that uncoupling occurs between position and orientation.

Let us express in global coordinates the location of the wrist center

$$\boldsymbol{p} = \boldsymbol{T}\boldsymbol{a} \qquad \text{with} \qquad \boldsymbol{a}^T = [0\ 0\ -d_6] \tag{5.10.1}$$

where $d_6$is the length of the tool. The resulting point is located at the intersection of the wrist axes $\boldsymbol{z}_4$, $\boldsymbol{z}_5$ and $\boldsymbol{z}_6$, and substitution of (5.10.1) into (5.4.8) provides the new translation vector

$$\boldsymbol{p} = \begin{bmatrix} C_1(S_{23}d_4 + a_2C_2) - S_1d_2 \\ S_1(S_{23}d_4 + a_2C_2) + S_1d_2 \\ C_{23}d_4 - a_2S_2 \end{bmatrix} \tag{5.10.2}$$

By gathering the rotation part of (5.4.9) and the translation vector (5.10.2), a new transformation can be defined

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix} \tag{5.10.3}$$

which verifies now the uncoupling property.

5.10.2 Pieper's technique

In order to perform the closed-form inversion of (5.10.3) in a systematic manner, let us introduce a systematic procedure proposed by Pieper.

It consists to start from the global transformation (5.4.2) and extract recursively from it the following equalities

$$\begin{aligned} \boldsymbol{T} &=^0 \boldsymbol{A}_1 \cdot ^1\boldsymbol{A}_2 \cdot ^2\boldsymbol{A}_3 \cdot ^3\boldsymbol{A}_4 \cdot ^4\boldsymbol{A}_5 \cdot ^5\boldsymbol{A}_6 =^0 \boldsymbol{A}_6 \\ ^0\boldsymbol{A}_1^{-1} \cdot \boldsymbol{T} &= \quad ^1\boldsymbol{A}_2 \cdot ^2\boldsymbol{A}_3 \cdot ^3\boldsymbol{A}_4 \cdot ^4\boldsymbol{A}_5 \cdot ^5\boldsymbol{A}_6 =^1 \boldsymbol{A}_6 \\ ^1\boldsymbol{A}_2^{-1} \cdot ^0\boldsymbol{A}_1^{-1} \cdot \boldsymbol{T} &= \quad ^2\boldsymbol{A}_3 \cdot ^3\boldsymbol{A}_4 \cdot ^4\boldsymbol{A}_5 \cdot ^5\boldsymbol{A}_6 =^2 \boldsymbol{A}_6 \\ &\vdots \\ ^4\boldsymbol{A}_5^{-1} \dots ^1\boldsymbol{A}_2^{-1} \cdot ^0\boldsymbol{A}_1^{-1} \cdot \boldsymbol{T} &= \quad =^5 \boldsymbol{A}_6 \end{aligned} \tag{5.10.4}$$

It is then obvious that for each equation of (5.10.4),

- The left-hand side is a function of the task coordinates $\boldsymbol{x}$ (contained in the $\boldsymbol{n}$, $\boldsymbol{o}$, $\boldsymbol{a}$ and $\boldsymbol{p}$ vectors) and of the joint coordinates $(q_1, q_2, \dots q_i, \quad i = 1, \dots 5)$.

- The right-hand side is a function of the remaining coordinates $(q_{i+1}, \dots q_6)$, unless some zeros or constants appear in the explicit expressions of matrices ${}^{i+1}\boldsymbol{A}_6$.

The trick of the method is to identify the elements of ${}^{i+1}\boldsymbol{A}_6$ which allow to extract an explicit expression for q_i.

5.10.3 General procedure to determine joint angles

Before applying the technique to the kinematic model of the PUMA, let us note that

ATAN2 function

In order to determine the joint angles in the correct quadrant, they should always be evaluated from their sine and cosine values u and v by

$$\theta = ATAN2\ (u, v) \tag{5.10.5}$$

where $ATAN2\ (u, v)$ is the $\tan^{-1}$ function modified in the following way

$$\theta = ATAN2\ (u, v) = \begin{cases} \tan^{-1}\frac{u}{v} & \text{if } v > 0 \\ \tan^{-1}\frac{u}{v} + \pi sign(u) & \text{if } v < 0 \\ +\frac{\pi}{2} sign(u) & \text{if } v = 0 \end{cases} \tag{5.10.6}$$

General solution of trigonometric equation $a\cos\theta + b\sin\theta = c$

Equations of type

$$a\cos\theta + b\sin\theta = c \tag{5.10.7}$$

can be solved by assuming

$$a = \rho\sin\phi \qquad b = \rho\cos\phi \tag{5.10.8}$$

giving

$$\rho = \sqrt{a^2 + b^2} \qquad \text{and} \qquad \phi = ATAN2\ (a, b) \tag{5.10.9}$$

in which case

$$\sin(\theta + \phi) = \frac{c}{\rho} \qquad \cos(\theta + \phi) = \pm\sqrt{1 - \frac{c^2}{\rho^2}} \tag{5.10.10}$$

The solution to (5.10.7) is thus

$$\theta = ATAN2\ (\frac{c}{\rho}, \pm\sqrt{1 - \frac{c^2}{\rho^2}}) - ATAN2\ (a, b)$$

$$\theta = ATAN2\ (c, \pm\sqrt{\rho^2 - c^2}) - ATAN2\ (a, b) \tag{5.10.11}$$

It has

- 2 solutions if $\rho^2 = a^2 + b^2 > c^2$
- 1 solution if $\rho^2 = c^2$
- 0 solution if $\rho^2 < c^2$.

5.10.4 Calculation of θ_1

According to Pieper's technique and making use of (5.4.1) and (5.4.8), letus express the equation

$$ {}^0\boldsymbol{A}_1^{-1}\,\boldsymbol{T} = \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -S_1 & C_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.10.12} $$

in the form

$$ {}^0\boldsymbol{A}_1^{-1}\cdot\boldsymbol{T} = \begin{bmatrix} f_{11}(\boldsymbol{n}) & f_{11}(\boldsymbol{o}) & f_{11}(\boldsymbol{a}) & f_{11}(\boldsymbol{p}) \\ f_{12}(\boldsymbol{n}) & f_{12}(\boldsymbol{o}) & f_{12}(\boldsymbol{a}) & f_{12}(\boldsymbol{p}) \\ f_{13}(\boldsymbol{n}) & f_{13}(\boldsymbol{o}) & f_{13}(\boldsymbol{a}) & f_{13}(\boldsymbol{p}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.10.13} $$

with the functions

$$ \begin{aligned} f_{11}(\lambda) &= C_1\lambda_x + S_1\lambda_y \\ f_{12}(\lambda) &= -\lambda_z \\ f_{13}(\lambda) &= -S_1\lambda_x + C_1\lambda_y \end{aligned} \tag{5.10.14} $$

where λ is n, o, a or p. Let us identify it with ${}^1\boldsymbol{A}_6$ (where d_6 is set equal to 0 for the wrist center location)

$$ {}^1\boldsymbol{A}_6 = \begin{bmatrix} \star & \star & \star & S_{23}d_4 + a_2C_2 \\ \star & \star & \star & -C_{23}d_4 + a_2S_2 \\ \star & \star & \star & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.10.15} $$

One obtains directly

$$ f_{13}(\boldsymbol{p}) = -S_1p_x + C_1p_y = d_2 \tag{5.10.16} $$

Equation (5.10.16) has the general form (5.10.7), and its solution is thus

$$ \theta_1 = ATAN2\ (p_y, p_x) + ATAN2\ (d_2, \pm\sqrt{p_x^2 + p_y^2 - d_2^2}) \tag{5.10.17} $$

Equation (5.10.17) provides two possible values of θ_1 which correspond to right- and left-hand configurations (figure 5.10.1). Note that the multiplicity of the solution to θ_1 is generated by the arm offset d_2.

5.10.5 Calculation of θ_2 and θ_3

Rather than following strictly Pieper's technique, it is easier to still use the information contained in equations (5.10.12)- (5.10.13). The following relationships may be extracted

$$ \begin{aligned} f_{11}(\boldsymbol{p}) &= C_1p_x + S_1p_y = S_{23}d_4 + a_2C_2 = \alpha \\ f_{12}(\boldsymbol{p}) &= -p_z = -C_{23}d_4 + a_2S_2 = \beta \end{aligned} \tag{5.10.18} $$

Summing the squares gives

$$ \alpha^2 + \beta^2 = a_2^2 + d_4^2 + 2a_2d_4S_3 \tag{5.10.19} $$

and a second equation for S_3 can be generated in the form

$$ \alpha C_2 + \beta S_2 = a_2 + S_3d_4 \tag{5.10.20} $$

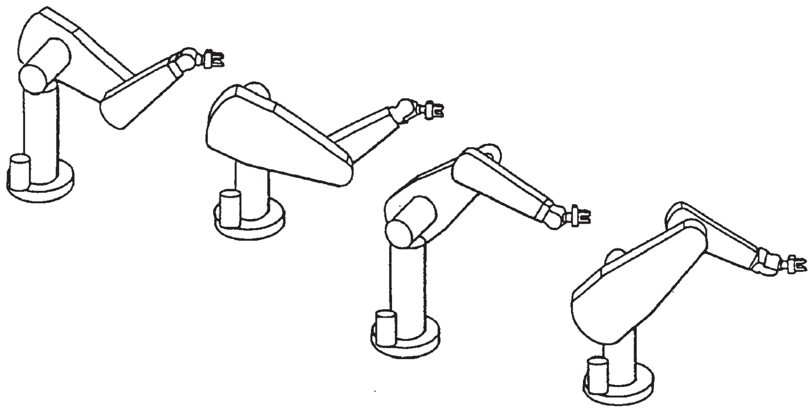


Figure 5.10.1: four possible solutions for the PUMA 560 arm

Elimination of S_3 between (5.10.19) and (5.10.20) provides an expression for θ_2

$$\alpha C_2 + \beta S_2 = \frac{\alpha^2 + \beta^2 + a_2^2 - d_4^2}{2a_2} = \gamma \tag{5.10.21}$$

which according to (5.10.11) has the solution

$$\theta_2 = -ATAN2\ (\alpha, \beta) + ATAN2\ (\gamma, \pm\sqrt{\alpha^2 + \beta^2 - \gamma^2}) \tag{5.10.22}$$

Again, two possible solutions appear which correspond to upper and lower arm configurations (see figure 5.10.1).

Finally, θ_3 can be obtained by making use of (5.10.20) and of another equation for θ_3 generated by

$$\alpha S_2 - \beta C_2 = C_3 d_4 \tag{5.10.23}$$

giving

$$\theta_3 = ATAN2\ (\alpha C_2 + \beta S_2 - a_2, \alpha S_2 - \beta C_2) \tag{5.10.24}$$

Let us note that equations (5.10.17), (5.10.22) and (5.10.24) involve only position parameters: they correspond to the solution of the uncoupled position problem

$$\boldsymbol{p} = \boldsymbol{f}\ (\theta_1, \theta_2, \theta_3) \tag{5.10.25}$$

5.10.6 Calculation of θ_4, θ_5 and θ_6

To obtain the values of wrist angles, we rely again on Pieper's technique by making use of the relationship

$$^2\boldsymbol{A}_3^{-1} \cdot {}^1\boldsymbol{A}_2^{-1} \cdot {}^0\boldsymbol{A}_1^{-1} \cdot \boldsymbol{T} = {}^3\boldsymbol{A}_6 \tag{5.10.26}$$

which can be put in the form

$$\begin{bmatrix} f_{31}(\boldsymbol{n}) & f_{31}(\boldsymbol{o}) & f_{31}(\boldsymbol{a}) & f_{31}(\boldsymbol{p}) - a_2 \\ f_{32}(\boldsymbol{n}) & f_{32}(\boldsymbol{o}) & f_{32}(\boldsymbol{a}) & f_{32}(\boldsymbol{p}) \\ f_{33}(\boldsymbol{n}) & f_{33}(\boldsymbol{o}) & f_{33}(\boldsymbol{a}) & f_{33}(\boldsymbol{p}) - d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \star & \star & C_4 S_5 & 0 \\ \star & \star & S_4 S_5 & 0 \\ -S_5 C_6 & S_5 S_6 & C_5 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.10.27}$$

with the functions

$$\begin{aligned} f_{31}(\lambda) &= C_1 C_{23}\lambda_x + S_1 C_{23}\lambda_y - S_{23}\lambda_z \\ f_{32}(\lambda) &= -S_1\lambda_x + C_1\lambda_y \\ f_{33}(\lambda) &= C_1 S_{23}\lambda_x + S_1 S_{23}\lambda_y + C_{23}\lambda_z \end{aligned} \tag{5.10.28}$$

with λ representing n, o, a or p One finds directly the angle values

$$\begin{aligned} \theta_4 &= ATAN2\ [f_{32}(\boldsymbol{a}), f_{31}(\boldsymbol{a})] \\ \theta_5 &= ATAN2\ [C_4 f_{31}(\boldsymbol{a}) + S_4 f_{32}(\boldsymbol{a}), f_{33}(\boldsymbol{a})] \\ \theta_6 &= ATAN2\ [f_{33}(\boldsymbol{o}), -f_{33}(\boldsymbol{n})] \end{aligned} \tag{5.10.29}$$

5.11 Numerical solution to inverse problem

Let us suppose first for sake of simplicity that the geometric kinematic model can be put in the form

$$x_i = f_i(q_j) \qquad (i = 1, \dots n,\ j = 1, \dots m) \tag{5.11.1}$$

and that, for given x_i, an approximate solution q_i^* is available.

The principle of all iterative techniques to solve a nonlinear problem of general form (5.11.1) consists in calculating a correction δq_i to the solution such that

$$x_i = f_i(q_j^* + \delta q_j) \tag{5.11.2}$$

A first order Taylor expansion of equation (5.11.2) provides the expression

$$x_i = f_i(q_j^*) + \sum_{i=1}^{m} \frac{\partial f_i}{\partial q_j}\delta q_j + O(\delta q_j^2) \tag{5.11.3}$$

Let us define next

$$x_i^* = f_i(q_j^*) \tag{5.11.4}$$

and the residual quantity

$$r_i = x_i - x_i^* \tag{5.11.5}$$

and let us observe that the right-hand side in (5.11.3) involves the jacobian matrix of the system

$$\boldsymbol{J} \ = \ \left[\frac{\partial f_i}{\partial q_j}\right] \quad \text{dimension} \quad n \times m \tag{5.11.6}$$

Equation (5.11.3) can then be written in the matrix form

$$\boldsymbol{r} = \boldsymbol{J}\delta\boldsymbol{q} + O(\delta\boldsymbol{q}^2) \tag{5.11.7}$$

which forms the basis for iterative solutions.

The method of solution depends on whether $m = n$ or not. Three cases have thus to be considered:

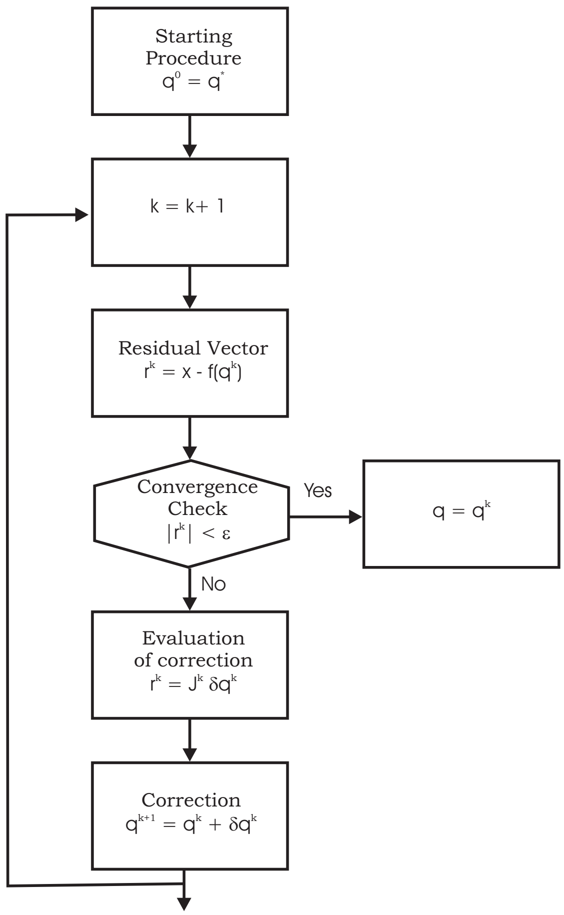


Figure 5.11.1: Newton-Raphson solution to inverse problem

a) $m = n$

Provided that the jacobian matrix remains non singular, the linearized equation

$$\boldsymbol{r} = \boldsymbol{J}\delta\boldsymbol{q} \tag{5.11.8}$$

possesses then a unique solution, and the Newton-Raphson technique may then be used to solve equation (5.11.1). The principle is then as described by figure 5.11.1.

Note that:

- The cost of the procedure depends on the number of iterations to be performed, which depends itself upon parameters such as the distance between estimated and effective solutions and the condition number of the jacobian matrix at the solution.
- Since the solution to the inverse problem is not unique, it may generate different configurations according to the choice of the estimated solution.
- No convergence may be observed if the initial estimate of the solution falls outside the convergence domain of the algorithm.

Much effort is thus needed to develop more robust numerical solutions to (5.11.1).

b) $m < n$

This is the overdetermined case for which no solution exists in general, since the number of joints is not sufficient to generate an arbitrary configuration of the tool.

A solution can however be generated which minimizes the position error. Consider the problem

$$\min_{q_1,\dots q_m} \left\{ F = \frac{1}{2} \sum_{i=1}^{n} w_i [x_i - f_i(q_j)]^2 \right\} \tag{5.11.9}$$

or, in matrix form

$$\min_{\boldsymbol{q}} \left\{ F = \frac{1}{2} [\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{q})]^T \boldsymbol{W} [\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{q})] \right\} \tag{5.11.10}$$

where

$$\boldsymbol{W} = diag(w_1 \dots w_n) \tag{5.11.11}$$

is a set of weighting factors giving a relative importance to each of the kinematic equations.

The position error is minimum when

$$\frac{\partial F}{\partial q_j} = - \sum_i \frac{\partial f_j}{\partial q_j} w_i [x_i - f_i(q_j)] = 0 \tag{5.11.12}$$

or, in matrix form

$$\boldsymbol{J}^T \boldsymbol{W} [\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{q})] = 0 \tag{5.11.13}$$

A Taylor expansion of the third factor shows that the linear correction to an estimated solution $\boldsymbol{q}^*$ is

$$\boldsymbol{J}^T \boldsymbol{W} [\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{q}^*) - \boldsymbol{J}\delta\boldsymbol{q}] = 0 \tag{5.11.14}$$

The correction equation is thus

$$\boldsymbol{J}^T \boldsymbol{W} \boldsymbol{J} \delta\boldsymbol{q} = \boldsymbol{J}^T \boldsymbol{W} \boldsymbol{r} \tag{5.11.15}$$

where $\boldsymbol{r}$ is the residual vector defined by (5.11.5).

Since $\boldsymbol{W}$ is a positive definite diagonal matrix, the matrix $\boldsymbol{J}^T\boldsymbol{W}\boldsymbol{J}$ is always symmetric and may thus be inverted. It provides the *generalized inverse* to the jacobian matrix

$$\boldsymbol{J}^+ = (\boldsymbol{J}^T\boldsymbol{W}\boldsymbol{J})^{-1}\boldsymbol{J}^T\boldsymbol{W} \tag{5.11.16}$$

which verifies the property

$$\boldsymbol{J}^+\boldsymbol{J} = \boldsymbol{I} \tag{5.11.17}$$

It is easy to verify that whenever $\boldsymbol{J}$ is invertible,

$$\boldsymbol{J}^+ = \boldsymbol{J}^{-1} \tag{5.11.18}$$

and the solution to (5.11.10) does not differ from the solution to the nonlinear system (5.11.1).

c) $m > n$

This is the redundant case for which an infinity of solutions is generally available. Selection of an appropriate solution can be made under the condition that it is optimal in some sense.

For example, let us seek for a solution to (5.11.1) which minimizes the deviation from a given reference configuration $\boldsymbol{q}_0$. The problem may then be formulated as that of finding the minimum of a constrained function

$$\min_{\boldsymbol{q}} \left\{ F = \frac{1}{2}[\boldsymbol{q}-\boldsymbol{q}_0]^T\boldsymbol{W}[\boldsymbol{q}-\boldsymbol{q}_0] \right\} \tag{5.11.19}$$

subject to

$$\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{q}) = 0 \tag{5.11.20}$$

Using the technique of Lagrangian multipliers, problem (5.11.19)-(5.11.20) may be replaced by the equivalent one

$$\frac{\partial G}{\partial \boldsymbol{q}} = 0 \qquad \frac{\partial G}{\partial \lambda} = 0 \tag{5.11.21}$$

with the definition of the augmented functional

$$G(\boldsymbol{q},\lambda) = \frac{1}{2}[\boldsymbol{q}-\boldsymbol{q}_0]^T\boldsymbol{W}[\boldsymbol{q}-\boldsymbol{q}_0] + \lambda^T\left[\boldsymbol{x}-\boldsymbol{f}(\boldsymbol{q})\right] \tag{5.11.22}$$

It leads to a system of $m+n$ equations with $m+n$ unknowns

$$\begin{aligned} &\boldsymbol{W}[\boldsymbol{q}-\boldsymbol{q}_0] - \boldsymbol{J}^T\lambda = 0 \\ &\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{q}) = 0 \end{aligned} \tag{5.11.23}$$

Linearization of equations (5.11.23) provides the system of equations for the displacement corrections and variations of lagrangian multipliers

$$\begin{aligned} &\boldsymbol{W}\delta\boldsymbol{q} - \boldsymbol{J}^T\delta\lambda = 0 \\ &\boldsymbol{J}\delta\boldsymbol{q} = \boldsymbol{r} \end{aligned} \tag{5.11.24}$$

Substitution of the solution $\delta\boldsymbol{q}$ obtained from the first equation (5.11.24) into the second one yields to

$$\boldsymbol{J}\boldsymbol{W}^{-1}\boldsymbol{J}^T\delta\lambda = \boldsymbol{r} \tag{5.11.25}$$

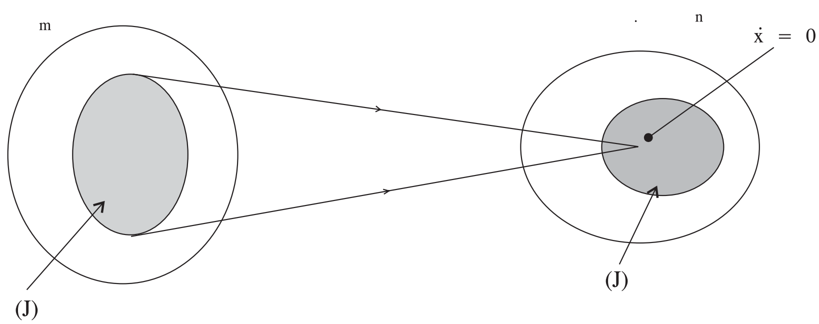


Figure 5.12.1: Linear mapping diagram of jacobian matrix

or, in terms of the displacement correction

$$\delta \boldsymbol{q} = \boldsymbol{W}^{-1}\boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{W}^{-1}\boldsymbol{J}^T)^{-1}\boldsymbol{r} \tag{5.11.26}$$

The matrix

$$\boldsymbol{J}^{+} = \boldsymbol{W}^{-1}\boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{W}^{-1}\boldsymbol{J}^T)^{-1} \tag{5.11.27}$$

has the meaning of a *pseudo-inverse* to the singular jacobian matrix $\boldsymbol{J}$. It verifies the identity

$$\boldsymbol{J}\boldsymbol{J}^{+} = \boldsymbol{I} \tag{5.11.28}$$

and again, whenever $\boldsymbol{J}$ is invertible,

$$\boldsymbol{J}^{+} = \boldsymbol{J}^{-1} \tag{5.11.29}$$

5.12 Linear mapping diagram of the jacobian matrix

At the light of the results of the previous section, it is interesting to interpret the properties of the jacobian matrix in a linear algebra context.

The jacobian matrix $\boldsymbol{J}$ relates the infinitesimal displacements $\delta\boldsymbol{x} = [\delta x_1 \dots \delta x_n]$ of the end-effector to the infinitesimal joint displacements $\delta\boldsymbol{q} = [\delta q_1 \dots \delta q_m]$ and has thus dimension $n \times m$. When m is larger than n and $\boldsymbol{J}$ has full rank, there are $m - n$ redundancies in the system to which correspond $m - n$ arbitrary variables.

The jacobian matrix $\boldsymbol{J}$ determines also the relationship between end-effector velocities $\dot{\boldsymbol{x}}$ and joint velocities

$$\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}} \tag{5.12.1}$$

Equation (5.12.1) can be regarded as a linear mapping from a $n-$ dimensional vector space $\mathcal{X}^n$ to a $m-$dimensional vector space $\mathcal{Q}^m$ which can be interpreted as displayed in figure 5.12.1.

The subspace $\mathcal{R}(\boldsymbol{J})$ is the *range space* of the linear mapping, and represents all the possible end-effector velocities that can be generated by the n joints in the current configuration. If $\boldsymbol{J}$ has full row-rank, which means that the system does not present any singularity in that configuration, then the range space $\mathcal{R}(\boldsymbol{J})$ covers the entire vector space $\mathcal{X}^n$. Otherwise, there exists at least one direction in which the end-effector cannot be moved.

The null space $\mathcal{N}(\boldsymbol{J})$ of figure 5.12.1 represents the solutions of $\boldsymbol{J}\dot{\boldsymbol{q}} = 0$. Therefore, any vector $\dot{\boldsymbol{q}} \in \mathcal{N}(\boldsymbol{J})$ does not generate any motion of the end effector.

If the manipulator has full rank, the dimension of the null space is then equal to the number $n - m$ of redundant DOF. When $\boldsymbol{J}$ is degenerate, the dimension of $\mathcal{R}(\boldsymbol{J})$ decreases and at the same time the dimension of the null space increases by the same amount. Therefore, the relationship holds

$$dim\ \mathcal{R}(\boldsymbol{J}) + dim\ \mathcal{N}(\boldsymbol{J}) = m \tag{5.12.2}$$

Remark

Configurations in which the jacobian has no longer full rank corresponds to singularities of the mechanism, which are generally of two types:

- *Workspace boundary singularities* are those occurring when the manipulator is fully stretched out or folded back on itself, in which case the end effector is near or at the workspace boundary.

- *Workspace interior singularities* are those occurring away from the boundary, generally when two or more axes line up.

5.13 Effective computation of Jacobian matrix

According to the results of section (4.16), the infinitesimal transformation matrix

$$\delta\boldsymbol{T} = \begin{bmatrix} \delta\boldsymbol{R} & \delta\boldsymbol{r} \\ 0^T & 0 \end{bmatrix} \tag{5.13.1}$$

gives rise to the differential matrix

$$\delta\boldsymbol{T} \cdot \boldsymbol{T}^{-1} = \begin{bmatrix} \delta\tilde{\alpha} & \delta\boldsymbol{v} \\ 0^T & 0 \end{bmatrix} \tag{5.13.2}$$

where

$$\delta\tilde{\alpha} = \delta\boldsymbol{R} \cdot \boldsymbol{R}^T = \begin{bmatrix} 0 & -\delta\alpha_z & \delta\alpha_y \\ \delta\alpha_z & 0 & -\delta\alpha_x \\ -\delta\alpha_y & \delta\alpha_x & 0 \end{bmatrix} \tag{5.13.3}$$

is the matrix of infinitesimal rotations, and $\delta\boldsymbol{v}$ is a vector related to infinitesimal displacements such that

$$\delta\boldsymbol{r} = \delta\tilde{\alpha}\ \boldsymbol{r} + \delta\boldsymbol{v} \tag{5.13.4}$$

Collecting thus the information contained in (5.13.2), one obtains the variation of vector $\boldsymbol{x}$ describing the end-effector configuration in task space

$$\delta\boldsymbol{x} = \begin{bmatrix} \delta\alpha \\ \delta\boldsymbol{r} \end{bmatrix} \tag{5.13.5}$$

One knows on the other hand that $\boldsymbol{T}$ is a function of joint coordinates $(q_1,\ \ldots q_m)$ such that

$$\boldsymbol{T} = \boldsymbol{A}_1(q_1) \cdot \boldsymbol{A}_2(q_2) \ldots \boldsymbol{A}_m(q_m) \tag{5.13.6}$$

Its differentiation leads thus to

$$\delta\boldsymbol{T} = \sum_{i=1}^{m} \boldsymbol{A}_1(q_1) \cdot \boldsymbol{A}_2(q_2) \ldots \frac{\partial \boldsymbol{A}_i}{\partial q_i} \ldots \boldsymbol{A}_m(q_m)\delta q_i \tag{5.13.7}$$

Let us next express the partial derivatives in the form

$$\frac{\partial \boldsymbol{A}_i}{\partial q_i} = \boldsymbol{\Delta}_i \boldsymbol{A}_i \tag{5.13.8}$$

By referring to the DH transformation matrix (5.2.2), it is easy to verify that

- for a revolute joint:

$$\boldsymbol{\Delta}_i = \begin{bmatrix} \tilde{\boldsymbol{k}} & 0 \\ 0^T & 0 \end{bmatrix} \tag{5.13.9}$$

  since it has the meaning of a unit rotation about $\boldsymbol{z}$ axis, and

- for a prismatic joint:

$$\boldsymbol{\Delta}_i = \begin{bmatrix} 0 & \boldsymbol{k} \\ 0^T & 0 \end{bmatrix} \tag{5.13.10}$$

  since it corresponds to a unit displacement about the $\boldsymbol{z}$ axis.

Let us next express each term of the series (5.13.7) in the form

$$\boldsymbol{A}_1(q_1) \cdot \boldsymbol{A}_2(q_2) \dots \frac{\partial \boldsymbol{A}_i}{\partial q_i} \dots \boldsymbol{A}_m(q_m) = \boldsymbol{C}_i \cdot \boldsymbol{T} \tag{5.13.11}$$

where $\boldsymbol{C}_i$ has the expression

$$\begin{aligned} \boldsymbol{C}_i &= (\boldsymbol{A}_1 \cdot \boldsymbol{A}_2 \dots \boldsymbol{A}_{i-1}) \frac{\partial \boldsymbol{A}_i}{\partial q_i} (\boldsymbol{A}_1 \cdot \boldsymbol{A}_2 \dots \boldsymbol{A}_i)^{-1} \\ &= (\boldsymbol{A}_1 \cdot \boldsymbol{A}_2 \dots \boldsymbol{A}_{i-1}) \boldsymbol{\Delta}_i (\boldsymbol{A}_1 \cdot \boldsymbol{A}_2 \dots \boldsymbol{A}_{i-1})^{-1} \end{aligned} \tag{5.13.12}$$

and is thus of the form

$$\boldsymbol{C}_i = \begin{bmatrix} \tilde{\boldsymbol{c}}_i & \boldsymbol{v}_i \\ 0^T & 0 \end{bmatrix} \tag{5.13.13}$$

Alike equation (5.13.2), it contains six independent terms which we can combine in the vector

$$\boldsymbol{d}_i = \begin{bmatrix} \boldsymbol{c}_i \\ \boldsymbol{v}_i + \tilde{\boldsymbol{c}}_i \boldsymbol{r} \end{bmatrix} \tag{5.13.14}$$

The jacobian matrix describing the instantaneous kinematics in configuration $\boldsymbol{T}$ may then be obtained from the relationship

$$\delta \boldsymbol{x} = \sum_{i=1}^{m} \boldsymbol{d}_i \delta q_i \tag{5.13.15}$$

In terms of the individual matrices (5.13.14), its full expression is

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{c}_1 & \boldsymbol{c}_2 & \dots & \boldsymbol{c}_m \\ \boldsymbol{v}_1 + \tilde{\boldsymbol{c}}_1 \boldsymbol{r} & \boldsymbol{v}_2 + \tilde{\boldsymbol{c}}_2 \boldsymbol{r} & \dots & \boldsymbol{v}_m + \tilde{\boldsymbol{c}}_m \boldsymbol{r} \end{bmatrix} \tag{5.13.16}$$

In order to interpret geometrically the result (5.13.16), let us consider separately the case of revolute and prismatic joints.

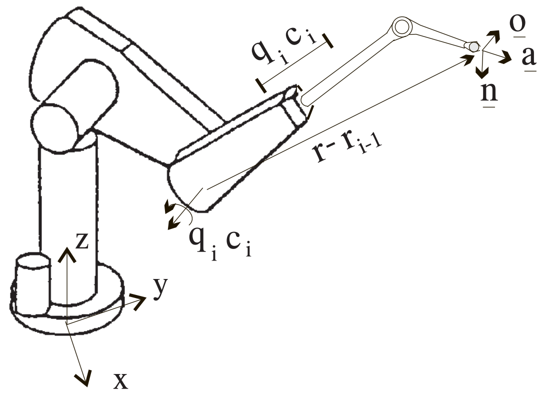


Figure 5.13.1: Geometric meaning of revolute and prismatic joint contributions to jacobian matrix

- *For a revolute joint*, $\boldsymbol{\Delta}_i$ is given by equation (5.13.9) and $\boldsymbol{C}_i$ may be written in the form

$$\boldsymbol{C}_i =^0 \boldsymbol{A}_{i-1} \cdot \boldsymbol{\Delta}_i \cdot^0 \boldsymbol{A}_{i-1}^{-1} \tag{5.13.17}$$

If $\boldsymbol{R}_{i-1}$ is the rotation operator from reference frame to frame attached to link $i-1$, then

$$\tilde{\boldsymbol{c}}_i = \boldsymbol{R}_{i-1} \cdot \tilde{\boldsymbol{k}} \cdot \boldsymbol{R}_{i-1}^T \tag{5.13.18}$$

which means that the direction $\boldsymbol{c}_i$ is the direction of joint axis i in global coordinates, and

$$\boldsymbol{v}_i = -\tilde{\boldsymbol{c}}_i \boldsymbol{r}_{i-1} \tag{5.13.19}$$

The contribution to $\boldsymbol{J}$ of a revolute joint is thus

$$\boldsymbol{d}_i = \begin{bmatrix} \boldsymbol{c}_i \\ \tilde{\boldsymbol{c}}_i(\boldsymbol{r} - \boldsymbol{r}_{i-1}) \end{bmatrix} \tag{5.13.20}$$

where the translation part represents the cross-product of the rotation direction times the relative position of the effector in frame $i-1$.

- *For a prismatic joint*, equation (5.13.10) does not generate any rotation contribution, and

$$\boldsymbol{v}_i = \boldsymbol{R}_{i-1}\boldsymbol{k} = \boldsymbol{c}_i \tag{5.13.21}$$

is the direction of the joint axis in global coordinates. Therefore,

$$\boldsymbol{d}_i = \begin{bmatrix} 0 \\ \boldsymbol{c}_i \end{bmatrix} \tag{5.13.22}$$

5.14 Jacobian matrix of Puma manipulator

The Jacobian matrix $\boldsymbol{J}$ relates the linear and angular velocities of the hand system to the velocities of the joints.

$$\begin{bmatrix} \omega \\ \boldsymbol{v} \end{bmatrix} = \boldsymbol{J}(\boldsymbol{q}) = [\boldsymbol{J}_1(\boldsymbol{q}), \boldsymbol{J}_2(\boldsymbol{q}), \ldots \boldsymbol{J}_6(\boldsymbol{q})]\, \dot{\boldsymbol{q}}(t) \tag{5.14.1}$$

Because PUMA robot is 6 dof robot, the Jacobian matrix is a 6×6 matrix whose ith column vector $\boldsymbol{J}_i(\boldsymbol{q})$ is given by:

$$\boldsymbol{J}_i(\boldsymbol{q}) = \begin{cases} \begin{bmatrix} \boldsymbol{z}_{i-1} \\ \boldsymbol{z}_{i-1} \times \boldsymbol{p}_{(i-1),6} \end{bmatrix} & \text{if joint i is rotational,} \\ \\ \begin{bmatrix} \mathbf{0} \\ \boldsymbol{z}_{i-1} \end{bmatrix} & \text{if joint i is translational.} \end{cases} \tag{5.14.2}$$

The vector $\dot{\boldsymbol{q}}(t)$ is the joint velocity vector of the manipulator while $\boldsymbol{p}_{(i-1),6}$ is the position of the hand coordinate frame from the (i-1)th coordinate frame expressed in the base coordinate system and $\boldsymbol{z}_{i-1}$ is the unit vector along the axis of motion of the joint i expressed in the base frame as well.

For the Puma manipulator which is a 6R type robot, we have:

$$\boldsymbol{J}(\theta) = \begin{bmatrix} \boldsymbol{z}_0 & \boldsymbol{z}_1 & \boldsymbol{z}_2 & \boldsymbol{z}_3 & \boldsymbol{z}_4 & \boldsymbol{z}_5 \\ \boldsymbol{z}_0 \times \boldsymbol{p}_{0,6} & \boldsymbol{z}_1 \times \boldsymbol{p}_{1,6} & \boldsymbol{z}_2 \times \boldsymbol{p}_{2,6} & \boldsymbol{z}_3 \times \boldsymbol{p}_{3,6} & \boldsymbol{z}_4 \times \boldsymbol{p}_{4,6} & \boldsymbol{z}_5 \times \boldsymbol{p}_{5,6} \end{bmatrix} \tag{5.14.3}$$

For the PUMA, we use the results of direct kimenatic model to determine the different column vectors of the Jacobian matrix.

Joint 1

The contribution of joint 1 is quite easy to determine from ${}^0\boldsymbol{A}_6$. The direction of joint axis 1 is given by $\boldsymbol{z}_0$. As coordinate 0 is the reference frame, we have:

$$\boldsymbol{z}_0 = \boldsymbol{k} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

The translation part is given by:

$$\boldsymbol{z}_0 \times \boldsymbol{p}_{0,6}$$

with $\boldsymbol{p}_{0,6}$ which is the translation part of ${}^0\boldsymbol{A}_6$.

$${}^0\boldsymbol{p}_6 = \begin{bmatrix} C_1[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] - S_1(S_4S_5d_6 + d_2) \\ S_1[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] + C_1(S_4S_5d_6 + d_2) \\ (-S_{23}C_4S_5 + C_{23}C_5)d_6 + C_{23}d_4 - a_2S_2 \end{bmatrix} = \begin{bmatrix} ({}^0\boldsymbol{p}_6)_x \\ ({}^0\boldsymbol{p}_6)_y \\ ({}^0\boldsymbol{p}_6)_z \end{bmatrix}$$

It comes:

$$\boldsymbol{k} \times {}^0\boldsymbol{p}_6 = \tilde{\boldsymbol{k}}\, {}^0\boldsymbol{p}_6 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} ({}^0\boldsymbol{p}_6)_x \\ ({}^0\boldsymbol{p}_6)_y \\ ({}^0\boldsymbol{p}_6)_z \end{bmatrix} = \begin{bmatrix} -({}^0\boldsymbol{p}_6)_y \\ ({}^0\boldsymbol{p}_6)_x \\ 0 \end{bmatrix}$$

Therefore we have:

$$\boldsymbol{J}_1(\theta) = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -S_1[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] - C_1(S_4S_5d_6 + d_2) \\ C_1[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] - S_1(S_4S_5d_6 + d_2) \\ 0 \end{bmatrix} \tag{5.14.4}$$

Joint 2

Determining the orientation of joint vector 2, i.e. axis $\boldsymbol{z}_1$, requires the rotation of body system 1 with respect to base system ${}^0\boldsymbol{R}_1$ which is given by the rotation part of ${}^0\boldsymbol{A}_1$. Thus we calculate the absolute direction of the revolute joint axis:

$$\boldsymbol{z}_1 = {}^0\boldsymbol{R}_1 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} C_1 & 0 & -S_1 \\ S_1 & 0 & C_1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix}$$

For the translation component $\boldsymbol{z}_1 \times \boldsymbol{p}_{1,6}$, we need the position of the hand system with respect to body system 1: $\boldsymbol{p}_{1,6}$. This vector $\boldsymbol{p}_{1,6}$ can be extracted from the translation part of ${}^1\boldsymbol{A}_6$, but ${}^1\boldsymbol{p}_6$ is expressed in local coordinates of system 1: O_1, x_1, y_1, z_1. Then we have two possibilities. The first one is to express the two vectors in the reference coordinate system 0 (i.e. $\boldsymbol{p}_{1,6} = {}^0\boldsymbol{R}_1\, {}^1\boldsymbol{p}_6$) and then calculate the cross product $\boldsymbol{z}_1 \times \boldsymbol{p}_{1,6}$. The second alternative, which is generally the easier one) is to determine the product in local frame 1 (i.e. $\boldsymbol{k} \times {}^1\boldsymbol{p}_6$) and then express the result in the base frame with rotation matrix ${}^0\boldsymbol{R}_1$. The second method is chosen. It comes:

$$\boldsymbol{z}_1 \times \boldsymbol{p}_{1,6} = {}^0\boldsymbol{R}_1(\boldsymbol{k} \times {}^1\boldsymbol{p}_6)$$

$${}^1\boldsymbol{p}_6 = \begin{bmatrix} C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2 \\ S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6) + a_2S_2 \\ S_4S_5d_6 + d_2 \end{bmatrix}$$

$$\begin{aligned}\boldsymbol{z}_1 \times \boldsymbol{p}_{1,6} &= \begin{bmatrix} C_1 & 0 & -S_1 \\ S_1 & 0 & C_1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} -(S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6) + a_2S_2) \\ C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} -C_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6) + a_2S_2] \\ -S_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6) + a_2S_2] \\ -[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] \end{bmatrix}\end{aligned}$$

Therefore, $\boldsymbol{J}_2$ is given by:

$$\boldsymbol{J}_2(\theta) = \begin{bmatrix} -S_1 \\ C_1 \\ 0 \\ -C_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6) + a_2S_2] \\ -S_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6) + a_2S_2] \\ -[C_{23}C_4S_5d_6 + S_{23}(d_4 + C_5d_6) + a_2C_2] \end{bmatrix} \tag{5.14.5}$$

Joint 3

For joint 3, the procedure is the same, but we have to calculate ${}^0\boldsymbol{R}_2 = {}^0\boldsymbol{R}_1\, {}^1\boldsymbol{R}_2$ at first.

$$\begin{aligned}{}^0\boldsymbol{R}_2 &= {}^0\boldsymbol{R}_1\, {}^1\boldsymbol{R}_2 \\ &= \begin{bmatrix} C_1 & 0 & -S_1 \\ S_1 & 0 & C_1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} C_2 & -S_2 & 0 \\ S_2 & C_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_1C_2 & -C_1S_2 & -S_1 \\ S_1C_2 & -S_1S_2 & C_1 \\ -S_2 & -C_2 & 0 \end{bmatrix}\end{aligned}$$

The joint axis direction 2 in the base frame is then

$$\boldsymbol{z}_2 = {}^0\boldsymbol{R}_2\, \boldsymbol{k} = \begin{bmatrix} -S_1 \\ C_1 \\ 0 \end{bmatrix}$$

The translational contribution is given by:

$$\boldsymbol{z}_2 \times \boldsymbol{p}_{2,6} \;=\; {}^0\boldsymbol{R}_2(\tilde{\boldsymbol{k}}\,{}^2\boldsymbol{p}_6)$$

We have

$$^2\boldsymbol{p}_6 = \begin{bmatrix} C_3C_4S_5d_6 + S_3(d_4 + C_5d_6) \\ S_3C_4S_5d_6 - C_3(d_4 + C_5d_6) \\ S_4S_5d_6 \end{bmatrix}$$

$$\boldsymbol{k} \times {}^2\boldsymbol{p}_6 = \begin{bmatrix} -[S_3C_4S_5d_6 - C_3(d_4 + C_5d_6)] \\ C_3C_4S_5d_6 + S_3(d_4 + C_5d_6) \\ 0 \end{bmatrix}$$

$$\begin{aligned} \boldsymbol{z}_2 \times \boldsymbol{p}_{2,6} &= \begin{bmatrix} C_1C_2 & -C_1S_2 & -S_1 \\ S_1C_2 & -S_1S_2 & C_1 \\ -S_2 & -C_2 & 0 \end{bmatrix} \begin{bmatrix} -[S_3C_4S_5d_6 - C_3(d_4 + C_5d_6)] \\ C_3C_4S_5d_6 + S_3(d_4 + C_5d_6) \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} -C_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6)] \\ -S_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6)] \\ C_{23}C_4S_5d_6 - S_{23}(d_4 + C_5d_6)] \end{bmatrix} \end{aligned}$$

And $\boldsymbol{J}_3$ is given by:

$$\boldsymbol{J}_3(\theta) \;=\; \begin{bmatrix} -S_1 \\ C_1 \\ 0 \\ -C_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6)] \\ -S_1[S_{23}C_4S_5d_6 - C_{23}(d_4 + C_5d_6)] \\ C_{23}C_4S_5d_6 - S_{23}(d_4 + C_5d_6)] \end{bmatrix} \tag{5.14.6}$$

Joint 4

Contribution of joint 4 to Jacobian matrix is:

$$\boldsymbol{J}_4 \;=\; \begin{bmatrix} \boldsymbol{z}_3 \\ \boldsymbol{z}_3 \times \boldsymbol{p}_{3,6} \end{bmatrix}$$

At first we calculate the direction of the joint axis $\boldsymbol{z}_3$. It comes:

$$\begin{aligned} {}^0\boldsymbol{R}_3 &= {}^0\boldsymbol{R}_2\,{}^2\boldsymbol{R}_3 \\ &= \begin{bmatrix} C_1C_2 & -C_1S_2 & -S_1 \\ S_1C_2 & -S_1S_2 & C_1 \\ -S_2 & -C_2 & 0 \end{bmatrix} \begin{bmatrix} C_3 & 0 & S_3 \\ S_3 & 0 & -C_3 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} C_1C_{23} & -S_1 & C_1S_{23} \\ S_1C_{23} & C_1 & S_1S_{23} \\ -S_{23} & 0 & C_{23} \end{bmatrix} \end{aligned}$$

The joint axis direction in the base frame is then

$$\boldsymbol{z}_3 = {}^0\boldsymbol{R}_3\,\boldsymbol{k} \;=\; \begin{bmatrix} C_1S_{23} \\ S_1S_{23} \\ C_{23} \end{bmatrix}$$

The translational contribution is given by:

$$\begin{aligned}
\boldsymbol{z}_3 \times \boldsymbol{p}_{3,6} &= {}^0\boldsymbol{R}_3(\boldsymbol{k} \times {}^3\boldsymbol{p}_6) = {}^0\boldsymbol{R}_3 \begin{bmatrix} -C_4S_5d_6 \\ S_4S_5d_6 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} C_1C_{23} & -S_1 & C_1S_{23} \\ S_1C_{23} & C_1 & S_1S_{23} \\ -S_{23} & 0 & C_{23} \end{bmatrix} \begin{bmatrix} -S_4S_5d_6 \\ C_4S_5d_6 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} -C_1C_{23}S_4S_5d_6 - S_1C_4S_5d_6 \\ -S_1C_{23}S_4S_5d_6 + C_1C_4S_5d_6 \\ s_{23}S_4S_5d_6 \end{bmatrix}
\end{aligned}$$

Finally contribution of joint 4 is given by:

$$\boldsymbol{J}_4(\theta) = \begin{bmatrix} C_1S_{23} \\ S_1S_{23} \\ C_{23} \\ -C_1C_{23}S_4S_5d_6 - S_1C_4S_5d_6 \\ -S_1C_{23}S_4S_5d_6 + C_1C_4S_5d_6 \\ s_{23}S_4S_5d_6 \end{bmatrix} \tag{5.14.7}$$

Joint 5

Joint 5 contributes to Jacobian matrix with:

$$\boldsymbol{J}_5 = \begin{bmatrix} \boldsymbol{z}_4 \\ \boldsymbol{z}_4 \times \boldsymbol{p}_{4,6} \end{bmatrix}$$

The direction of the joint axis 5 is given by $\boldsymbol{z}_4$. It comes:

$$\begin{aligned}
{}^0\boldsymbol{R}_4 &= {}^0\boldsymbol{R}_3\,{}^3\boldsymbol{R}_4 \\
&= \begin{bmatrix} C_1C_{23} & -S_1 & C_1S_{23} \\ S_1C_{23} & C_1 & S_1S_{23} \\ -S_{23} & 0 & C_{23} \end{bmatrix} \begin{bmatrix} C_4 & 0 & -S_4 \\ S_4 & 0 & C_4 \\ 0 & -1 & 0 \end{bmatrix} \\
&= \begin{bmatrix} C_1C_{23}C_4 - S_1S_4 & -C_1S_{23} & -C_1C_{23}S_4 - S_1C_4 \\ S_1C_{23}C_4 + C_1S_4 & -S_1S_{23} & S_1C_{23}S_4 + C_1C_4 \\ -S_{23}C_4 & C_{23} & S_{23}S_4 \end{bmatrix}
\end{aligned}$$

and the direction of joint axis 5 in the base frame is then

$$\boldsymbol{z}_4 =^0 \boldsymbol{R}_3\,\boldsymbol{k} = \begin{bmatrix} -C_1C_{23}S_4 - S_1C_4 \\ S_1C_{23}S_4 + C_1C_4 \\ S_{23}S_4 \end{bmatrix}$$

The translational contribution of the joint is given by:

$$\begin{aligned}
\boldsymbol{z}_4 \times \boldsymbol{p}_{4,6} &= {}^0\boldsymbol{R}_4(\boldsymbol{k} \times {}^4\boldsymbol{p}_6) = {}^0\boldsymbol{R}_4 \begin{bmatrix} -C_5d_6 \\ S_5d_6 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} C_1C_{23}C_4 - S_1S_4 & -C_1S_{23} & -C_1C_{23}S_4 - S_1C_4 \\ S_1C_{23}C_4 + C_1S_4 & -S_1S_{23} & S_1C_{23}S_4 + C_1C_4 \\ -S_{23}C_4 & C_{23} & S_{23}S_4 \end{bmatrix} \begin{bmatrix} -C_5d_6 \\ S_5d_6 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} (-C_1C_{23}C_5 - S_1S_5)d_6 \\ (-S_1C_{23}C_5 + C_1S_5)d_6 \\ s_{23}C_5d_6 \end{bmatrix}
\end{aligned}$$

Finally contribution of joint 5 is given by:

$$\boldsymbol{J}_5(\theta) \;=\; \begin{bmatrix} -C_1C_{23}S_4 - S_1C_4 \\ S_1C_{23}S_4 + C_1C_4 \\ S_{23}S_4 \\ (-C_1C_{23}C_5 - S_1S_5)d_6 \\ (-S_1C_{23}C_5 + C_1S_5)d_6 \\ s_{23}C_5d_6 \end{bmatrix} \tag{5.14.8}$$

Joint 6

The 6th column of the Jacobian matrix is:

$$\boldsymbol{J}_6 \;=\; \begin{bmatrix} \boldsymbol{z}_5 \\ \boldsymbol{z}_5 \times\; \boldsymbol{p}_{5,6} \end{bmatrix}$$

We calculate at first the direction of the joint axis 6:

$$\begin{aligned} {}^0\boldsymbol{R}_5 \;&=\; {}^0\boldsymbol{R}_4\, {}^4\boldsymbol{R}_5 \\ &=\; \begin{bmatrix} C_1C_{23}C_4 - S_1S_4 & -C_1S_{23} & -C_1C_{23}S_4 - S_1C_4 \\ S_1C_{23}C_4 + C_1S_4 & -S_1S_{23} & S_1C_{23}S_4 + C_1C_4 \\ -S_{23}C_4 & C_{23} & S_{23}S_4 \end{bmatrix} \begin{bmatrix} C_5 & 0 & S_5 \\ S_5 & 0 & -C_5 \\ 0 & 1 & 0 \end{bmatrix} \\ &=\; \begin{bmatrix} [C_1C_{23}C_4 - S_1S_4]C_5 - C_1S_{23}S_5 & -C_1C_{23}S_4 - S_1C_4 & [C_1C_{23}C_4 - S_1S_4]S_5 + C_1S_{23}C_5 \\ [S_1C_{23}C_4 + C_1S_4]C_5 + S_1S_{23}S_5 & S_1C_{23}S_4 + C_1C_4 & [S_1C_{23}C_4 + C_1S_4]S_5 + S_1S_{23}C_5 \\ -S_{23}C_4C_5 - C_{23}S_5 & S_{23}S_4 & -S_{23}C_4S_5 + C_{23}C_5 \end{bmatrix} \end{aligned}$$

and the direction of joint axis 6 in the base frame is then

$$\boldsymbol{z}_5 ={}^0 \boldsymbol{R}_5\, \boldsymbol{k} \;=\; \begin{bmatrix} C_1C_{23}C_4S_5 - S_1S_4S_5 + C_1S_{23}C_5 \\ S_1C_{23}C_4S_5 + C_1S_4S_5 + S_1S_{23}C_5 \\ -S_{23}C_4S_5 + C_{23}C_5 \end{bmatrix}$$

The translational contribution of the joint is given by:

$$\boldsymbol{z}_5 \times\; \boldsymbol{p}_{5,6} \;=\; {}^0\boldsymbol{R}_5(\boldsymbol{k} \times\; {}^5\boldsymbol{p}_6) \;=\; \mathbf{o}$$

Finally contribution of joint 5 is given by:

$$\boldsymbol{J}_6(\theta) \;=\; \begin{bmatrix} C_1C_{23}C_4S_5 - S_1S_4S_5 + C_1S_{23}C_5 \\ S_1C_{23}C_4S_5 + C_1S_4S_5 + S_1S_{23}C_5 \\ -S_{23}C_4S_5 + C_{23}C_5 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.14.9}$$

5.15 Effective numerical solution of inverse problem

Let us consider the problem of finding $(q_1,\; \dots q_m)$ such that

$$\boldsymbol{T} = \boldsymbol{A}_1(q_1) \cdot \boldsymbol{A}_2(q_2) \dots \boldsymbol{A}_m(q_m) \tag{5.15.1}$$

starting from an approximate solution $(q_1^*,\; \dots q_m^*)$.

First, let us construct the matrix $\boldsymbol{T}^*$ corresponding to the given configuration

$$\boldsymbol{T}^* = \boldsymbol{A}_1(q_1^*) \cdot \boldsymbol{A}_2(q_2^*) \ldots \boldsymbol{A}_m(q_m^*) \tag{5.15.2}$$

The residual error on the transformation matrix can be defined as

$$\delta\boldsymbol{T} = \boldsymbol{T} - \boldsymbol{T}^* \tag{5.15.3}$$

and a first order expansion of equation (5.15.3) provides the corresponding expression in configuration space

$$\delta\boldsymbol{T} = \sum_{i=1}^{m}(\boldsymbol{C}_i \delta q_i) \cdot \boldsymbol{T} + O(\delta \boldsymbol{q}^2) \tag{5.15.4}$$

with the definition (5.13.12) of $\boldsymbol{C}_i$ matrices.

To the first order, one can thus obtain the joint displacement corrections from the equality

$$\delta\boldsymbol{T} \cdot \boldsymbol{T}^{-1} = \sum_{i=1}^{m}(\boldsymbol{C}_i \delta q_i) \tag{5.15.5}$$

From the left-hand side, six independent infinitesimal quantities can be extracted

$$\begin{cases} \delta\alpha_x & = (\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{32} \simeq -(\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{23} \\ \delta\alpha_y & = (\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{13} \simeq -(\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{31} \\ \delta\alpha_z & = (\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{21} \simeq -(\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{12} \end{cases} \tag{5.15.6}$$

$$\begin{cases} \delta r_x & = (\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{14} - \delta\alpha_z r_y + \delta\alpha_y r_z \\ \delta r_y & = (\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{24} + \delta\alpha_z r_x - \delta\alpha_x r_z \\ \delta r_z & = (\delta\boldsymbol{T}\cdot\boldsymbol{T}^{-1})_{34} - \delta\alpha_y r_x + \delta\alpha_x r_y \end{cases} \tag{5.15.7}$$

in which case

$$\delta\boldsymbol{x}^T = [\delta\alpha_x \ \delta\alpha_y \ \delta\alpha_z \ \delta r_x \ \delta r_y \ \delta r_z] \tag{5.15.8}$$

and the correction equation may then be written in the matrix form

$$\boldsymbol{J}\boldsymbol{q} = \delta\boldsymbol{x} \tag{5.15.9}$$

Equation (5.15.9) is then directly invertible if $n = 6$ and $\boldsymbol{J}$ has maximum rank.

The numerical procedure of figure (5.11.1) remains valid, except that the residual vector has to be evaluated from equations (5.15.6)- (5.15.7).

Remark

It is important noticing that vector $\delta\alpha$, since it is defined in a finite rotation context, is an non integrable quantity. In a classical mechanics sense, α has the meaning of a vector of *quasi-coordinates* and exists only through its variation $\delta\alpha = (\delta\alpha_x \ \delta\alpha_y \ \delta\alpha_z)$ which represents angular increments in the instantaneous configuration.

5.16 Recursive calculation of velocities in absolute coordinates

Our goal is to take advantage of the open-tree architecture described by equation (5.13.6) to compute

$$\dot{\boldsymbol{\Delta}} = \dot{\boldsymbol{T}} \cdot \boldsymbol{T}^{-1} \tag{5.16.1}$$

To this purpose, let us start from the transformation for frame i

$$\boldsymbol{T}_i =^0 \boldsymbol{A}_i =^0 \boldsymbol{A}_1 \cdot^1 \boldsymbol{A}_2 \ldots^{i-1} \boldsymbol{A}_i \tag{5.16.2}$$

describing the configuration of link i into the global frame, and let us define the differential matrix describing its absolute velocities

$$^0\dot{\boldsymbol{\Delta}}_i = \dot{\boldsymbol{T}}_i \cdot \boldsymbol{T}_i^{-1} \tag{5.16.3}$$

Then, the transformation for link $i+1$ is

$$\boldsymbol{T}_{i+1} = \boldsymbol{T}_i \cdot^i \boldsymbol{A}_{i+1} \tag{5.16.4}$$

and

$$\begin{aligned} ^0\dot{\boldsymbol{\Delta}}_{i+1} &= \dot{\boldsymbol{T}}_{i+1} \cdot \boldsymbol{T}_{i+1}^{-1} \\ &= (\boldsymbol{T}_i \cdot^i \dot{\boldsymbol{A}}_{i+1} + \dot{\boldsymbol{T}}_i \cdot^i \boldsymbol{A}_{i+1}) \cdot \boldsymbol{T}_{i+1}^{-1} \end{aligned} \tag{5.16.5}$$

It provides the recursive formula

$$\boxed{^0\dot{\boldsymbol{\Delta}}_{i+1} \ = \ ^0\dot{\boldsymbol{\Delta}}_i + \boldsymbol{T}_i \, ^i\dot{\boldsymbol{\Delta}}_{i+1} \boldsymbol{T}_i^{-1}} \tag{5.16.6}$$

whose meaning, according to the type of joint considered, is the following:

- For a *revolute joint*, the differential matrix in local coordinates is

  $$\dot{\boldsymbol{\Delta}}_{i+1} = \dot{q}_{i+1} \begin{bmatrix} \tilde{\boldsymbol{k}} & 0 \\ 0^T & 0 \end{bmatrix} \tag{5.16.7}$$

  and we have thus

  $$\omega_{i+1} = \omega_i + \dot{q}_{i+1} \boldsymbol{c}_{i+1} \tag{5.16.8}$$

  and

  $$\boldsymbol{v}_{i+1} = \boldsymbol{v}_i - \dot{q}_{i+1} \tilde{\boldsymbol{c}}_{i+1} \boldsymbol{r}_i \tag{5.16.9}$$

  from which one deduces the translational velocity

  $$\dot{\boldsymbol{r}}_{i+1} = \dot{\boldsymbol{r}}_i + \tilde{\omega}_{i+1}(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) \tag{5.16.10}$$

  Equation (5.16.8) means that angular velocity of frame $i+1$ is the angular velocity of frame i augmented by the relative angular velocity produced by joint q_{i+1}. Its linear velocity (5.16.10) is obtained by adding to the linear velocity of frame i the contribution of the rotation of link $i+1$.

- For a *prismatic joint*, the differential matrix in local coordinates is

$$\dot{\boldsymbol{\Delta}}_{i+1} = \dot{q}_{i+1} \begin{bmatrix} 0 & \boldsymbol{k} \\ 0^T & 0 \end{bmatrix} \tag{5.16.11}$$

and we have thus

$$\begin{aligned} \omega_{i+1} &= \omega_i \\ \dot{\boldsymbol{r}}_{i+1} &= \dot{\boldsymbol{r}}_i + \tilde{\omega}_{i+1}(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) + \dot{q}_{i+1}\boldsymbol{c}_{i+1} \end{aligned} \tag{5.16.12}$$

Angular velocity of frame $i+1$ is the same as that of link i, and linear velocity contains two contributions: the relative velocity due to angular motion of link $i+1$ and the relative velocity due to linear motion at joint q_i.

5.17 Recursive calculation of accelerations in absolute coordinates

To compute accelerations recursively, let us similarly compute

$$\ddot{\boldsymbol{\Delta}} = \ddot{\boldsymbol{T}} \cdot \boldsymbol{T}^{-1} \tag{5.17.1}$$

by defining at an intermediate stage the differential matrix describing the absolute accelerations of link i

$${}^0\ddot{\boldsymbol{\Delta}}_i = \ddot{\boldsymbol{T}}_i \cdot \boldsymbol{T}_i^{-1} \tag{5.17.2}$$

Then, the acceleration matrix for link $i+1$ is

$$\begin{aligned} {}^0\ddot{\boldsymbol{\Delta}}_{i+1} &= \ddot{\boldsymbol{T}}_{i+1} \cdot \boldsymbol{T}_{i+1}^{-1} \\ &= (\boldsymbol{T}_i \cdot {}^i\ddot{\boldsymbol{A}}_{i+1} + 2\dot{\boldsymbol{T}}_i \cdot {}^i\dot{\boldsymbol{A}}_{i+1} + \ddot{\boldsymbol{T}}_i \cdot {}^i\boldsymbol{A}_{i+1})\boldsymbol{T}_{i+1}^{-1} \\ &= \ddot{\boldsymbol{T}}_i \cdot \boldsymbol{T}_i^{-1} + \boldsymbol{T}_i \cdot {}^i\ddot{\boldsymbol{A}}_{i+1} \cdot {}^i\boldsymbol{A}_{i+1}^{-1} \cdot \boldsymbol{T}_i^{-1} + 2\,\dot{\boldsymbol{T}}_i \cdot \boldsymbol{T}_i^{-1} \cdot \boldsymbol{T}_i^{-1} \cdot {}^i\dot{\boldsymbol{A}}_{i+1} \cdot {}^i\boldsymbol{A}_{i+1}^{-1} \cdot \boldsymbol{T}_i^{-1} \end{aligned}$$

and can be put in the recursive form

$$\boxed{{}^0\ddot{\boldsymbol{\Delta}}_{i+1} \;=\; {}^0\ddot{\boldsymbol{\Delta}}_i + 2\,{}^0\dot{\boldsymbol{\Delta}}_i \cdot \boldsymbol{T}_i \cdot^i \dot{\boldsymbol{\Delta}}_{i+1} \cdot \boldsymbol{T}_i^{-1} + \boldsymbol{T}_i \cdot^i \ddot{\boldsymbol{\Delta}}_{i+1} \cdot \boldsymbol{T}_i^{-1}} \tag{5.17.3}$$

In order to interpret it, let us consider successively

- *The revolute joint:*

  The differential matrices in local coordinates are

$${}^i\dot{\boldsymbol{\Delta}}_{i+1} \;=\; \dot{q}_{i+1} \begin{bmatrix} \tilde{\boldsymbol{k}} & 0 \\ 0^T & 0 \end{bmatrix} \tag{5.17.4}$$

  and

$${}^i\ddot{\boldsymbol{\Delta}}_{i+1} \;=\; q\ddot{_{i+1}} \begin{bmatrix} \tilde{\boldsymbol{k}} & 0 \\ 0^T & 0 \end{bmatrix} - \dot{q}_{i+1}^2 \begin{bmatrix} \tilde{\boldsymbol{k}}\tilde{\boldsymbol{k}}^T & 0 \\ 0^T & 0 \end{bmatrix} \tag{5.17.5}$$

  For angular accelerations, we have thus the recurrence relationship

$$\beta_{i+1} = \beta_i + 2\dot{q}_{i+1}\tilde{\omega}_i\tilde{\boldsymbol{c}}_{i+1} + \ddot{q}_{i+1}\tilde{\boldsymbol{c}}_{i+1} - \dot{q}_{i+1}^2\tilde{\boldsymbol{c}}_{i+1}\tilde{\boldsymbol{c}}_{i+1}^T$$

with

$$\beta_i = \dot{\tilde{\omega}}_i - \tilde{\omega}_i \tilde{\omega}_i^T \qquad \text{and} \qquad \beta_{i+1} = \dot{\tilde{\omega}}_{i+1} - \tilde{\omega}_{i+1} \tilde{\omega}_{i+1}^T$$

Therefore

$$\dot{\tilde{\omega}}_{i+1} = \dot{\tilde{\omega}}_i + (\tilde{\omega}_{i+1}\tilde{\omega}_{i+1}^T - \tilde{\omega}_i\tilde{\omega}_i^T) + 2\dot{q}_{i+1}\tilde{\omega}_i\tilde{\boldsymbol{c}}_{i+1} + \ddot{q}_{i+1}\tilde{\boldsymbol{c}}_{i+1} - \dot{q}_{i+1}^2\tilde{\boldsymbol{c}}_{i+1}\tilde{\boldsymbol{c}}_{i+1}^T \tag{5.17.6}$$

Making next use of (5.16.7) we get

$$\tilde{\omega}_{i+1}\tilde{\omega}_{i+1}^T - \tilde{\omega}_i\tilde{\omega}_i^T = \dot{q}_{i+1}(\tilde{\omega}_i\tilde{\boldsymbol{c}}_{i+1}^T + \tilde{\boldsymbol{c}}_{i+1}\tilde{\omega}_{i+1}^T) + \dot{q}_{i+1}^2\tilde{\boldsymbol{c}}_{i+1}\tilde{\boldsymbol{c}}_{i+1}^T$$

If we note further that

$$2\tilde{\omega}_i\tilde{\boldsymbol{c}}_{i+1} + \tilde{\omega}_i\tilde{\boldsymbol{c}}_{i+1}^T + \tilde{\boldsymbol{c}}_{i+1}\tilde{\omega}_{i+1}^T = \tilde{\omega}_i\tilde{\boldsymbol{c}}_{i+1} - \tilde{\boldsymbol{c}}_{i+1}\tilde{\omega}_i = \omega_i\boldsymbol{c}_{i+1}^T - \boldsymbol{c}_{i+1}\omega_i^T$$

we obtain the final result

$$\dot{\omega}_{i+1} = \dot{\omega}_i + \dot{q}_{i+1}\tilde{\omega}_i\boldsymbol{c}_{i+1} + \ddot{q}_{i+1}\boldsymbol{c}_{i+1} \tag{5.17.7}$$

Similarly, for the linear part of (5.17.3) by performing the triple products we get the result

$$\boldsymbol{a}_{i+1} = \boldsymbol{a}_i - (2\dot{q}_{i+1}\tilde{\omega}_i + \ddot{q}_{i+1} - \dot{q}_{i+1}^2\tilde{\boldsymbol{c}}_{i+1})\tilde{\boldsymbol{c}}_{i+1}\boldsymbol{r}_i$$

which, by taking account of

$$\boldsymbol{a}_i = \ddot{\boldsymbol{r}}_i - (\dot{\omega}_i - \tilde{\omega}_i\tilde{\omega}_i^T)\boldsymbol{r}_i$$

and (5.17.7), can be put in the final form

$$\ddot{\boldsymbol{r}}_{i+1} = \ddot{\boldsymbol{r}}_i + (\dot{\tilde{\omega}}_{i+1} - \tilde{\omega}_{i+1}\tilde{\omega}_{i+1}^T)(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) \tag{5.17.8}$$

- *The prismatic joint:*

  The differential matrices in local coordinates are

$$ {}^i\dot{\boldsymbol{\Delta}}_{i+1} = \dot{q}_{i+1}\begin{bmatrix} 0 & \boldsymbol{k} \\ 0^T & 0 \end{bmatrix} \qquad \text{and} \qquad {}^i\ddot{\boldsymbol{\Delta}}_{i+1} = q_{i+1}^{\cdot\cdot}\begin{bmatrix} 0 & \boldsymbol{k} \\ 0^T & 0 \end{bmatrix} \tag{5.17.9}$$

  and equation (5.17.3) provides thus the relationships

$$\beta_{i+1} = \beta_i \tag{5.17.10}$$

  and

$$\boldsymbol{v}_{i+1} = \boldsymbol{v}_i + \boldsymbol{c}_{i+1}\ddot{q}_{i+1} + 2\tilde{\omega}_i\boldsymbol{c}_{i+1}\dot{q}_{i+1} \tag{5.17.11}$$

  From (5.17.10), we deduce that the angular acceleration remains unchanged

$$\dot{\omega}_{i+1} = \dot{\omega}_i \tag{5.17.12}$$

  while

$$\ddot{\boldsymbol{r}}_{i+1} = \ddot{\boldsymbol{r}}_i + (\dot{\tilde{\omega}}_i - \tilde{\omega}_i\tilde{\omega}_i^T)(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) + \boldsymbol{c}_{i+1}\ddot{q}_{i+1} + 2\tilde{\omega}_i\boldsymbol{c}_{i+1}\dot{q}_{i+1} \tag{5.17.13}$$

To summarize:

1. The differential matrices verify the relationships

$$
\begin{aligned}
{}^0\dot{\boldsymbol{\Delta}}_{i+1} &= {}^0\dot{\boldsymbol{\Delta}}_i + \boldsymbol{T}_i \cdot {}^i\dot{\boldsymbol{\Delta}}_{i+1} \cdot \boldsymbol{T}_i^{-1} \\
{}^0\ddot{\boldsymbol{\Delta}}_{i+1} &= {}^0\ddot{\boldsymbol{\Delta}}_i + 2\,{}^0\dot{\boldsymbol{\Delta}}_i \cdot \boldsymbol{T}_i \cdot {}^i\dot{\boldsymbol{\Delta}}_{i+1} \cdot \boldsymbol{T}_i^{-1} + \boldsymbol{T}_i \cdot {}^i\ddot{\boldsymbol{\Delta}}_{i+1} \cdot \boldsymbol{T}_i^{-1}
\end{aligned}
\tag{5.17.14}
$$

2. *For a revolute joint*

$$
\begin{aligned}
\omega_{i+1} &= \omega_i + \dot{q}_{i+1}\boldsymbol{c}_{i+1} \\
\dot{\boldsymbol{r}}_{i+1} &= \dot{\boldsymbol{r}}_i + \tilde{\omega}_{i+1}(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) \\
\dot{\omega}_{i+1} &= \dot{\omega}_i + \ddot{q}_{i+1}\boldsymbol{c}_{i+1} + \dot{q}_{i+1}\tilde{\omega}_i\boldsymbol{c}_{i+1} \\
\ddot{\boldsymbol{r}}_{i+1} &= \ddot{\boldsymbol{r}}_i + (\dot{\tilde{\omega}}_{i+1} - \tilde{\omega}_{i+1}\tilde{\omega}_{i+1}^T)(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i)
\end{aligned}
\tag{5.17.15}
$$

3. *For a prismatic joint*

$$
\begin{aligned}
\omega_{i+1} &= \omega_i \\
\dot{\boldsymbol{r}}_{i+1} &= \dot{\boldsymbol{r}}_i + \tilde{\omega}_{i+1}(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) + \dot{q}_{i+1}\boldsymbol{c}_{i+1} \\
\dot{\omega}_{i+1} &= \dot{\omega}_i \\
\ddot{\boldsymbol{r}}_{i+1} &= \ddot{\boldsymbol{r}}_i + (\dot{\tilde{\omega}}_{i+1} - \tilde{\omega}_{i+1}\tilde{\omega}_{i+1}^T)(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i) + \boldsymbol{c}_{i+1}\ddot{q}_{i+1} + 2\tilde{\omega}_{i+1}\boldsymbol{c}_{i+1}\dot{q}_{i+1}
\end{aligned}
\tag{5.17.16}
$$

The forward recursive kinematic equations (5.17.14) to (5.17.16) will form the basis for an efficient formulation of dynamic equations, either in Lagrange or in Euler-Newton form.

5.18 Recursive calculation of velocities in body coordinates

Exactly in the same way as we have computed $\dot{\boldsymbol{\Delta}}$ in section 5.15, let us consider the matrix

$$\dot{\boldsymbol{\Gamma}} = \boldsymbol{T}^{-1} \cdot \dot{\boldsymbol{T}} \tag{5.18.1}$$

It is easy to verify that it has the meaning

$$\dot{\boldsymbol{\Gamma}} = \begin{bmatrix} \tilde{\omega}' & \boldsymbol{v}' \\ 0^T & 0 \end{bmatrix} \tag{5.18.2}$$

where

$$\tilde{\omega}' = \boldsymbol{R}^T\dot{\boldsymbol{R}} \tag{5.18.3}$$

is the matrix of angular velocities expressed in body coordinates, and

$$\boldsymbol{v}' = \boldsymbol{R}^T\dot{\boldsymbol{r}} \tag{5.18.4}$$

represents the linear velocities expressed in the same frame. Let us start again from the differential matrix describing the velocities of frame i

$$\dot{\boldsymbol{\Gamma}}_i = \boldsymbol{T}_i^{-1} \cdot \dot{\boldsymbol{T}}_i \tag{5.18.5}$$

Since the transformation from frame $i+1$ is

$$\boldsymbol{T}_{i+1} = \boldsymbol{T}_i \cdot {}^i\boldsymbol{A}_{i+1} \tag{5.18.6}$$

then

$$\dot{\boldsymbol{\Gamma}}_{i+1} = \boldsymbol{T}_{i+1}^{-1} \left(\boldsymbol{T}_i \cdot {}^i\dot{\boldsymbol{A}}_{i+1} + \dot{\boldsymbol{T}}_i \cdot {}^i\boldsymbol{A}_{i+1}\right)$$

and it provides the result similar to (5.16.5)

$$\dot{\boldsymbol{\Gamma}}_{i+1} \;=\; {}^i\boldsymbol{A}_{i+1}^{-1} \cdot \dot{\boldsymbol{T}}_i \cdot {}^i\boldsymbol{A}_{i+1} + {}^i\boldsymbol{A}_{i+1}^{-1} \cdot {}^i\dot{\boldsymbol{A}}_{i+1}$$

or

$$\boxed{\dot{\boldsymbol{\Gamma}}_{i+1} \;=\; {}^i\boldsymbol{A}_{i+1}^{-1} \left[\dot{\boldsymbol{\Gamma}}_i + {}^i\, \dot{\boldsymbol{\Delta}}_{i+1}\right] {}^i\boldsymbol{A}_{i+1}} \tag{5.18.7}$$

Let us define:

- $\boldsymbol{R}_{i+1}$ the rotation operator from frame i to frame $i+1$
- σ a parameter characterizing the type of joint: $\sigma = 0$ for a revolute joint, $\sigma = 1$ for a prismatic joint
- $\boldsymbol{d}_{i+1}^{\star}$ the vector expressing the position of the origin of frame $i+1$ in frame i (displacement part of ${}^i\boldsymbol{A}_{i+1}$).
- $\omega_{i+1}^{\star}$ the angular velocity of frame $i+1$ expressed in frame i

The angular velocity of frame $i+1$ expressed in frame i is given by

$$\omega_{i+1}^{\star} = \omega_i^{'} + (1-\sigma)\dot{q}_{i+1}\boldsymbol{k} \tag{5.18.8}$$

Then, velocity propagation in frame $i+1$ can be expressed in the form

$$\begin{aligned} \omega_{i+1}' &= \boldsymbol{R}_{i+1}^T \omega_{i+1}^{\star} \\ \boldsymbol{v}_{i+1}' &= \boldsymbol{R}_{i+1}^T [\boldsymbol{v}_i' + \tilde{\omega}_{i+1}^{\star} \boldsymbol{d}_{i+1}^{\star} + \sigma \boldsymbol{k} \dot{q}_{i+1}] \end{aligned} \tag{5.18.9}$$

5.19 Recursive calculation of accelerations in body coordinates

To compute accelerations in body coordinates, we similarly consider the matrix

$$\ddot{\boldsymbol{\Gamma}} = \boldsymbol{T}^{-1} \cdot \ddot{\boldsymbol{T}} \tag{5.19.1}$$

for which it is easy to verify that it has the meaning

$$\ddot{\boldsymbol{\Gamma}} = \begin{bmatrix} \beta' & \boldsymbol{a}' \\ 0^T & 0 \end{bmatrix} \tag{5.19.2}$$

where

$$\boldsymbol{a}' = \boldsymbol{R}^T \ddot{\boldsymbol{r}} \tag{5.19.3}$$

is the acceleration vector in body coordinates, while the angular acceleration $\dot{\omega}'$ can be extracted from the form

$$\beta' = \boldsymbol{R}^T \ddot{\boldsymbol{R}} = \dot{\tilde{\omega}}' - (\tilde{\omega}')^T \tilde{\omega}' \tag{5.19.4}$$

Starting from

$$\ddot{\boldsymbol{\Gamma}}_i = \boldsymbol{T}_i^{-1} \cdot \ddot{\boldsymbol{T}}_i \tag{5.19.5}$$

we obtain

$$\ddot{\boldsymbol{\Gamma}}_{i+1} = \boldsymbol{T}_{i+1}^{-1}(\ddot{\boldsymbol{T}}_i \cdot {}^i\boldsymbol{A}_{i+1} + 2\dot{\boldsymbol{T}}_i \cdot {}^i\dot{\boldsymbol{A}}_{i+1} + \boldsymbol{T}_i \cdot {}^i\ddot{\boldsymbol{A}}_{i+1})$$

which can be put in the recurrence form

$$\boxed{\ddot{\boldsymbol{\Gamma}}_{i+1} = {}^i\boldsymbol{A}_{i+1}^{-1}\,[\ddot{\boldsymbol{\Gamma}}_i + 2\dot{\boldsymbol{\Gamma}}_i \cdot {}^i\dot{\boldsymbol{\Delta}}_{i+1} + {}^i\ddot{\boldsymbol{\Delta}}_{i+1}]\,{}^i\boldsymbol{A}_{i+1}} \tag{5.19.6}$$

It is easy to check that it provides the following recurrence relationships

- The angular acceleration of frame $i+1$ can be expressed in frame i by

$$\dot{\omega}_{i+1}^{\star} = \dot{\omega}_i' + (1-\sigma)[\dot{q}_{i+1}\tilde{\omega}_i' + \ddot{q}_{i+1}]\boldsymbol{k} \tag{5.19.7}$$

- The acceleration propagation may then be expressed in the form

$$\begin{aligned} \dot{\omega}_{i+1}' &= \boldsymbol{R}_{i+1}^T \dot{\omega}_{i+1}^{\star} \\ \boldsymbol{a}_{i+1}' &= \boldsymbol{R}_{i+1}^T[\boldsymbol{a}_i' + (\dot{\tilde{\omega}}_{i+1}^{\star} - (\tilde{\omega}_{i+1}^{\star})^T\tilde{\omega}_{i+1}^{\star})\boldsymbol{d}_{i+1}^{\star}] + \sigma(2\dot{q}_{i+1}\tilde{\omega}_{i+1}' + \ddot{q}_{i+1})\boldsymbol{k} \end{aligned} \tag{5.19.8}$$

Chapter 6

DYNAMICS OF OPEN-TREE SIMPLY-CONNECTED STRUCTURES

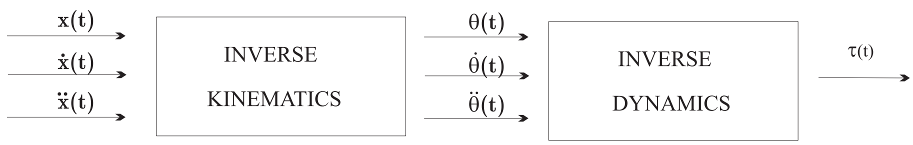


Figure 6.1.1: The inverse dynamic problem

6.1 Introduction

The dynamic equations of motion of an articulated structure describe its dynamic behavior.

They can be used for several purposes:

- computer simulation of the robot arm motion,
- design of suitable control equations,
- evaluation of the dynamic performance of the design.

Just as in kinematics, the problem of robot arm dynamics may be considered from two complementary point-of-views.

- The *inverse dynamics* problem;
- The *direct dynamics* problem.

The inverse dynamics problem

The *inverse dynamics* problem consists, given positions, velocities and accelerations, to compute the forces and torques necessary to generated the prescribed trajectory. It may thus be summarized as presented in figure 6.1.1.

For control purpose, the inverse dynamic model is used to determine the nominal torque to achieve the goal trajectory : the controller is then left with the responsability to compensate for deviations from the nominal trajectory.

The inverse dynamic model is also used at design level to evaluate the power requirements at actuator level.

The direct dynamics problem

The *direct dynamics* problem consists, given an initial state of the system (specified by initial positions and velocities θ_0 and $\dot{\theta}_0$), to predict the trajectory when the torques are specified at the joints.

The dynamic model takes the form of a system of differential equations in time, with coefficients which are configuration dependent

$$\boldsymbol{M}(\theta)\,\ddot{\theta} \;+\; \boldsymbol{A}(\theta)\,\dot{\theta}^2 \;+\; \boldsymbol{G}(\theta) \;=\; \tau(t) \tag{6.1.1}$$

where

- $\boldsymbol{M}(\theta)$ is the inertia matrix of the system,
- $\boldsymbol{A}(\theta)\,\dot{\theta}^2$ represents centrifugal and Coriolis terms,

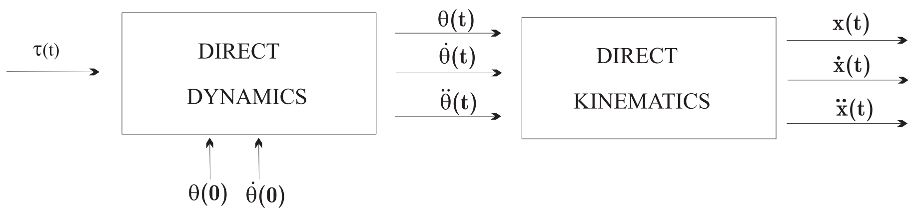


Figure 6.1.2: The direct dynamic problem

- $\boldsymbol{G}(\theta)$ represents the action of gravity on the system,
- $\tau(t)$ represents the external torques applied on the system.

When considered in inverse form, the dynamic problem consists simply in algebraic calculations, since trajectory information can be introduced in equation (6.1.1) to provide as an output torque values $\tau(t)$.

The direct problem (Figure 6.1.2) is much more difficult to solve since it consists of a set of second-order differential equations with non-linear coefficients. Its use is mainly for computer simulation of the dynamics of the system.

Evaluating the coefficients of the dynamic model (6.1.1) involves a significant amount of floating point operations. In order to minimize the associated cost, it is essential to take account of the recursive property of open-tree articulated structures. The problem is still further simplified when the structure is simply-connected, as it occurs in most industrial robot architectures.

Before establishing the dynamic model of an articulated chain, we will establish the *dynamic equilibrium equation of a single link, treated as an isolated rigid body.* They will be expressed in matrix form, using either a body frame or a global frame representation.

We will next build the dynamic model of articulated chain using a *recursive Euler-Newton formalism*: in the same way as a *forward recursion* propagates *kinematic information*, a *backward recursion* propagates the *forces and moments* from the end effector to the base of the manipulator.

In the last part of the chapter, the same recursion concept will be used to establish the equations of motion in Lagrange form.

6.2 Equation of motion of the rigid body

Let us consider a rigid body of volume V (Figure 6.2.1) with its center of mass taken as the origin of the body frame $O'x'y'z'$.

We suppose that it is submitted to a gravity field $\boldsymbol{g}$, and that a force per unit of volume $\boldsymbol{f}$ is acting on it.

Let us adopt Hamilton's principle to describe the dynamics of the system above.

We define:

- $\mathcal{K}$: the kinetic energy of the body,
- $\mathcal{U}$: its potential energy (here, due to gravity),

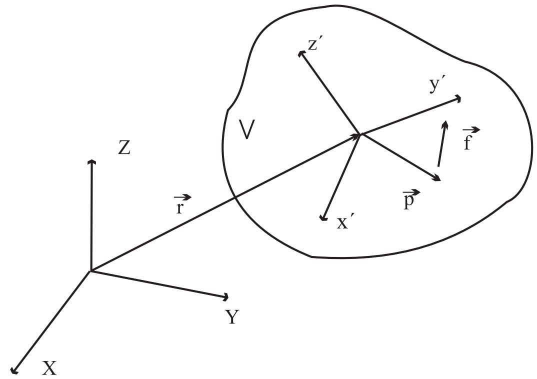


Figure 6.2.1: Kinematics of rigid body

- $\delta\mathcal{W}$: the virtual work of the non-conservative forces acting on the system.

Hamilton's principle states that, between two times t_1 and t_2 at which the position of the system is prescribed, the real trajectory of the system is such that

$$\delta \int_{t_1}^{t_2} (\mathcal{K} - \mathcal{U})\, dt \;+\; \int_{t_1}^{t_2} \delta\mathcal{W}\, dt \;=\; 0 \tag{6.2.1}$$

6.2.1 Potential energy

Since the only form of potential energy considered is gravity, it can be expressed in the form

$$\mathcal{U} \;=\; \int_V \boldsymbol{g}^T \left(\boldsymbol{r} + \boldsymbol{R}\boldsymbol{p}'\right)\; dm \tag{6.2.2}$$

and, since O' is the center of mass

$$\int_V \boldsymbol{p}'\; dm \;=\; 0 \tag{6.2.3}$$

and equation (6.2.2) reduces thus to

$$\mathcal{U} \;=\; m\,\boldsymbol{g}^T\,\boldsymbol{r} \tag{6.2.4}$$

where m is the total mass of the body.

6.2.2 Kinetic energy

The kinetic energy is evaluated by integrating the kinetic energy of individual particles over the volume

$$\mathcal{K} \;=\; \frac{1}{2}\int_V \boldsymbol{v}^T\boldsymbol{v}\; dm \tag{6.2.5}$$

where the velocity at a given point is calculated by

$$\boldsymbol{v} \;=\; \dot{\boldsymbol{r}} \;+\; \dot{\boldsymbol{R}}\boldsymbol{p}' \tag{6.2.6}$$

When substituting (6.2.6) into (6.2.5), the kinetic energy of the rigid body can be decomposed into three terms

$$\mathcal{K} \;=\; \mathcal{K}_1 + \mathcal{K}_2 + \mathcal{K}_3 \tag{6.2.7}$$

$\mathcal{K}_1$ is the *translational kinetic energy*

$$\mathcal{K}_1 \;=\; \frac{1}{2}\int_V \dot{\boldsymbol{r}}^T\dot{\boldsymbol{r}}\; dm \tag{6.2.8}$$

It can be expressed either in global or body coordinates

$$\mathcal{K}_1 \;=\; \frac{1}{2}m\;\dot{\boldsymbol{r}}^T\dot{\boldsymbol{r}} \;=\; \frac{1}{2}m\;\boldsymbol{v}'^T\boldsymbol{v}' \tag{6.2.9}$$

with

$$\boldsymbol{v}' \;=\; \boldsymbol{R}^T\dot{\boldsymbol{r}} \tag{6.2.10}$$

defined as the velocity of O' expressed in body components.

$\mathcal{K}_2$ is the *mutual kinetic energy*

$$\mathcal{K}_2 \;=\; \int_V \dot{\boldsymbol{r}}^T\dot{\boldsymbol{R}}^T\boldsymbol{p}'\; dm \tag{6.2.11}$$

and it vanishes according to (6.2.3).

$$\mathcal{K}_2 \;=\; \dot{\boldsymbol{r}}^T\dot{\boldsymbol{R}}^T\int_V \boldsymbol{p}'\; dm \;=\; 0$$

$\mathcal{K}_3$ is the *rotational kinetic energy*

$$\mathcal{K}_3 \;=\; \frac{1}{2}\int_V \boldsymbol{p}'^T\dot{\boldsymbol{R}}^T\dot{\boldsymbol{R}}\boldsymbol{p}'\; dm \tag{6.2.12}$$

It can be transformed by noticing that

$$\dot{\boldsymbol{R}}^T\dot{\boldsymbol{R}} \;=\; \dot{\boldsymbol{R}}^T\boldsymbol{R}\,\boldsymbol{R}^T\dot{\boldsymbol{R}} \;=\; (\tilde{\omega}')^T\tilde{\omega}' \tag{6.2.13}$$

where

$$\tilde{\omega}' \;=\; \boldsymbol{R}^T\dot{\boldsymbol{R}} \tag{6.2.14}$$

is the angular velocity matrix expressed *in body axes.*

Let us next make use of

$$\tilde{\omega}'\boldsymbol{p}' \;=\; -\tilde{\boldsymbol{p}}'\omega' \tag{6.2.15}$$

to put equation (6.2.12) in the final form

$$\mathcal{K}_3 \;=\; \frac{1}{2}(\omega')^T\boldsymbol{J}'\omega' \tag{6.2.16}$$

where

$$\boldsymbol{J}' \;=\; \int_V (\tilde{\boldsymbol{p}}')^T\tilde{\boldsymbol{p}}'\; dm \;=\; \int_V (\boldsymbol{p}'^2\,\boldsymbol{I} - \boldsymbol{p}'\boldsymbol{p}'^T)\; dm \tag{6.2.17}$$

is the tensor of the rigid body in the body axes. Its explicit expression is:

$$\boldsymbol{J}' \;=\; \int_V \begin{bmatrix} y'^2+z'^2 & -x'y' & -x'z' \\ -x'y' & x'^2+z'^2 & -y'z' \\ -x'z' & -y'z' & x'^2+y'^2 \end{bmatrix}\; dm \tag{6.2.18}$$

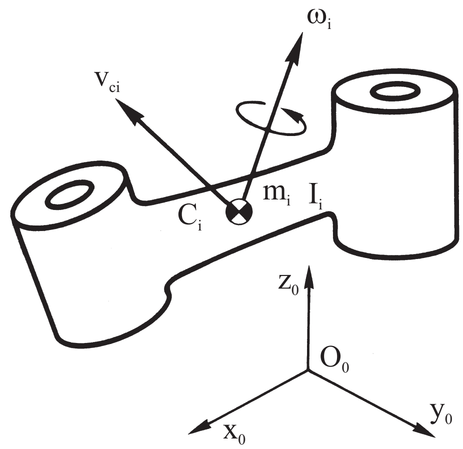


Figure 6.2.2: Dynamics of a isolated link

The rotational kinetic energy can similarly be expressed in the absolute reference frame in the form

$$\mathcal{K}_3 \;=\; \frac{1}{2}\omega^T \boldsymbol{J}\omega \tag{6.2.19}$$

where ω is the angular velocity in the global axes

$$\omega \;=\; \boldsymbol{R}\omega' \tag{6.2.20}$$

and $\boldsymbol{J}$ is the inertia tensor in global axes

$$\boldsymbol{J} \;=\; \boldsymbol{R}\,\boldsymbol{J}'\,\boldsymbol{R}^T \tag{6.2.21}$$

Note that it is time-dependent and that its time derivative can be expressed in the form

$$\dot{\boldsymbol{J}} \;=\; \tilde{\omega}\,\boldsymbol{J} \;-\; \boldsymbol{J}\tilde{\omega} \tag{6.2.22}$$

According to (6.2.9), (6.2.11), (6.2.16) and (6.2.19), the kinetic energy (6.2.7) can thus be expressed in one of the alternate forms

$$\begin{aligned}\mathcal{K} \;&=\; \frac{1}{2}m\dot{\boldsymbol{r}}^T\dot{\boldsymbol{r}} \;+\; \frac{1}{2}\omega^T\boldsymbol{J}\omega \\ &=\; \frac{1}{2}m\boldsymbol{v}'^T\boldsymbol{v}' \;+\; \frac{1}{2}\omega'^T\boldsymbol{J}'\omega'\end{aligned} \tag{6.2.23}$$

6.2.3 Virtual work of external loads

Supposing a force distribution $\boldsymbol{f}$ over the body, the global axis expression of the associated virtual work is

$$\delta\mathcal{W} \;=\; \int_V \boldsymbol{f}^T\,\delta(\boldsymbol{r} + \boldsymbol{R}\,\boldsymbol{p}')\;dV \tag{6.2.24}$$

It can be explicited by making use of the fact that

$$\delta(\boldsymbol{r} + \boldsymbol{R}\boldsymbol{p}') = \delta\boldsymbol{r} + \delta\tilde{\alpha}\,(\boldsymbol{p} - \boldsymbol{r}) \tag{6.2.25}$$

where $\delta\alpha$ is the skew-symetric matrix

$$\delta\tilde{\alpha} \;=\; \delta\boldsymbol{R}\,\boldsymbol{R}^T \tag{6.2.26}$$

Let us make next use of the fact that

$$\delta\tilde{\alpha}\,(\boldsymbol{p}-\boldsymbol{r}) \;=\; -\,\widetilde{(\boldsymbol{p}-\boldsymbol{r})}\,\delta\alpha \tag{6.2.27}$$

to put (6.2.24) in the final form

$$\delta\mathcal{W} \;=\; \boldsymbol{F}^T\delta\boldsymbol{r} + \boldsymbol{N}^T\,\delta\alpha \tag{6.2.28}$$

The vector

$$\boldsymbol{F} \;=\; \int_V \boldsymbol{f}\;dV \tag{6.2.29}$$

is the total force acting on the body, and

$$\boldsymbol{N} \;=\; \int_V \widetilde{(\boldsymbol{p}-\boldsymbol{r})}\boldsymbol{f}\;dV \tag{6.2.30}$$

is the resulting torque about the center of mass.

Equation (6.2.24) can still be expressed in body axes

$$\delta\mathcal{W} \;=\; \boldsymbol{F'}^T\delta\boldsymbol{r'} \;+\; \boldsymbol{N'}^T\delta\alpha' \tag{6.2.31}$$

where

$$\delta\boldsymbol{r'} \;=\; \boldsymbol{R}^T\,\delta\boldsymbol{r} \tag{6.2.32}$$

is the virtual displacement of the center of mass projected into body axes.

$$\boldsymbol{F'} \;=\; \boldsymbol{R}^T\,\boldsymbol{F} \tag{6.2.33}$$

is the resultant force at the center of mass projected into body axes.

$$\delta\tilde{\alpha}' \;=\; \boldsymbol{R}^T\,\delta\boldsymbol{R} \tag{6.2.34}$$

is the matrix of infinitesimal rotations, and

$$\begin{aligned}\boldsymbol{N'} \;&=\; \int_V \boldsymbol{R}\,\widetilde{(\boldsymbol{p}-\boldsymbol{r})}\,\boldsymbol{R}^T\;\boldsymbol{R}\,\boldsymbol{f}\;dV\\ &=\; \int_V \tilde{\boldsymbol{p}}'\boldsymbol{f'}\;dV\end{aligned} \tag{6.2.35}$$

is the torque about the center of mass.

6.2.4 Relationship between angular velocities and quasi-coordinates

In order to express the variation of (6.2.1) where $\delta\omega$ and $\delta\alpha$ appear as separate quantities, it is necessary to recognize that the α parameters are quasi-coordinates in the sense that they are defined only in a differential manner. We need thus to express the relationship existing between $\delta\alpha$ and $\delta\omega$.

To get it, let us start from the definition

$$\delta\tilde{\alpha} \;=\; \delta\boldsymbol{R}\,\boldsymbol{R}^T \qquad \text{and} \qquad \tilde{\omega} \;=\; \dot{\boldsymbol{R}}\boldsymbol{R}^T$$

we obtain on one hand

$$\delta\dot{\tilde{\alpha}} \;=\; \delta\dot{\boldsymbol{R}}\boldsymbol{R}^T \;+\; \delta\boldsymbol{R}\dot{\boldsymbol{R}}^T \tag{6.2.36}$$

and on the other hand

$$\delta\tilde{\omega} \;=\; \delta\dot{\boldsymbol{R}}\boldsymbol{R}^T \;+\; \dot{\boldsymbol{R}}\delta\boldsymbol{R}^T \tag{6.2.37}$$

Equations (6.2.36) and (6.2.37) can be combined in the form

$$\delta\tilde{\omega} \;=\; \delta\dot{\tilde{\alpha}} - \delta\boldsymbol{R}\,\dot{\boldsymbol{R}}^T + \dot{\boldsymbol{R}}\,\delta\boldsymbol{R}^T \tag{6.2.38}$$

or

$$\delta\tilde{\omega} \;=\; \delta\dot{\tilde{\alpha}} \;+\; \delta\tilde{\alpha}\,\tilde{\omega} - \tilde{\omega}\,\delta\tilde{\alpha} \tag{6.2.39}$$

If we note that

$$\delta\tilde{\alpha}\,\tilde{\omega} - \tilde{\omega}\,\delta\tilde{\alpha} \;=\; \omega\,\delta\alpha^T - \delta\alpha\,\omega^T \;=\; -\widetilde{(\tilde{\omega}\,\delta\alpha)} \tag{6.2.40}$$

generates a skew-symetric matrix with components

$$-\widetilde{(\tilde{\omega}\,\delta\alpha)} \;=\; \begin{bmatrix} \omega_z\,\delta\alpha_y - \omega_y\,\delta\alpha_z \\ \omega_x\,\delta\alpha_z - \omega_z\,\delta\alpha_x \\ \omega_y\,\delta\alpha_x - \omega_x\,\delta\alpha_y \end{bmatrix}$$

we obtain that the associated vectors are related by

$$\delta\omega \;=\; \delta\dot{\alpha} \;-\; \tilde{\omega}\,\delta\alpha \tag{6.2.41}$$

Similarly, starting from

$$\delta\tilde{\alpha}' \;=\; \boldsymbol{R}^T\,\delta\boldsymbol{R} \qquad \text{and} \qquad \omega' \;=\; \boldsymbol{R}^T\dot{\boldsymbol{R}} \tag{6.2.42}$$

we would obtain

$$\delta\tilde{\omega}' \;=\; \delta\dot{\tilde{\alpha}}' + \tilde{\omega}'\,\delta\tilde{\alpha}' - \delta\tilde{\alpha}'\tilde{\omega}' \tag{6.2.43}$$

This can be further written as

$$\delta\tilde{\omega}' \;=\; \delta\dot{\tilde{\alpha}}' + \widetilde{(\tilde{\omega}'\,\delta\alpha')}$$

and this gives the result:

$$\delta\omega' \;=\; \delta\dot{\alpha}' + \tilde{\omega}'\,\delta\alpha' \tag{6.2.44}$$

6.2.5 Equations of motion

In order to express the dynamic equilibrium of the rigid body, let us start first with global coordinates where the variation of the kinetic energy is

$$\delta\mathcal{K} \;=\; m\,\dot{\boldsymbol{r}}^T\,\delta\dot{\boldsymbol{r}} \;+\; \omega^T\,\boldsymbol{J}\,\delta\omega \;+\; \frac{1}{2}\,\omega^T\,\delta\boldsymbol{J}\,\omega$$

The tensor of inertia $\boldsymbol{J}$ being configuration dependent, its variation can be obtained in the form

$$\delta\boldsymbol{J} \;=\; \delta\tilde{\alpha}\,\boldsymbol{J} \;-\; \boldsymbol{J}\,\delta\tilde{\alpha} \tag{6.2.45}$$

and owing to (6.2.41) and (6.2.45), $\delta\mathcal{K}$ can be written in the form

$$\delta\mathcal{K} \;=\; m\,\dot{\boldsymbol{r}}^T\,\delta\dot{\boldsymbol{r}} \;+\; \omega^T\boldsymbol{J}(\delta\dot{\alpha} - \tilde{\omega}\delta\alpha) \;+\; \frac{1}{2}\omega^T(\delta\tilde{\alpha}\,\boldsymbol{J} \;-\; \boldsymbol{J}\,\delta\tilde{\alpha})\,\omega \tag{6.2.46}$$

After some manipulations we get the simplified expression

$$\delta\mathcal{K} \;=\; m\,\dot{\boldsymbol{r}}^T\,\delta\dot{\boldsymbol{r}} \;+\; \omega^T\,\boldsymbol{J}\,\delta\dot{\alpha} \tag{6.2.47}$$

Similarly

$$\delta\mathcal{U} \;=\; m\boldsymbol{g}^T\delta\boldsymbol{r} \tag{6.2.48}$$

and

$$\delta\mathcal{W} \;=\; \boldsymbol{F}^T\,\delta\boldsymbol{r} \;+\; \boldsymbol{N}^T\,\delta\alpha \tag{6.2.49}$$

Therefore, owing to equation (6.2.1)

$$\int_{t_1}^{t_2}[m\,\dot{\boldsymbol{r}}^T\delta\dot{\boldsymbol{r}} \;+\; \omega^T\boldsymbol{J}\delta\dot{\alpha} - m\,\boldsymbol{g}^T\delta\boldsymbol{r} \;+\; \boldsymbol{F}^T\delta\boldsymbol{r} \;+\; \boldsymbol{N}^T\delta\alpha]\,dt \;=\; 0 \tag{6.2.50}$$

Let us integrate by parts the terms in $\delta\dot{\boldsymbol{r}}$ and $\delta\dot{\alpha}$

$$\left[\omega^T\,\boldsymbol{J}\,\delta\alpha \;+\; m\dot{\boldsymbol{r}}^T\delta\boldsymbol{r}\right]_{t_1}^{t_2} \tag{6.2.51}$$

$$+\;\int_{t_1}^{t_2}\left\{\delta\boldsymbol{r}^T(-m\ddot{\boldsymbol{r}} - m\,\boldsymbol{g} + \boldsymbol{F}) \;+\; \delta\alpha^T\left[\boldsymbol{N} - \frac{d}{dt}(\boldsymbol{J}\omega)\right]\right\}dt \;=\; 0 \tag{6.2.52}$$

Since the end-conditions are prescribed ($\delta\boldsymbol{r} = 0$ and $\delta\alpha = 0$ in $t = t_1$ and $t = t_2$), the boundary term in (6.2.51) vanishes.

The arbitrariness of $\delta\boldsymbol{r}$ and $\delta\alpha$ provides the dynamic equilibrium equations in global coordinates:

$$\boxed{m\ddot{\boldsymbol{r}} \;=\; -\,m\boldsymbol{g} \;+\; \boldsymbol{F}} \tag{6.2.53}$$

and

$$\boxed{\frac{d}{dt}(\boldsymbol{J}\omega) \;=\; \boldsymbol{N}} \tag{6.2.54}$$

Equation (6.2.22) can be used to transform (6.2.54) into

$$\boldsymbol{J}\dot{\omega} \;+\; (\tilde{\omega}\boldsymbol{J} - \boldsymbol{J}\tilde{\omega})\,\omega \;=\; \boldsymbol{N} \tag{6.2.55}$$

or

$$\boxed{\boldsymbol{J}\,\dot{\omega} \;+\; \tilde{\omega}\boldsymbol{J}\omega \;=\; \boldsymbol{N}} \tag{6.2.56}$$

To express translational equilibrium in body coordinates, the easiest is to premultiply equation (6.2.53) by $\boldsymbol{R}^T$, in which case one obtains the result

$$\boxed{m\boldsymbol{a}' \;=\; -\,m\,\boldsymbol{g}' + \boldsymbol{F}'} \tag{6.2.57}$$

where

$$\boxed{\boldsymbol{a}' \;=\; \boldsymbol{R}^T\ddot{\boldsymbol{r}}} \tag{6.2.58}$$

is the *acceleration vector in body axes.*

For the rotational part, let us make use of the fact that

$$\delta\mathcal{K}_3 \;=\; \omega'^T\,\boldsymbol{J}'\,(\delta\dot{\alpha}' \;+\; \tilde{\omega}'\delta\alpha') \tag{6.2.59}$$

in which case

$$\int_{t_1}^{t_2}\delta\mathcal{K}_3\,dt \;=\; \left[\delta\alpha'^T\,\boldsymbol{J}'\,\omega'\right]_{t_1}^{t_2} - \int_{t_1}^{t_2}\delta\alpha'^T(\boldsymbol{J}'\dot{\omega}' \;+\; \tilde{\omega}'\boldsymbol{J}'\omega')\;dt \tag{6.2.60}$$

and rotational equilibrium expresses thus in the body axis form

$$\boxed{\boldsymbol{J}'\dot{\omega}' \;+\; \tilde{\omega}'\boldsymbol{J}'\omega' \;=\; \boldsymbol{N}'} \tag{6.2.61}$$

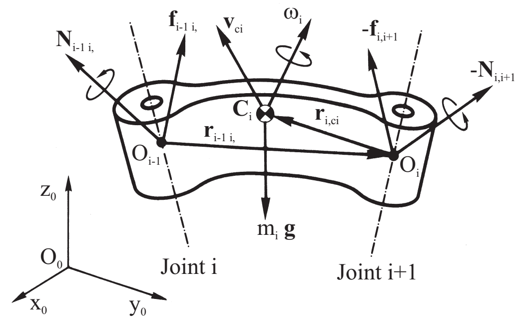


Figure 6.3.1: Dynamics of a binary link

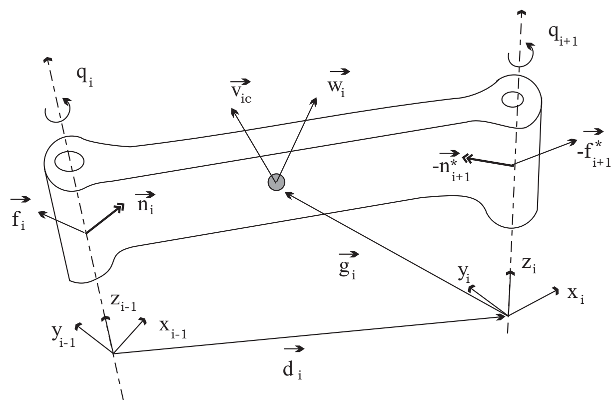


Figure 6.3.2: Geometry of binary link

6.3 Recursive Newton-Euler technique for simple open-tree structures

Let us consider member i of an open-tree simply connected articulated structure, and define

- ω_i' the angular velocity of link i expressed in the local frame $\boldsymbol{T}_i$;
- $\boldsymbol{v}_{ic}'$ the linear velocity of its center of mass in the local frame $\boldsymbol{T}_i$;
- $\boldsymbol{a}_{ic}'$ its acceleration in the local frame $\boldsymbol{T}_i$;
- $\boldsymbol{n}_{i+1}'$ the torque exerced by link i on link $i+1$, projected in frame $\boldsymbol{T}_{i+1}$;
- $\boldsymbol{n}_{i+1}^{\star}$ the torque exerced by link i on link $i+1$, projected in frame $\boldsymbol{T}_i$;
- $\boldsymbol{f}_{i+1}'$ the force exerted by link i on link $i+1$, projected in frame $\boldsymbol{T}_{i+1}$;
- $\boldsymbol{f}_{i+1}^{\star}$ the force exerced by link i on link $i+1$, projected in frame $\boldsymbol{T}_i$;
- $\boldsymbol{g}_i^{\star}$ the location of the center of mass of link i measured from the origin of frame $\boldsymbol{T}_i$, and expressed in frame $\boldsymbol{T}_i^{\star}$
- $\boldsymbol{d}_i^{\star}$ the vector describing the length of member i, expressed in frame$\boldsymbol{T}_{i-1}$.

According to this notation, the total external force acting on link i is

$$\boldsymbol{F}_i' \;=\; -\,\boldsymbol{f}_{i+1}^{\star} \;+\; \boldsymbol{f}_i' \tag{6.3.1}$$

and similarly, the total external torque is

$$\boldsymbol{N}_i' \;=\; -\,\boldsymbol{n}_{i+1}^{\star} + \boldsymbol{n}_i' + \tilde{\boldsymbol{g}}_i'\,\boldsymbol{f}_{i+1}^{\star} - (\widetilde{\boldsymbol{d}_i' + \boldsymbol{g}_i'})\boldsymbol{f}_i' \tag{6.3.2}$$

According to (6.2.55) and (6.2.59), the equations of motion of link i, expressed in body coordinates, are

$$m_i\,\boldsymbol{a}_{ic}' \;=\; -\,m_i\boldsymbol{g} \;+\; \boldsymbol{F}_i' \tag{6.3.3}$$

$$\boldsymbol{J}_i'\,\dot{\omega}_i' \;+\; \tilde{\omega}_i'\,\boldsymbol{J}_i'\omega_i' \;=\; \boldsymbol{N}_i' \tag{6.3.4}$$

Where $\boldsymbol{g}$ is the vector of the gravity acceleration. From the kinematics, we know that positions, velocities and accelerations can be obtained from a *forward recursive procedure*. Forces and torques, on the other hand, have to be determined *backwards* since the reaction forces at the manipulator base are unknown. The recursive Newton-Euler procedure is then as follows.

6.3.1 Forward kinematic recursion

initialization

Gravity terms can be omitted from the procedure assuming that the ground has the acceleration of gravity, taken with negative sign

$$\begin{array}{lll} \omega_0' = 0 & \quad & \boldsymbol{v}_0' = 0 \\ \dot{\omega}_0' = 0 & \quad & \boldsymbol{a}_0' = -\,g \end{array} \tag{6.3.5}$$

Where g is the gravity acceleration assumed to be oriented in the negative direction of z_0 axis.

iteration i, $(i = 1, \ldots n)$

Velocities and accelerations are incremented and projected into the current body frame

$$\begin{aligned}
\omega_{i+1}^{\star} &= \omega_i' \;+\; (1-\sigma)\,\dot{\boldsymbol{q}}_{i+1}\,\boldsymbol{k} \\
\dot{\omega}_{i+1}^{\star} &= \dot{\omega}_i' \;+\; (1-\sigma)\,[\dot{\boldsymbol{q}}_{i+1}\tilde{\omega}_i' \;+\; \ddot{\boldsymbol{q}}_{i+1}]\,\boldsymbol{k} \\
\boldsymbol{a}_{i+1}^{\star} &= \boldsymbol{a}_i' \;+\; (\dot{\tilde{\omega}}_{i+1}^{\star} - \tilde{\omega}_{i+1}^{\star T}\,\tilde{\omega}_{i+1}^{\star})\boldsymbol{d}_{i+1}^{\star} \;+\; \sigma(2\dot{\boldsymbol{q}}_i'\tilde{\omega}_i' + \ddot{\boldsymbol{q}}_{i+1})\boldsymbol{k} \\
\omega_{i+1}' &= \boldsymbol{R}_{i+1}^{T}\omega_{i+1}^{\star} \\
\dot{\omega}_{i+1}' &= \boldsymbol{R}_{i+1}^{T}\dot{\omega}_{i+1}^{\star} \\
\boldsymbol{a}_{i+1}' &= \boldsymbol{R}_{i+1}^{T}\,\boldsymbol{a}_{i+1}^{\star}
\end{aligned} \tag{6.3.6}$$

The rotation operator $\boldsymbol{R}_{i+1}$ and the length vector being extracted from the relative transformation

$${}^{i}\boldsymbol{A}_{i+1} \;=\; \begin{bmatrix} \boldsymbol{R}_{i+1} & \boldsymbol{d}_{i+1}^{\star} \\ 0 & 1 \end{bmatrix} \tag{6.3.7}$$

for the center of mass, we simply have

$$\boldsymbol{a}_{i+1,c}' \;=\; \boldsymbol{a}_{i+1}' \;+\; \boldsymbol{R}_{i+1}^{T}\left(\dot{\tilde{\omega}}_{i+1}^{\star} - \dot{\tilde{\omega}}_{i+1}^{\star T}\dot{\tilde{\omega}}_{i+1}^{\star}\right)\boldsymbol{g}_{i+1}^{\star} \tag{6.3.8}$$

where

$$\boldsymbol{g}_{i+1}^{\star} \;=\; \boldsymbol{R}_{i+1}\,\boldsymbol{g}_{i+1}' \tag{6.3.9}$$

is the position of the center of mass of link $i+1$ expressed in frame $\boldsymbol{T}_i$.

6.3.2 Link inertia forces

The link inertia forces are evaluated from (6.3.3) and (6.3.4)

$$\begin{aligned} \boldsymbol{F}'_{i+1} &= m_{i+1}\boldsymbol{a}'_{i+1,c} \\ \boldsymbol{N}'_{i+1} &= \boldsymbol{J}'_{i+1}\dot{\omega}'_{i+1} + \tilde{\omega}'_{i+1}\boldsymbol{J}'_{i+1}\omega'_{i+1} \end{aligned} \tag{6.3.10}$$

6.3.3 Force backward recursion

The link interaction torques and forces are calculated from link n to link 1 according to the following scheme :

Initialization

The force $\boldsymbol{F}$ and torque $\boldsymbol{C}$ at the end of link n are those generated by the task

$$\begin{aligned} \boldsymbol{f}^{\star}_{n+1} &= \boldsymbol{R}^T\boldsymbol{F} \\ \boldsymbol{n}^{\star}_{n+1} &= \boldsymbol{R}^T\boldsymbol{C} \end{aligned} \tag{6.3.11}$$

Iteration

$$\begin{aligned} \boldsymbol{f}'_i &= \boldsymbol{F}'_i + \boldsymbol{f}^{\star}_{i+1} \\ \boldsymbol{n}'_i &= \boldsymbol{N}'_i + \boldsymbol{n}^{\star}_{i+1} + \tilde{\boldsymbol{g}}'_i\,\boldsymbol{F}'_i \ + \ \tilde{\boldsymbol{d}}'_i\,\boldsymbol{f}_{i+1} \\ \boldsymbol{f}^{\star}_i &= \boldsymbol{R}_i\,\boldsymbol{f}_{i+1} \\ \boldsymbol{n}^{\star}_i &= \boldsymbol{R}_i\,\boldsymbol{n}'_i \end{aligned} \tag{6.3.12}$$

Note that the driving torque is the z component of torque $\boldsymbol{n}^{\star}_i$. Therefore,

$$\tau_i = \boldsymbol{k}^T\,\boldsymbol{n}^{\star}_i \tag{6.3.13}$$

6.4 Lagrangian dynamics

Let $(q_1, q_2, \dots q_n)$ be the generalized coordinates which define completely the current configuration of the dynamic system.

Let $\mathcal{K}$ and $\mathcal{U}$ be the kinetic energy and potential energy stored in the system, obtained by summing the contributions of individual links

$$\mathcal{K} = \sum_{i=1}^{n}\mathcal{K}_i \quad , \qquad \mathcal{U} = \sum_{i=1}^{n}\mathcal{U}_i \tag{6.4.1}$$

we define the Lagrangian of the total system by

$$\mathcal{L}(\boldsymbol{q}_k, \dot{\boldsymbol{q}}_k) = \mathcal{K} - \mathcal{U} \tag{6.4.2}$$

Note that, since the kinetic and potential energies are functions of $\boldsymbol{q}_k$ and $\dot{\boldsymbol{q}}_k$ $(k = 1, \dots n)$, so is the Lagrangian $\mathcal{L}$. In terms of the Lagrangian, the equations of motion of the system are given by

$$\frac{d}{dt}\Big(\frac{\partial\mathcal{L}}{\partial\dot{\boldsymbol{q}}_k}\Big) - \frac{\partial\mathcal{L}}{\partial\boldsymbol{q}_k} = Q_k \qquad k = 1, \dots n \tag{6.4.3}$$

where Q_k are the generalized forces conjugated to the generalized coordinates $\boldsymbol{q}_k$. They can be obtained by expressing the virtual work of the non-conservative forces acting on the system and writing it in the form

$$\delta\mathcal{W} \;=\; \sum_{k=1}^{n} Q_k\,\delta\boldsymbol{q}_k \tag{6.4.4}$$

6.4.1 Structure of the kinetic energy

In the previous section, the kinetic energy of one single link has been put in the form

$$\mathcal{K}_i \;=\; \frac{1}{2}m_i\boldsymbol{v}_i^{\prime\,T}\boldsymbol{v}_i' \;+\; \frac{1}{2}\omega_i^{\prime\,T}\boldsymbol{J}_i'\omega_i' \tag{6.4.5}$$

when expressing linear and angular velocities in body axes.

According to kinematics, in the absence of overall motion $\boldsymbol{v}_i'$ and ω_i' can be expressed in the form of linear combinations of joint velocities

$$\boldsymbol{v}_i' \;=\; \sum_{j=1}^{i}\boldsymbol{a}_{ij}(\boldsymbol{q})\,\dot{q}_j$$

$$\omega_i' \;=\; \sum_{j=1}^{i}\boldsymbol{b}_{ij}(\boldsymbol{q})\,\dot{q}_j \tag{6.4.6}$$

where the coefficients $\boldsymbol{a}_{ij}$ and $\boldsymbol{b}_{ij}$ are vectors dependent on the current configuration.

The total kinetic energy $\mathcal{K}$ stored in the whole structure is then given by

$$\mathcal{K} \;=\; \sum_{i=1}^{n}\mathcal{K}_i \;=\; \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n}m_{jk}(\boldsymbol{q})\,\dot{q}_j\,\dot{q}_k \tag{6.4.7}$$

where, according to (6.4.5) and (6.4.6):

- The coefficients m_{jk} are configuration dependent and symmetric i.e.

$$m_{jk}(\boldsymbol{q}) \;=\; m_{kj}(\boldsymbol{q}) \tag{6.4.8}$$

- The resulting mass matrix

$$\boldsymbol{M} \;=\; [m_{jk}]$$

  is thus symmetric and also semi-positive definite since $\mathcal{K} > 0$ for $\dot{\boldsymbol{q}} \neq 0$.

6.4.2 Potential energy

In the absence of elastic deformation, the total potential energy of the system is made of the gravity contributions of the individual links

$$\mathcal{U} \;=\; \sum_{i=1}^{n}\mathcal{U}_i \;=\; \sum_{i=1}^{n}m_i\boldsymbol{g}^T\boldsymbol{r}_{ic} \tag{6.4.9}$$

Where $\boldsymbol{r}_{ic}$ is the position vector of the center of mass of link i and $\boldsymbol{g}$ is the gravity acceleration vector. The potential energy is thus a function of configuration variables $(q_1, \ldots q_n)$.

6.4.3 Virtual work of external forces

Let us consider the situation where actuators exert torques $\tau = [\tau_1, \dots \tau_n]^T$ at individual joints and that the arm's endpoint, while it is in contact with the environment, exerts an external force and moment $\boldsymbol{F}$.

The virtual work is then given by

$$\delta\mathcal{W} = \tau^T \delta\boldsymbol{q} - \boldsymbol{F}^T \delta\boldsymbol{x} \tag{6.4.10}$$

where $\boldsymbol{x}$ represents the tool configuration into task space. Making next use of the fact that the Jacobian matrix relates displacements increments into cartesian and joint spaces

$$\delta\boldsymbol{x} = \boldsymbol{J}\,\delta\boldsymbol{q} \tag{6.4.11}$$

equation (6.4.11) can be put in the form

$$\delta\mathcal{W} = \delta\boldsymbol{q}^T (\tau - \boldsymbol{J}^T \boldsymbol{F}) \tag{6.4.12}$$

6.4.4 Structure of inertia forces

Inertia forces can be expressed in the form

$$\frac{d}{dt}\Big(\frac{\partial\mathcal{K}}{\partial\dot{q}_k}\Big) - \frac{\partial\mathcal{K}}{\partial q_k} \qquad (k = 1, \dots n) \tag{6.4.13}$$

According to (6.4.7), the first term of (6.4.13) can be computed as

$$\frac{d}{dt}\Big(\sum_{j=1}^{n} m_{kj}\dot{q}_j\Big) = \sum_{j=1}^{n} [m_{kj}\ddot{q}_j + \frac{d}{dt}(m_{kj})\dot{q}_j] \tag{6.4.14}$$

with

$$\frac{d}{dt}(m_{kj}) = \sum_{l=1}^{n} \frac{\partial m_{kj}}{\partial q_l}\dot{q}_l$$

the second term of (6.4.13) is

$$\frac{\partial\mathcal{K}}{\partial q_k} = \frac{1}{2}\sum_{j=1}^{n}\sum_{l=1}^{n} \frac{\partial m_{jl}}{\partial q_k}\,\dot{q}_j\dot{q}_l \tag{6.4.15}$$

The result of (6.4.13) may thus be expressed in matrix form

$$\boldsymbol{M}\,\ddot{\boldsymbol{q}} + \boldsymbol{A}\,\dot{\boldsymbol{q}}^2 \tag{6.4.16}$$

or alternatively

$$\boldsymbol{M}\,\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\,\dot{\boldsymbol{q}} \tag{6.4.17}$$

where the second term accounts for centrigugal and Coriolis forces, and where the coefficient c_{kj} of matrix $\boldsymbol{C}$ are given by

$$c_{kj} = \sum_{l=1}^{n} \Big(\frac{\partial m_{kj}}{\partial q_l} - \frac{1}{2}\frac{\partial m_{lj}}{\partial q_k}\Big)\,\dot{q}_l \tag{6.4.18}$$

It is interesting to observe that the matrix

$$\boldsymbol{S} \;=\; \dot{\boldsymbol{M}} \;-\; 2\boldsymbol{C} \tag{6.4.19}$$

has general term s_{kj} which can be expressed in the form

$$\sum_{l=1}^{n}\Big(-\frac{\partial m_{kj}}{\partial q_l}+\frac{\partial m_{lj}}{\partial q_k}\Big)\dot{q}_l \tag{6.4.20}$$

so that it is anti-symmetric. This remark will be of importance for control purpose.

6.4.5 Gravity forces

Gravity forces produce generalized forces computed by

$$\boldsymbol{G}_k \;=\; \frac{\partial \mathcal{U}}{\partial q_k} \;=\; \sum_{i=1}^{n} m_i \boldsymbol{g}^T \frac{\partial \boldsymbol{r}_{ci}}{\partial q_k} \tag{6.4.21}$$

6.4.6 Equations of motion

Collecting the different contributions to (6.4.3), we obtain the dynamic equilibrium equations in the general form

$$\boldsymbol{M}\,\ddot{\boldsymbol{q}} \;+\; \boldsymbol{A}\,\dot{\boldsymbol{q}}^2 \;+\; \boldsymbol{G} \;+\; \boldsymbol{J}^T\boldsymbol{F} \;=\; \tau \tag{6.4.22}$$

6.5 Recursive Lagrangian formulation

Just as Newton-Euler formulation, the Lagrangian formulation for the dynamics of an open-tree, simply-connected structure can be put in recursive form.

To simplify the presentation, we will use homogeneous transformations to describe the kinematics of the system.

6.5.1 Forward kinematics

Let us express that the position of an arbitrary point P on member i is obtained in homogeneous form by

$$\boldsymbol{p} \;=\; \boldsymbol{T}_i\,\boldsymbol{p}' \tag{6.5.1}$$

where $\boldsymbol{p}'$ is the position of same point P in the local frame $\boldsymbol{T}_i$. Transformation matrix $\boldsymbol{T}_i$ is obtained from a forward recursion

$$\boldsymbol{T}_i \;=\; {}^0\boldsymbol{A}_1\,{}^1\boldsymbol{A}_2 \ldots {}^{i-1}\boldsymbol{A}_i \tag{6.5.2}$$

Similarly, velocities and accelerations are obtained by

$$\dot{\boldsymbol{p}} \;=\; \dot{\boldsymbol{T}}_i\,\boldsymbol{p}' \tag{6.5.3}$$

and

$$\ddot{\boldsymbol{p}} \;=\; \ddot{\boldsymbol{T}}_i\,\boldsymbol{p}' \tag{6.5.4}$$

where $\dot{\boldsymbol{T}}_i$ and $\ddot{\boldsymbol{T}}_i$ may also be computed recursively. Starting from

$$\boldsymbol{T}_i \;=\; \boldsymbol{T}_{i-1}\boldsymbol{A}_i \qquad \text{with} \qquad \boldsymbol{A}_i \;=\; {}^{i-1}\boldsymbol{A}_i(q_i) \tag{6.5.5}$$

the first derivative is computed as follows

$$\dot{\boldsymbol{T}}_i \;=\; \dot{\boldsymbol{T}}_{i-1}\,\boldsymbol{A}_i \;+\; \boldsymbol{T}_{i-1}\dot{\boldsymbol{A}}_i$$

or

$$\boxed{\dot{\boldsymbol{T}}_i \;=\; \left[\dot{\boldsymbol{T}}_{i-1} + \boldsymbol{T}_{i-1}\;{}^{i-1}\dot{\boldsymbol{\Delta}}_i\right]\boldsymbol{A}_i} \tag{6.5.6}$$

with

$${}^{i-1}\dot{\boldsymbol{\Delta}}_i \;=\; \dot{q}_i \begin{bmatrix} (1-\sigma)\tilde{\boldsymbol{k}} & \sigma\boldsymbol{k} \\ \mathbf{o}^T & 0 \end{bmatrix} \tag{6.5.7}$$

Similarly, for the second derivative

$$\ddot{\boldsymbol{T}}_i \;=\; \ddot{\boldsymbol{T}}_{i-1}\boldsymbol{A}_i \;+\; 2\dot{\boldsymbol{T}}_{i-1}\dot{\boldsymbol{A}}_i \;+\; \boldsymbol{T}_{i-1}\ddot{\boldsymbol{A}}_{i-1}$$

or

$$\boxed{\ddot{\boldsymbol{T}}_i \;=\; \left[\ddot{\boldsymbol{T}}_{i-1} \;+\; 2\dot{\boldsymbol{T}}_{i-1}\;{}^{i-1}\dot{\boldsymbol{\Delta}}_i \;+\; \boldsymbol{T}_{i-1}\;{}^{i-1}\ddot{\boldsymbol{\Delta}}_i\right]\boldsymbol{A}_i} \tag{6.5.8}$$

with

$${}^{i-1}\ddot{\boldsymbol{\Delta}}_i \;=\; \ddot{q}_{i+1} \begin{bmatrix} (1-\sigma)\tilde{\boldsymbol{k}} & \sigma\boldsymbol{k} \\ \mathbf{o}^T & 0 \end{bmatrix} \;-\; \dot{q}_{i+1}^2 \begin{bmatrix} (1-\sigma)\tilde{\boldsymbol{k}}\,\tilde{\boldsymbol{k}}^T & 0 \\ \mathbf{o}^T & 0 \end{bmatrix} \tag{6.5.9}$$

6.5.2 Kinetic energy

The kinetic energy is the sum of individual link contributions

$$\mathcal{K} \;=\; \sum_{i=1}^{n} \mathcal{K}_i \tag{6.5.10}$$

and the kinetic energy of one member is calculated by

$$\mathcal{K}_i \;=\; \frac{1}{2}\int_{V_i} \dot{\boldsymbol{p}}^T\,\dot{\boldsymbol{p}}\,dm \tag{6.5.11}$$

Note that it can be expressed in the equivalent form

$$\mathcal{K}_i \;=\; \frac{1}{2}\int_{V_i} trace\{\dot{\boldsymbol{p}}\,\dot{\boldsymbol{p}}^T\}\,dm \tag{6.5.12}$$

in which case, in local coordinates

$$\mathcal{K}_i \;=\; \frac{1}{2} trace\Big\{\dot{\boldsymbol{T}}_i \int_{V_i} \boldsymbol{p}'\,\boldsymbol{p}'^T\,dm\;\dot{\boldsymbol{T}}_i\Big\} \tag{6.5.13}$$

6.5.3 Matrix of inertia

The matrix

$$\boldsymbol{J}_i \;=\; \int_{V_i} \boldsymbol{p}'\boldsymbol{p}'^T dm \tag{6.5.14}$$

represents again, but this time in homogeneous coordinates, the inertia properties of link i (rotational and translational properties). It is easy to control that with the definitions of moments of inertia

$$I_{xx} \;=\; \int_{V_i}(y'^2+z'^2)dm \quad I_{yy} \;=\; \int_{V_i}(x'^2+z'^2)dm \quad I_{zz} \;=\; \int_{V_i}(x'^2+y'^2)dm$$

$$I_{xy} \;=\; \int_{V_i} x'\,y'dm \qquad I_{yz} \;=\; \int_{V_i} y'\,z'dm \qquad I_{zx} \;=\; \int_{V_i} z'\,x'dm$$

and mass properties

$$m = \int_{V_i} dm$$

$$m\,x'_c = \int_{V_i} x' dm \quad m\,y'_c = \int_{V_i} y' dm \quad m\,z'_c = \int_{V_i} z' dm$$

inertia matrix $\boldsymbol{J}_i$ can be has the explicit form

$$\boldsymbol{J} = \begin{bmatrix} (-I_{xx}+I_{yy}+I_{zz})/2 & I_{xy} & I_{xz} & mx'_c \\ I_{xy} & (I_{xx}-I_{yy}+I_{zz})/2 & I_{yz} & my'_c \\ I_{xz} & I_{yz} & (I_{xx}+I_{yy}-I_{zz})/2 & mz'_c \\ mx'_c & my'_c & mz'_c & m \end{bmatrix} \tag{6.5.15}$$

Note that in the 4×4 representation, the fourth row and column of the matrix represents the translation inertia properties.

6.5.4 Potential energy

We consider only the potential energy due to gravity

$$\mathcal{U}_i = -\int_{V_i} \boldsymbol{g}^T \boldsymbol{p}\; dm \tag{6.5.16}$$

or in terms of the position vector of the center of mass

$$\begin{aligned} \mathcal{U}_i &= -\,m_i\, \boldsymbol{g}^T\, \boldsymbol{p}_{ic} \\ &= -\,m_i\, \boldsymbol{g}^T\, \boldsymbol{T}_i\, \boldsymbol{p}'_{ic} \end{aligned} \tag{6.5.17}$$

6.5.5 Partial derivatives of kinetic energy

In order to express dynamic equilibrium, we have to calculate the inertia terms in the form

$$\frac{d}{dt}\Big(\frac{\partial \mathcal{K}}{\partial \dot{q}_k}\Big) - \frac{\partial \mathcal{K}}{\partial q_k} \qquad (k = 1, \ldots\; n) \tag{6.5.18}$$

let us start by

$$\begin{aligned} \frac{\partial \mathcal{K}}{\partial \dot{q}_k} &= \frac{\partial}{\partial \dot{q}_k}\Big[\frac{1}{2} trace\{\sum_{i=1}^{n} \dot{\boldsymbol{T}}_i\, \boldsymbol{J}_i \dot{\boldsymbol{T}}_i^T\}\Big] \\ &= \sum_{i=1}^{n} trace\{\frac{\partial \dot{\boldsymbol{T}}_i}{\partial \dot{q}_k} \boldsymbol{J}_i \dot{\boldsymbol{T}}_i\} \end{aligned} \tag{6.5.19}$$

where we note that

$$\begin{cases} \dfrac{\partial \dot{\boldsymbol{T}}_i}{\partial \dot{q}_k} = 0 & \text{if} \quad k > i \\[2ex] \dfrac{\partial \dot{\boldsymbol{T}}_i}{\partial \dot{q}_k} = \dfrac{\partial \boldsymbol{T}_i}{\partial q_k} & \text{if} \quad k \le i \end{cases} \tag{6.5.20}$$

giving the result

$$\frac{\partial \mathcal{K}}{\partial \dot{q}_k} = \sum_{i=k}^{n} trace\{\frac{\partial \boldsymbol{T}_i}{\partial q_k}\, \boldsymbol{J}_i\, \dot{\boldsymbol{T}}_i^T\} \tag{6.5.21}$$

and thus

$$\frac{d}{dt}\left(\frac{\partial \mathcal{K}}{\partial \dot{q}_k}\right) = \sum_{i=k}^{n} trace\{\frac{\partial \dot{\boldsymbol{T}}_i}{\partial q_k}\, \boldsymbol{J}_i\, \dot{\boldsymbol{T}}_i^T + \frac{\partial \boldsymbol{T}_i}{\partial q_k}\, \boldsymbol{J}_i\, \ddot{\boldsymbol{T}}_i^T\} \tag{6.5.22}$$

Similarly, we have

$$\frac{\partial \mathcal{K}}{\partial q_k} = \sum_{i=k}^{n} trace\{\frac{\partial \dot{\boldsymbol{T}}_i}{\partial q_k}\, \boldsymbol{J}_i\, \dot{\boldsymbol{T}}_i^T\} \tag{6.5.23}$$

and therefore we obtain the very compact result

$$\frac{d}{dt}\left(\frac{\partial \mathcal{K}}{\partial \dot{q}_k}\right) - \frac{\partial \mathcal{K}}{\partial q_k} = \sum_{i=k}^{n} trace\left\{\frac{\partial \boldsymbol{T}_i}{\partial q_k}\, \boldsymbol{J}_i\, \ddot{\boldsymbol{T}}_i^T\right\} \tag{6.5.24}$$

which according to (6.5.8), can be computed recursively.

6.5.6 Partial derivatives of potential energy

The partial derivatives of potential energy are similarly expressed by

$$\frac{\partial \mathcal{U}}{\partial q_k} = -\sum_{i=k}^{n} m_i\, \boldsymbol{g}^T \frac{\partial \boldsymbol{T}_i}{\partial q_k} \boldsymbol{p}'_{ic} \tag{6.5.25}$$

6.5.7 Equations of motion

The equations of motion of the system are thus given by

$$\boxed{\sum_{i=k}^{n} \left[trace\{\frac{\partial \boldsymbol{T}_i}{\partial q_k}\, \boldsymbol{J}_i\, \ddot{\boldsymbol{T}}_i^T\} - m_i \boldsymbol{g}^T \frac{\partial \boldsymbol{T}_i}{\partial q_k} \boldsymbol{p}'_{ic}\right] = Q_k(t) \qquad (k = 1, \dots\ n)} \tag{6.5.26}$$

In this form, making use of *forward recursive procedure to compute* $\ddot{\boldsymbol{T}}_i$, the number of arithmetic operation has n^2 dependence. For $n = 6$, 705 multiplies and 5652 adds are required.

6.5.8 Recursive form of equations of motion

Let us note that

$$\frac{\partial \boldsymbol{T}_i}{\partial q_k} = \frac{\partial \boldsymbol{T}_k}{\partial q_k}\, {}^k\boldsymbol{T}_i \tag{6.5.27}$$

in which case, the generalized forces (6.5.26) can be expanded as

$$\sum_{i=k}^{n} \left[trace\{\frac{\partial \boldsymbol{T}_k}{\partial q_k}\, {}^k\boldsymbol{T}_i\, \boldsymbol{J}_i\, \ddot{\boldsymbol{T}}_i^T\} - m_i\, \boldsymbol{g}^T\, \frac{\partial \boldsymbol{T}_k}{\partial q_k}\, {}^k\boldsymbol{T}_i\, \boldsymbol{p}'_{ic}\right] = \boldsymbol{Q}_k(t) \tag{6.5.28}$$

To find a recursive computation procedure, one has to recast generalized forces in joints as:

$$trace\{\frac{\partial \boldsymbol{T}_k}{\partial q_k}\, [\sum_{i=k}^{n} {}^k\boldsymbol{T}_i\, \boldsymbol{J}_i\, \ddot{\boldsymbol{T}}_i^T]\} - \boldsymbol{g}^T\, \frac{\partial \boldsymbol{T}_k}{\partial q_k}\, [\sum_{i=k}^{n} m_i\, {}^k\boldsymbol{T}_i\, \boldsymbol{p}'_{ic}] = \boldsymbol{Q}_k(t) \tag{6.5.29}$$

Let us define

$$\begin{aligned} \boldsymbol{D}_k(t) &= \sum_{i=k}^{n} {}^k\boldsymbol{T}_i\, \boldsymbol{J}_i\, \ddot{\boldsymbol{T}}_i^T \\ &= \boldsymbol{J}_k\, \ddot{\boldsymbol{T}}_k^T \;+\; \sum_{i=k+1}^{n} \boldsymbol{A}_{k+1}\, {}^{k+1}\boldsymbol{T}_i\, \boldsymbol{J}_i\, \ddot{\boldsymbol{T}}_i^T \end{aligned}$$

or

$$\boxed{\boldsymbol{D}_k \;=\; \boldsymbol{J}_k\, \ddot{\boldsymbol{T}}_k^T \;+\; \boldsymbol{A}_{k+1}\, \boldsymbol{D}_{k+1}} \tag{6.5.30}$$

Similarly,

$$\boldsymbol{C}_k(t) \;=\; \sum_{i=k}^{n} m_i\, {}^k\boldsymbol{T}_i\, \boldsymbol{p}'_{ic}$$

or

$$\boxed{\boldsymbol{C}_k \;=\; m_k \boldsymbol{p}'_{kc} \;+\; \boldsymbol{A}_{k+1}\, \boldsymbol{C}_{k+1}} \tag{6.5.31}$$

Computations of $\boldsymbol{D}_k$ and $\boldsymbol{C}_k$ lend themselves to a *recursive backward computation procedure*. These two quantities in hands, the generalized forces $\boldsymbol{Q}_k$ developed in joints (6.5.29) take then the very simple form:

$$\boxed{\boldsymbol{Q}_k(t) \;=\; trace\{\frac{\partial \boldsymbol{T}_k}{\partial q_k}\, \boldsymbol{D}_k\} \;-\; \boldsymbol{g}^T\, \frac{\partial \boldsymbol{T}_k}{\partial q_k}\, \boldsymbol{C}_k} \tag{6.5.32}$$

In this backward recursive form, the number of operations has order n dependence. For $n = 6$, there are only 4388 multiplies and 3586 adds.

There would still be a 50% improvement by omitting the 4×4 representation.

Chapter 7

TRAJECTORY GENERATION

Appendix A

ELEMENTS OF VECTOR AND MATRIX ALGEBRA

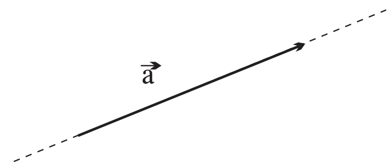


Figure A.2.1: Geometric representation of a vector

A.1 Introduction

Kinematics and dynamic analysis of mechanical systems such as articulated mechanisms and robots implies using the appropriate tools for representing the geometry of the motion.

Vector algebra is the appropriate tool to this end. Two different points of view may however be considered, according to the goal pursued. In the *geometric approach*, the basic operations of vector calculus are described in an *intrinsic manner*, i.e. vectors are "geometric beings" independent of the reference frame in which they are expressed.

In the *matrix approach*, the basic operations of vector calculus are defined in an *extrinsic manner*, using algebraic quantities that describe the vectors in the reference frame under consideration.

Most textbooks of dynamics use the geometric approach, which is the classical way of describing the kinematics of 3-dimensional motion.

Here, a different point of view will be adopted. It is found that the matrix approach leads to a formalism which is much more appropriate for subsequent computer implementation.

We will thus briefly recall the geometric concept of vector and the basic operations between vectors. We will next translate them in terms of matrix algebra.

A.2 Definitions and basic operations of vector calculus

Scalars

In various physical applications there appear certain quantities, such as temperature, the specific mass of a material or the pressure in a fluid, which possess only *magnitude*. These can be represented by real numbers and are called scalars.

Vectors

On the other hand, a vector is a geometric quantity which possesses both *magnitude* and *direction*. Its geometric notation is the arrow symbol. Three types of vectors can be formally distinguished:

- A *free vector* is a vector for which magnitude and orientation are specified, but not its line of action.
- A *bound vector* is a vector issued from a specified point in space.
- A *sliding vector* is a vector acting along a specified line in space (called its line of action), but at an arbitrary point of application.

For example:

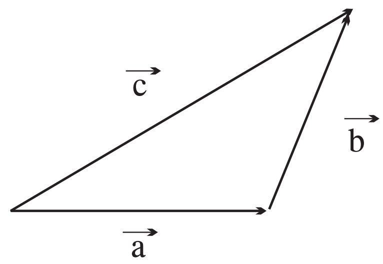


Figure A.2.2: Geometric representation of the sum of two vectors

- The position vector $\vec{p}$ of a given point P in a reference frame centered at O is a bound vector;
- The force vector $\vec{f}$ representing the external force acting on a rigid body acts along a specified line of action and is thus a sliding vector;
- The displacement vector $\vec{d}$ representing the displacement of a material point from A to B is a free vector.

The *magnitude* or *norm* of a vector $\vec{a}$ is its length, which we denote by $|\vec{a}|$, or a.

A *unit vector*, i.e. a vector with a norm equal to 1, specifies a direction. For example, we will denote $\vec{u}_a$ the unit vector having same direction as $\vec{a}$.

We begin by considering the two following operations on vectors.

Scalar multiplication

The product of a vector $\vec{a}$ by a real scalar α is obtained by multiplying the magnitude of $\vec{a}$ by α and retaining the same direction if $\alpha \geq 0$ or the opposite direction if $\alpha < 0$.

The scalar multiplication is distributive:

$$(\alpha + \beta)\,\vec{a} = \alpha\vec{a} + \beta\vec{a} \tag{A.2.1}$$

Addition of two vectors

The addition of two vectors $\vec{a}$ and $\vec{b}$ is governed by the so-called *parallelogram rule*, i.e. the resulting vector

$$\vec{c} = \vec{a} + \vec{b} \tag{A.2.2}$$

is the diagonal of the parallelogram formed by $\vec{a}$ and $\vec{b}$ as shown on figureA.2.

The addition is *commutative* since (see figure A.2.3)

$$\vec{a} + \vec{b} = \vec{b} + \vec{a} \tag{A.2.3}$$

It is also associative (see figure A.2.4) since

$$\vec{a} + \vec{b} + \vec{c} = (\vec{a} + \vec{b}) + \vec{c} = \vec{a} + (\vec{b} + \vec{c}) \tag{A.2.4}$$

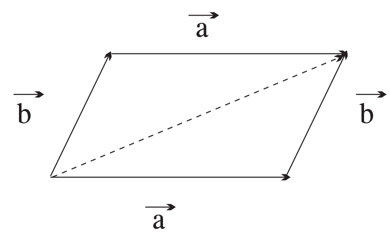


Figure A.2.3: Commutativity of the sum of two vectors

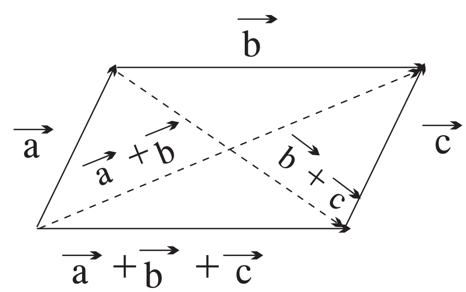


Figure A.2.4: Associativity of the vector sum

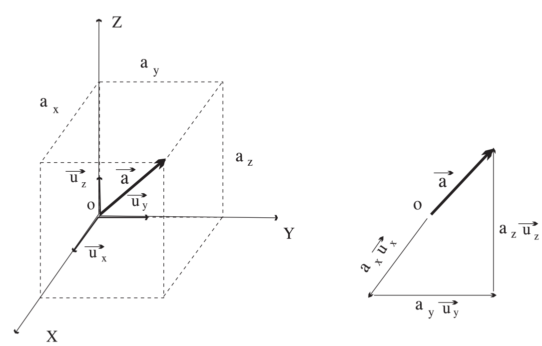


Figure A.2.5: Decomposition of a vector into cartesian components

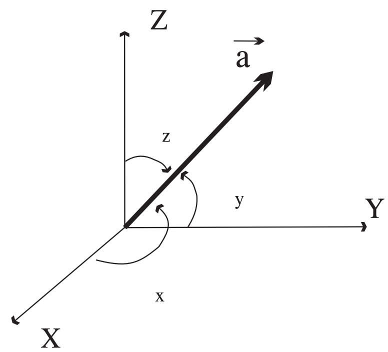


Figure A.2.6: Direction cosines of a vector

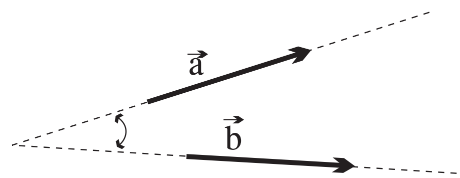


Figure A.2.7: Dot product of two vectors

Cartesian components of a vector

Let us consider (figure A.2.5) a cartesian frame $OXYZ$ with reference axes defined by the unit vectors $\vec{u}_x$, $\vec{u}_y$ and $\vec{u}_z$.

An arbitrary vector $\vec{a}$ may be decomposed in a sum of three vectors with respective lengths a_x, a_y and a_z along the three reference axes. Using vector notation

$$\vec{a} \;=\; a_x\,\vec{u}_x \;+\; a_y\,\vec{u}_y \;+\; a_z\,\vec{u}_z \tag{A.2.5}$$

Let us denote next by θ_x, θ_y and θ_z the angles between the vector $\vec{a}$ and the coordinate axes. The components of $\vec{a}$ are then given by

$$\begin{aligned} a_x &= a\;\cos\theta_x \;=\; a\;\ell_x \\ a_y &= a\;\cos\theta_y \;=\; a\;\ell_y \\ a_z &= a\;\cos\theta_z \;=\; a\;\ell_z \end{aligned} \tag{A.2.6}$$

where ℓ_x, ℓ_y and ℓ_z stand for the direction cosines of $\vec{a}$.

Dot product of two vectors

The *dot or scalar product* of two vectors $\vec{a}$ and $\vec{b}$ is defined as the scalar resulting from the product of the norm of these vectors and the cosine of their relative angle (see figure A.2.7)

$$\vec{a} \bullet \vec{b} \;=\; ab\;\cos\theta \tag{A.2.7}$$

the θ angle being measured in the intersecting plane of both vectors.

Immediate consequences of this definition are that

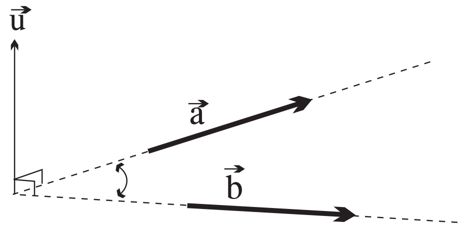


Figure A.2.8: Cross product of two vectors

- The dot product of two *orthogonal* vectors is zero; In particular, the unit vectors along the reference axes form a normal basis such that

$$\begin{aligned} \vec{u}_x \bullet \vec{u}_y &= \vec{u}_y \bullet \vec{u}_z = \vec{u}_z \bullet \vec{u}_x = 0 \\ \vec{u}_x \bullet \vec{u}_x &= \vec{u}_y \bullet \vec{u}_y = \vec{u}_z \bullet \vec{u}_z = 1 \end{aligned} \tag{A.2.8}$$

- The dot product of a vector $\vec{a}$ by itself is equal to the square of its norm

$$\vec{a} \bullet \vec{a} = a^2 \tag{A.2.9}$$

- The dot product is *commutative*

$$\vec{a} \bullet \vec{b} = \vec{b} \bullet \vec{a} \tag{A.2.10}$$

- In terms of its cartesian components, the dot product $\vec{a} \bullet \vec{b}$ reads

$$\begin{aligned} \vec{a} \bullet \vec{b} &= a_x\vec{u}_x + a_y\vec{u}_y + a_z\vec{u}_z) \bullet (b_x\vec{u}_x + b_y\vec{u}_y + b_z\vec{u}_z) \\ &= a_x b_x + a_y b_y + a_z b_z \end{aligned} \tag{A.2.11}$$

since the unit vectors $\vec{u}_x$, $\vec{u}_y$ and $\vec{u}_z$ form an orthonormal basis;

- The dot product is *distributive*

$$\vec{a} \bullet (\vec{b} + \vec{c}) = \vec{a} \bullet \vec{b} + \vec{a} \bullet \vec{c} \tag{A.2.12}$$

- The angle between the two vectors may be found from their dot product by

$$\begin{aligned} \cos\theta = \frac{\vec{a} \bullet \vec{b}}{ab} &= \frac{a_x b_x + a_y b_y + a_z b_z}{ab} \\ &= \ell_x m_x + \ell_y m_y + \ell_z m_z \end{aligned} \tag{A.2.13}$$

where the m_i stand for the direction cosines of vector $\vec{b}$.

Cross or vector product of two vectors

The *cross product* of two vectors $\vec{a}$ and $\vec{b}$ (see figure A.2.8) is defined as a vector whose magnitude equals the product of the magnitudes of $\vec{a}$ and $\vec{b}$ multiplied by the sine of their relative angle. Its direction, denoted by the unit vector $\vec{u}$, is normal to the plane defined by the right-hand screw rule. The cross product may thus be written

$$\vec{a} \times \vec{b} = ab \sin\theta \, \vec{u} \tag{A.2.14}$$

Let us note that according to this definition

- Reversing the order of the cross product would reverse the sign of the vector $\vec{u}$. Hence, the cross product is *not commutative*:

$$\vec{a} \times \vec{b} \;=\; -\; \vec{b} \times \vec{a} \tag{A.2.15}$$

- The cross product of two parallel vectors is zero. In particular,

$$\vec{a} \times \vec{a} \;=\; 0 \tag{A.2.16}$$

- The orthogonal vectors defining the cartesian frame are such that

$$\begin{aligned} \vec{u}_x \times \vec{u}_y &= -\,\vec{u}_y \times \vec{u}_x \;=\; \vec{u}_z \\ \vec{u}_y \times \vec{u}_z &= -\,\vec{u}_z \times \vec{u}_y \;=\; \vec{u}_x \\ \vec{u}_z \times \vec{u}_x &= -\,\vec{u}_x \times \vec{u}_z \;=\; \vec{u}_y \end{aligned} \tag{A.2.17}$$

- In terms of cartesian coordinates, the dot product has for expression

$$\begin{aligned} \vec{a} \times \vec{b} &= (a_x\vec{u}_x \;+\; a_y\vec{u}_y \;+\; a_z\vec{u}_z) \times (b_x\vec{u}_x \;+\; b_y\vec{u}_y \;+\; b_z\vec{u}_z) \\ &= (a_y b_z \;-\; a_z b_y)\vec{u}_x \;+\; (a_z b_x \;-\; a_x b_z)\vec{u}_y \\ &\quad +\; (a_x b_y \;-\; a_y b_x)\vec{u}_z \end{aligned} \tag{A.2.18}$$

- From the relation above, it is easy to control that the dot product is distributive

$$(\vec{a} + \vec{b}) \times \vec{c} \;=\; \vec{a} \times \vec{c} \;+\; \vec{b} \times \vec{c} \tag{A.2.19}$$

Additional relationships

Two additional relations of vector algebra are given without proof, but their validity is easy to demonstrate by decomposing them into cartesian components.

The *triple scalar product* is the dot product of two vectors where one of them is specified as a cross product of two additional vectors. The result is a scalar and is given by one of the equivalent expressions obtained by cyclic permutation

$$(\vec{a} \times \vec{b}) \bullet \vec{c} \;=\; (\vec{b} \times \vec{c}) \bullet \vec{a} \;=\; (\vec{c} \times \vec{a}) \bullet \vec{b} \tag{A.2.20}$$

It may also be seen upon expansion that

$$(\vec{a} \times \vec{b}) \bullet \vec{c} \;=\; dtm \begin{pmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \end{pmatrix} \tag{A.2.21}$$

which means that its absolute value is the volume of the parallelipiped with sides $\vec{a}$, $\vec{b}$ and $\vec{c}$.

The *triple vector product* is the cross product of two vectors where one of them is specified as a cross product of two additional vectors. The result is a vector and is given by one of the equivalent expressions

$$(\vec{a} \times \vec{b}) \times \vec{c} \;=\; -\; \vec{c} \times (\vec{a} \times \vec{b}) \;=\; \vec{c} \times (\vec{b} \times \vec{a}) \tag{A.2.22}$$

It is easily shown that it is equivalent to

$$(\vec{a} \times \vec{b}) \times \vec{c} \;=\; (\vec{a} \bullet \vec{c}).\vec{b} \;-\; (\vec{b} \bullet \vec{c}).\vec{a} \tag{A.2.23}$$

and similarly

$$\vec{a} \times (\vec{b} \times \vec{c}) \;=\; (\vec{a} \bullet \vec{c}).\vec{b} \;-\; (\vec{a} \bullet \vec{b}).\vec{c} \tag{A.2.24}$$

A.3 Definitions and basic operations of matrix algebra

By definition, a *rectangular matrix* $\boldsymbol{A}$ of dimension $m \times n$ is the rectangular array made of m *rows* and n *columns*

$$\boldsymbol{A} = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \quad (A.3.1)$$

where its general element a_{ij}, a scalar, is located at the intersection of row i and column j. To note it, we use boldface capital letters.

Transpose of a matrix

The transpose of matrix $\boldsymbol{a}$, denoted by $\boldsymbol{A}^T$, is the rectangular array of dimension $n \times m$ obtained by interverting the rows and columns of the original matrix.

Column- and row-matrices

A matrix made of only one column is a *column matrix*; it will be denoted by using boldface small letters.

$$\boldsymbol{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \quad (A.3.2)$$

Similarly, a matrix made of only one row is a *row matrix*. It may be written as the transpose of a column matrix

$$\boldsymbol{a}^T = \begin{bmatrix} a_1 & a_2 & \dots a_n \end{bmatrix} \quad (A.3.3)$$

Null matrix

A null matrix is a rectangular matrix made of zeros

$$\boldsymbol{A} = 0 \qquad \text{is such that} \qquad a_{ij} = 0 \qquad i = 1, \dots m, \quad j = 1, \dots m \quad (A.3.4)$$

Square matrix

A *square matrix* is a matrix having equal number of rows and columns. Furthermore:

- It is *symmetric*, if two terms located symmetrically with respect to the diagonal are equal

$$a_{ij} = a_{ji} \qquad \text{for all} \quad i, j = 1, \dots n \quad (A.3.5)$$

- It is *skew-symmetric*, or *antisymmetric*, if the sum of two terms located symmetrically with respect to the diagonal is zero

$$a_{ij} + a_{ji} = 0 \qquad \text{for all} \quad i, j = 1, \dots n \quad (A.3.6)$$

A consequence of this definition is that a skew-symmetric matrix has zero diagonal terms

$$a_{ii} = 0 \qquad \text{for all} \quad i = 1, \dots n \quad (A.3.7)$$

- A square matrix is *diagonal* if only the diagonal terms do not vanish

$$a_{ij} = 0 \qquad \text{for all} \quad i \neq j \tag{A.3.8}$$

in which case it may also be represented by

$$\boldsymbol{A} = diag \begin{bmatrix} a_{11} & a_{22} & \dots & a_{nn} \end{bmatrix} \tag{A.3.9}$$

- It is *quasi-diagonal* if it is made of square matrices $\boldsymbol{B}_1$, $\boldsymbol{B}_2$, $\dots$ $\boldsymbol{B}_n$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{B}_1 & & & \mathbf{o} \\ & \boldsymbol{B}_2 & & \\ & & \ddots & \\ \mathbf{o} & & & \boldsymbol{B}_n \end{bmatrix} \tag{A.3.10}$$

- The *unit* or *identity matrix* of dimension $n \times n$ is the square diagonal matrix made of diagonal terms equal to 1

$$\boldsymbol{I} = diag \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix} \tag{A.3.11}$$

Equality of two matrices

Two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are said to be equal if they have the same dimension and if the corresponding elements are equal:

$$a_{ij} = b_{ij} \qquad (i = 1, \dots m, \quad j = 1, \dots n) \tag{A.3.12}$$

Addition and difference of two matrices

The *addition* or *sum* of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ having the same dimension is defined by the operation

$$\boldsymbol{C} = \boldsymbol{A} + \boldsymbol{B} \tag{A.3.13}$$

where the general term of $\boldsymbol{C}$ is given by

$$c_{ij} = a_{ij} + b_{ij} \qquad (i = 1, \dots m, \quad j = 1, \dots n) \tag{A.3.14}$$

Similarly, the *difference* of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ of same dimension is defined by the operation

$$\boldsymbol{C} = \boldsymbol{A} - \boldsymbol{B} \tag{A.3.15}$$

with

$$c_{ij} = a_{ij} - b_{ij} \qquad (i = 1, \dots m, \quad j = 1, \dots n) \tag{A.3.16}$$

According to the definition, matrix addition is *commutative*

$$\boldsymbol{A} + \boldsymbol{B} = \boldsymbol{B} + \boldsymbol{A} \tag{A.3.17}$$

It is also *associative*

$$(\boldsymbol{A} + \boldsymbol{B}) + \boldsymbol{C} = \boldsymbol{A} + (\boldsymbol{B} + \boldsymbol{C}) = \boldsymbol{A} + \boldsymbol{B} + \boldsymbol{C} \tag{A.3.18}$$

The *transpose of the sum* of two matrices is equal to the sum of the transposed matrices

$$(\boldsymbol{A} + \boldsymbol{B})^T = \boldsymbol{A}^T + \boldsymbol{B}^T \tag{A.3.19}$$

Finally, any square matrix may be expressed as the sum of a symmetric and a skew-symmetric matrices

$$\boldsymbol{A} = \boldsymbol{B} + \boldsymbol{C} \tag{A.3.20}$$

with

$$\boldsymbol{B} = \frac{1}{2}(\boldsymbol{A} + \boldsymbol{A}^T) \qquad \text{and} \qquad \boldsymbol{C} = \frac{1}{2}(\boldsymbol{A} - \boldsymbol{A}^T) \tag{A.3.21}$$

Scalar multiplication of a matrix

The multiplication of a matrix $\boldsymbol{A}$ by a scalar α (a scalar being always, with our notation, denoted by a small Greek letter)

$$\alpha \boldsymbol{A} \;=\; \boldsymbol{C} \tag{A.3.22}$$

where

$$c_{ij} \;=\; \alpha a_{ij} \qquad (i = 1, \ldots m, \quad j = 1, \ldots n) \tag{A.3.23}$$

Matrix multiplication

Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be two matrices of dimensions $m \times p$ and $p \times n$ respectively, which we write in the form

$$\boldsymbol{A} \;=\; \begin{bmatrix} \boldsymbol{a}_1^T \\ \boldsymbol{a}_2^T \\ \vdots \\ \boldsymbol{a}_n^T \end{bmatrix} \quad \text{and} \quad \boldsymbol{B} \;=\; \begin{bmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \ldots & \boldsymbol{b}_n \end{bmatrix} \tag{A.3.24}$$

where the $\boldsymbol{a}_i^T \quad (i = 1, \ldots m)$ and $\boldsymbol{b}_j \quad (j = 1, \ldots n)$ are row- and column-matrices of dimension p.

One defines first the dot product $\boldsymbol{a}^T\boldsymbol{b}$ of a row matrix with a column matrix

$$\begin{aligned} \boldsymbol{a}^T\boldsymbol{b} \;&=\; a_1 b_1 \;+\; a_2 b_2 \;+\; \ldots a_n b_n \\ &=\; \sum_{i=1}^{p} a_i b_i \end{aligned} \tag{A.3.25}$$

The matrix product of $\boldsymbol{A}$ and $\boldsymbol{B}$ is defined next as the matrix $\boldsymbol{C}$ of dimension $m \times n$

$$\boldsymbol{C} \;=\; \boldsymbol{A}\,\boldsymbol{B} \tag{A.3.26}$$

such that

$$c_{ij} \;=\; \boldsymbol{a}_i^T \boldsymbol{b}_j \tag{A.3.27}$$

or, in terms of the elements of $\boldsymbol{A}$ and $\boldsymbol{B}$

$$c_{ij} \;=\; \sum_{k=1}^{p} a_{ik} b_{kj} \tag{A.3.28}$$

It is fundamental noticing that a matrix multiplication can be performed only if the number of columns of the first matrix equals the number of rows of the second one.

Let us also note that, from its very definition,

- The operation of matrix multiplication is *non-commutative*

$$\boldsymbol{AB} \neq \boldsymbol{CA} \tag{A.3.29}$$

- It is *distributive*, since for $\boldsymbol{A}$ and $\boldsymbol{B}$ of dimension $m \times p$ and $\boldsymbol{C}$ of dimension $p \times n$ one can write

$$(\boldsymbol{A} \;+\; \boldsymbol{B})\,\boldsymbol{C} \;=\; \boldsymbol{A}\,\boldsymbol{C} \;+\; \boldsymbol{B}\,\boldsymbol{C} \tag{A.3.30}$$

- It is *associative*, since for $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ of dimension $m \times p$, $p \times q$ and $q \times n$ respectively

$$(\boldsymbol{A}\,\boldsymbol{B})\,\boldsymbol{C} \;=\; \boldsymbol{A}\,(\boldsymbol{B}\,\boldsymbol{C}) \;=\; \boldsymbol{A}\,\boldsymbol{B}\,\boldsymbol{C} \tag{A.3.31}$$

- The transpose of a matrix product is the product of the transposes taken in reverse order

$$(\boldsymbol{A}\,\boldsymbol{B})^T \;=\; \boldsymbol{B}^T\,\boldsymbol{A}^T \tag{A.3.32}$$

Rank of a matrix

Let us consider a matrix $\boldsymbol{A}$ of dimension $m \times n$, decomposed in terms of its columns

$$\boldsymbol{A} = [\, \boldsymbol{a}_1 \; \boldsymbol{a}_2 \; \dots \; \boldsymbol{a}_n \,] \tag{A.3.33}$$

Its columns are said to be *linearly independent* if any linear combination of these does not vanish. In other terms, the columns of $\boldsymbol{a}$ are linearly independent if, for $\boldsymbol{b}$ arbitrary

$$\boldsymbol{b}^T = [\, b_1 \; b_2 \; \dots \; b_n \,] \tag{A.3.34}$$

one has always

$$\boldsymbol{A}\, \boldsymbol{b} \neq 0 \tag{A.3.35}$$

If (A.3.35) does not necessarily hold, then the columns of $\boldsymbol{A}$ are linearly dependent and one of them at least may be expressed as a linear combination of the others.

The *column rank* (or the *row rank*) of a matrix $\boldsymbol{A}$ is defined as the largest number of linearly independent columns (or rows) that can be extracted from $\boldsymbol{A}$.

It is possible to show that the row rank and the column rank of a matrix are equal. Therefore, we define the rank of matrix $\boldsymbol{A}$ as the common value of its row and column ranks.

The rank of matrix $\boldsymbol{A}$ is also equal to the dimension of the largest square submatrix with non-zero determinant that can be extracted from $\boldsymbol{A}$ through appropriate deletion of rows and/or columns.

A square matrix $\boldsymbol{A}$ having linearly independent rows and columns is said to have *maximum rank*. Conversely, if it has not maximum rank, it is said *singular*.

A square matrix $\boldsymbol{A}$ having maximum rank is also said nonsingular.

Inverse of a matrix

A non singular square matrix $\boldsymbol{A}$ possesses an inverse, denoted $\boldsymbol{A}^{-1}$, such that

$$\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I} \tag{A.3.36}$$

where $\boldsymbol{I}$ is the unit matrix. Its explicit expression can be obtained from Cramer's rule for the solution of linear systems of equations, and textbooks on numerical analysis present various techniques for calculating it numerically.

It is easy to show that

- The inverse of $\boldsymbol{A}$ is the transpose of the inverse

$$(\boldsymbol{A}^{-1})^T = (\boldsymbol{A}^T)^{-1} = \boldsymbol{A}^{-T} \tag{A.3.37}$$

Orthogonal matrix

A non-singular matrix which plays a fundamental role to describe the kinematics of a rigid body is the *orthogonal matrix*. It is such that

$$\boldsymbol{A}^{-1} = \boldsymbol{A}^T / det(\boldsymbol{A}) \tag{A.3.38}$$

where $det(\boldsymbol{A})$ stands for the determinant of $\boldsymbol{A}$. If $det(\boldsymbol{A}) = 1$, then $\boldsymbol{A}$ is *orthonormal*:

$$\boldsymbol{A}^{-1} = \boldsymbol{A}^T \tag{A.3.39}$$

As computing the inverse of a nonsingular matrix is a costly operation, it is important to know whether a matrix is orthonormal or not.

Let us further note that in most cases, a matrix is called orthogonal even if it is also orthonormal.

A.4 Matrix representation of vector operations

The matrix representation of vectors provides, as it will be seen, a powerful formalism for performing the main operations of vector calculus.

Once a specific cartesian reference frame is adopted in which vectors are represented in an extrinsic manner, any vector may be represented by a column matrix collecting its cartesian components in this frame.

Let $\vec{a}$ and $\vec{b}$ two vectors which we represent in matrix form by

$$\boldsymbol{a}^T = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix} \tag{A.4.1}$$

$$\boldsymbol{b}^T = \begin{bmatrix} b_x & b_y & b_z \end{bmatrix} \tag{A.4.2}$$

Vector sum

The vector sum $\vec{c} = \vec{a} + \vec{b}$ is equivalent to the matrix operation

$$\boldsymbol{c} = \boldsymbol{a} + \boldsymbol{b} \tag{A.4.3}$$

Equality of two vectors

The equality $\vec{a} = \vec{b}$ holds if and only if $\vec{a}$ and $\vec{b}$ have the same cartesian components in the same frame. Thus,

$$\boldsymbol{a} = \boldsymbol{b} \tag{A.4.4}$$

Scalar multiplication

The product of $\vec{a}$ by a scalar α, which gives the vector $\alpha\vec{a}$, has the matrix representation

$$\boldsymbol{c} = \alpha\boldsymbol{a} \tag{A.4.5}$$

Dot product

The dot product $\vec{a} \bullet \vec{b}$ of two vectors $\vec{a}$ and $\vec{b}$ can be written in matrix form

$$\boldsymbol{a}^T\boldsymbol{b} = \boldsymbol{b}^T\boldsymbol{a} = a_x b_x + a_y b_y + a_z b_z \tag{A.4.6}$$

Cross product

Let us define the *skew-symmetric* matrix associated to a vector $\vec{a}$, such that

$$\tilde{\boldsymbol{a}} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \tag{A.4.7}$$

The properties of the skew-symmetric matrix $\tilde{\boldsymbol{a}}$ are the following:

- Its transpose is such that

$$\tilde{\boldsymbol{a}}^T = -\tilde{\boldsymbol{a}} \tag{A.4.8}$$

- The scalar multiplication is such that

$$\alpha\,\tilde{\boldsymbol{a}} = \widetilde{\alpha\boldsymbol{a}} \tag{A.4.9}$$

- The vanishing of the cross-product of a vector by itself leads to

$$\tilde{\boldsymbol{a}}\boldsymbol{a} = 0 \tag{A.4.10}$$

- For any two vectors $\vec{a}$ and $\vec{b}$, one may verify that

$$\tilde{\boldsymbol{a}}\boldsymbol{b} = -\tilde{\boldsymbol{b}}\boldsymbol{a} \tag{A.4.11}$$

and

$$\tilde{\boldsymbol{a}}\,\tilde{\boldsymbol{b}} = \boldsymbol{b}\,\boldsymbol{a}^T - \boldsymbol{a}^T\boldsymbol{b}\,\boldsymbol{I} \tag{A.4.12}$$

so that

$$\tilde{\boldsymbol{a}}\,\tilde{\boldsymbol{b}} + \boldsymbol{a}\,\boldsymbol{b}^T = \tilde{\boldsymbol{b}}\,\tilde{\boldsymbol{a}} + \boldsymbol{b}\,\boldsymbol{a}^T \tag{A.4.13}$$

Appendix B

ELEMENTS OF QUATERNION ALGEBRA

B.1 Definition

A quaternion is defined a a 4-dimensional complex number

$$\hat{q} \;=\; q_0 \;+\; iq_1 \;+\; jq_2 \;+\; kq_3 \tag{B.1.1}$$

with i, j, k being imaginary unit numbers such that

$$\begin{aligned} i^2 \;&=\; j^2 \;=\; k^2 \;=\; -1 \\ jk \;&=\; -kj \;= i \\ ki \;&=\; -ik \;= j \\ ij \;&=\; -ij \;= k \end{aligned} \tag{B.1.2}$$

It can be alternatively written in vector notation

$$\hat{q} \;=\; q_0 \;+\; \vec{q} \;=\; q_0 \;+\; \boldsymbol{q} \tag{B.1.3}$$

where q_0 is the scalar part of the quaternion $\hat{q}$ while $\vec{q}$ or $\boldsymbol{q}$ are its vector part.

The *multiplication* rule is a direct consequence of the definition (B.1.1). In algebraic form, the resulting quaternion can be written:

$$\begin{aligned} \hat{p}\,\hat{q} \;&=\; (p_0 + ip_1 + jp_2 + kp_3)\;(q_0 + iq_1 + jq_2 + kq_3) \\ &=\; (p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3) \\ &\quad + i\;(p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2) \\ &\quad + j\;(p_0q_2 + p_2q_0 + p_3q_1 - p_1q_3) \\ &\quad + k\;(p_0q_3 + p_3q_0 + p_1q_2 - p_2q_1) \end{aligned} \tag{B.1.4}$$

In vector form the quaternion product is given by

$$\begin{aligned} \hat{p}\,\hat{q} \;&=\; (p_0 + \vec{p})\;(q_0 + \vec{q}) \\ &=\; p_0\;q_0 - \vec{p}\cdot\vec{q} \;+\; p_0\;\vec{q} \;+\; q_0\;\vec{p} \;+\; \vec{p}\times\vec{q} \\ &=\; p_0\;q_0 - \boldsymbol{p}^T\boldsymbol{q} \;+\; p_0\;\boldsymbol{q} \;+\; q_0\;\boldsymbol{p} \;+\; \tilde{\boldsymbol{p}}\;\boldsymbol{q} \end{aligned} \tag{B.1.5}$$

Because of the presence of the cross product in definition (B.1.5) the product is an associative but non-commutative operation.

The *conjugate quaternion* to $\hat{q}$ is defined as

$$\begin{aligned} \hat{q}^* \;&= q_0 \;-\; iq_1 \;-\; jq_2 \;-\; kq_3 \\ & q_0 \;-\; \vec{q} \end{aligned} \tag{B.1.6}$$

It is easily verified that the conjugate of a quaternion product is such that

$$(\hat{p}\;\hat{q})^* \;=\; \hat{q}^*\;\hat{p}^* \tag{B.1.7}$$

The *norm* of a quaternion is calculated by

$$||\hat{q}||^2 \;=\; (\hat{q}\;\hat{q})^* \;=\; q_0^2 \;+\; \vec{q}\cdot\vec{q} \tag{B.1.8}$$

In particular, $\hat{q}$ is a *unit quaternion* if

$$||\hat{q}|| \;=\; 1 \tag{B.1.9}$$

A quaternion $\hat{q}$ is a *vector quaternion* if

$$\hat{q}^* = 0 + \vec{q} \qquad \text{(B.1.10)}$$

in which case

$$\hat{q} + \hat{q}^* = 0 \qquad \text{(B.1.11)}$$

Vector quaternions describe position vectors, linear and angular velocity vectors, etc.

B.2 representation of finite rotations in terms of quaternions

Given a unit quaternion $\hat{e} = e_0 + \vec{e}$ and the position vector $\hat{x} = 0 + \vec{x}$, the finite rotation of $\vec{x}$ to a new position $\vec{y}$ may be represented by the triple quaternion product

$$\hat{y} = \hat{e}\ \hat{x}\ \hat{e}^* \qquad \text{(B.2.1)}$$

The proof holds by noting that

- $\hat{y}$ is also a vector quaternion

$$\hat{y} + \hat{y}^* = 0$$

- the length of $\hat{x}$ is conserved

$$||\hat{y}||^2 = \hat{y}\ \hat{y}^* = ||\hat{x}||^2$$

The *inverse rotation* is directly obtained in terms of the conjugate quaternion

$$\hat{x} = \hat{e}^*\ \hat{y}\ \hat{e} \qquad \text{(B.2.2)}$$

Let the position vector undergo *two successive rotations*

$$\begin{aligned} \hat{y} &= \hat{e}_1\ \hat{x}\ \hat{e}_1^* \\ \hat{z} &= \hat{e}_2\ \hat{y}\ \hat{e}_2^* = (\hat{e}_2\hat{e}_1)\ \hat{x}\ (\hat{e}_2\hat{e}_1)^* \end{aligned} \qquad \text{(B.2.3)}$$

The resulting rotation is given by

$$\hat{z} = \hat{e}\ \hat{x}\ \hat{e}^* \quad \text{with} \quad \hat{e} = \hat{e}_2\ \hat{e}_1 \qquad \text{(B.2.4)}$$

Clearly every unit quaternion can be expressed in the form

$$\hat{e} = \cos\alpha + \boldsymbol{n}\sin\alpha \qquad \text{(B.2.5)}$$

where $\boldsymbol{n}$ a unit vector.
If we perform the operations indicated in (B.2.2), we obtain:

$$\hat{e}\,\hat{x} = -\sin\alpha\ \boldsymbol{n}^T\boldsymbol{x} + \cos\alpha\ \boldsymbol{x} + \sin\alpha\ \tilde{\boldsymbol{n}}\ \boldsymbol{x} \qquad \text{(B.2.6)}$$

The vector character of the result is restored after performing the 'symmetric' operation:

$$\begin{aligned} \hat{y} =&\ \hat{e}\ \hat{x}\ \hat{e}^* \\ =&\ \sin^2\alpha\ (\boldsymbol{n}\cdot\boldsymbol{x}) + \cos^2\alpha\boldsymbol{x} + 2\ \sin\alpha\ \cos\alpha\ \tilde{\boldsymbol{n}}\ \boldsymbol{x} - \sin^2\alpha\ \widetilde{(\tilde{\boldsymbol{n}}\ \boldsymbol{x})}\ \boldsymbol{x} \end{aligned}$$

Which provides after developing

$$\hat{y} = 0 + (\cos 2\alpha\,\boldsymbol{I} + (1-\cos 2\alpha)\,\boldsymbol{n}\,\boldsymbol{n}^T + \sin 2\alpha\,\tilde{\boldsymbol{n}})\ \boldsymbol{x} \qquad \text{(B.2.7)}$$

Comparison with expression of rotation matrix in terms of Euler parameters shows this operation is a rotation of angle 2α about $\boldsymbol{n}$. This demonstrates that there is an *equivalence between Euler parameters and a double quaternion products.*

B.3 Matrix representation of quaternions

A quaternion may be represented in matrix form by 4-dimensional column matrix

$$\begin{aligned}\hat{q} &= [\, q_0 \; q_1 \; q_2 \; q_3 \,]^T \\ &= [\, q_0 \; \boldsymbol{q} \,]^T\end{aligned} \tag{B.3.1}$$

in which case the quaternion product $\hat{a} \;=\; \hat{p}\,\hat{q}$ can be written in either form

$$\hat{a} \;=\; \boldsymbol{A}_p\,\hat{q} \;=\; \boldsymbol{B}_q\hat{p} \tag{B.3.2}$$

with the 4×4 matrices

$$\boldsymbol{A}_p \;=\; \begin{bmatrix} p_0 & -\boldsymbol{p}^T \\ \boldsymbol{p} & p_0\,\boldsymbol{I} \,+\, \tilde{p} \end{bmatrix} \qquad \boldsymbol{B}_q \;=\; \begin{bmatrix} q_0 & -\boldsymbol{q}^T \\ \boldsymbol{q} & q_0\,\boldsymbol{I} \,-\, \tilde{q} \end{bmatrix} \tag{B.3.3}$$

where $\boldsymbol{I}$ is the unit matrix and $\tilde{\boldsymbol{q}}$ is a skew-symmetric matrix attached to the vector part $\boldsymbol{q}$

$$\tilde{\boldsymbol{q}} \;=\; \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \tag{B.3.4}$$

B.4 Matrix form of finite rotations

By using (B.3.2), the rotation operator can recast in the form

$$\hat{y} \;=\; \boldsymbol{A}\,\boldsymbol{B}^T\,\hat{x} \tag{B.4.1}$$

When computing the matrix product $\boldsymbol{A}\,\boldsymbol{B}^T$ one finds

$$\boldsymbol{A}\,\boldsymbol{B}^T \;=\; \begin{bmatrix} 1 & \mathbf{o}^T \\ \mathbf{o} & \boldsymbol{R} \end{bmatrix} \tag{B.4.2}$$

where the 3×3 submatrix $\boldsymbol{R}$ is the standard rotation operator. Developing the product (B.4.2) provides the classical result

$$\boldsymbol{R} \;=\; (2e_0^2 - 1)\boldsymbol{I} \;+\; 2\;(\boldsymbol{e}\boldsymbol{e}^T \;+\; e_0\hat{\boldsymbol{e}}) \tag{B.4.3}$$

where e_0, $\boldsymbol{e}$ are the components of the unit quaternion $\hat{e}$.
A simplified form of $\boldsymbol{R}$ is given by

$$\boldsymbol{R} \;=\; \boldsymbol{E}\,\boldsymbol{G}^T \tag{B.4.4}$$

with

$$\boldsymbol{E} \;=\; [\; -\boldsymbol{e} \quad e_0\,\boldsymbol{I} + \tilde{\boldsymbol{e}} \;] \quad \text{and} \quad \boldsymbol{G} \;=\; [\; -\boldsymbol{e} \quad e_0\,\boldsymbol{I} - \tilde{\boldsymbol{e}} \;] \tag{B.4.5}$$

where $\boldsymbol{E}$ and $\boldsymbol{G}$ verify the relationships:

$$\begin{aligned}\boldsymbol{E}\,\boldsymbol{E}^T &= \boldsymbol{G}\,\boldsymbol{G}^T \;=\; \boldsymbol{I} \\ \boldsymbol{E}^T\,\boldsymbol{E} &= \boldsymbol{G}^T\,\boldsymbol{G} \;=\; \boldsymbol{I} \;-\; \hat{\boldsymbol{e}}\,\hat{\boldsymbol{e}}^T \\ \boldsymbol{E}\,\hat{\boldsymbol{e}} &= \boldsymbol{G}\,\hat{\boldsymbol{e}} \;=\; \mathbf{o}\end{aligned} \tag{B.4.6}$$

B.5 Angular velocities in terms of quaternions

Let us start from the quaternion form of a finite rotation

$$\hat{y} \;=\; \hat{e}\,\hat{x}\,\hat{e}^* \tag{B.5.1}$$

where $\hat{x}$ is a vector quaternion describing the position of a point on a rigid body, and $\hat{e}$ is a unit quaternion (Euler parameters). If the length of $\hat{x}$ is assumed constant with time and that only $\hat{e}$ is time dependent, then the following equation describes *spherical motion*:

$$\hat{y}(t) \;=\; \hat{e}(t)\,\hat{x}\,\hat{e}^*(t) \tag{B.5.2}$$

The associated velocity field is

$$\dot{\hat{y}} \;=\; \dot{\hat{e}}\,\hat{x}\,\hat{e}^* \;+\; \hat{e}\,\hat{x}\,\hat{\dot{e}}^* \tag{B.5.3}$$

or in terms of $\hat{y}$

$$\dot{\hat{y}} \;=\; \dot{\hat{e}}\,\hat{e}^*\,\hat{y} \;+\; \hat{y}\,\hat{e}\,\hat{\dot{e}}^* \tag{B.5.4}$$

If one notes further that $\hat{e}\,\hat{e}^* \;=\; 1$ generates the identity

$$\hat{e}\,\hat{\dot{e}}^* \;+\; \dot{\hat{e}}\,\hat{e}^* \;=\; 0 \tag{B.5.5}$$

one deduces that the quaternion

$$\hat{\omega} \;=\; 2\,\dot{\hat{e}}\,\hat{e}^* \tag{B.5.6}$$

is also of vector type, and using the definition (B.5.6) equation (B.5.4) becomes

$$\dot{\hat{y}} \;=\; \frac{1}{2}\,(\hat{\omega}\,\hat{y} \;-\; \hat{y}\,\hat{\omega}) \tag{B.5.7}$$

It represents the velocity vector of P in terms of quantities in the frame attached to $\hat{y}$. Equations (B.5.1) and (B.5.7) are thus analog to the matrix expressions

$$\boldsymbol{y} \;=\; \boldsymbol{R}\,\boldsymbol{x} \quad \text{and} \quad \dot{\boldsymbol{y}} \;=\; \dot{\boldsymbol{R}}\,\boldsymbol{R}^T\,\boldsymbol{y} \tag{B.5.8}$$

meaning thus that the vector quaternion $\hat{\omega}$ represents the angular velocity matrix

$$\tilde{\omega} \;=\; \dot{\boldsymbol{R}}\,\boldsymbol{R}^T \tag{B.5.9}$$

Similarly, velocities can be calculated in the frame attached to $\hat{x}$

$$\hat{v} \;=\; \hat{e}^*\,\dot{\hat{y}}\,\hat{e} \tag{B.5.10}$$

or, making use of the equation (B.5.3)

$$\boldsymbol{v} \;=\; \hat{e}^*\,\dot{\hat{e}}\,\hat{x} \;+\; \hat{x}\,\dot{\hat{e}}^*\hat{e} \tag{B.5.11}$$

If one defines thus the quaternion vector

$$\hat{\omega}' \;=\; 2\,\hat{e}^*\,\dot{\hat{e}} \tag{B.5.12}$$

velocities in the frame attached to $\hat{x}$ may be written in the form

$$\hat{v} \;=\; \frac{1}{2}\,(\hat{\omega}'\,\hat{x} \;-\; \hat{x}\hat{\omega}') \tag{B.5.13}$$

Equation (B.5.13) is thus the quaternion analog of

$$\tilde{\omega}' \;=\; \boldsymbol{R}^T\,\dot{\boldsymbol{R}} \tag{B.5.14}$$

B.6 Matrix form of angular velocities

The matrix notation introduced in sections B.3 and B.4 allows us to rewrite (B.5.6) and (B.5.12) in the matrix forms

$$\hat{\omega} \;=\; 2\,\boldsymbol{B}(\hat{e}^*)\,\dot{\hat{e}} \quad \text{and} \quad \hat{\omega}' \;=\; 2\,\boldsymbol{A}(\hat{e}^*)\,\dot{\hat{e}} \tag{B.6.1}$$

Or if we limit ourselves to the vector part of $\hat{\omega}$ and $\hat{\omega}'$, one obtains the simplified expressions

$$\omega \;=\; 2\,\boldsymbol{E}\,\dot{\boldsymbol{e}} \quad \text{and} \quad \omega' \;=\; 2\,\boldsymbol{G}\,\dot{\boldsymbol{e}} \tag{B.6.2}$$

where $\boldsymbol{E}$ and $\boldsymbol{G}$ are 3×4 matrices extracted from $\boldsymbol{A}$ and $\boldsymbol{B}$

$$\boldsymbol{E} \;=\; [\,-\boldsymbol{e} \quad e_0\,\boldsymbol{I} \,+\, \tilde{\boldsymbol{e}}\,] \quad \text{and} \quad \boldsymbol{G} \;=\; [\,-\boldsymbol{e} \quad e_0\,\boldsymbol{I} \,-\, \tilde{\boldsymbol{e}}\,] \tag{B.6.3}$$

Explicitly, in terms off Euler parameters one gets

$$\omega \;=\; 2 \begin{bmatrix} -e_1 & e_0 & -e_3 & e_2 \\ -e_2 & e_3 & e_0 & -e_1 \\ -e_3 & -e_2 & e_1 & e_0 \end{bmatrix} \begin{bmatrix} \dot{e}_0 & \dot{e}_1 & \dot{e}_2 & \dot{e}_3 \end{bmatrix} \tag{B.6.4}$$

B.7 Examples

B.7.1 Example 1: Composition of rotations with quaternions

Find the equivalent angle-axis form of $\boldsymbol{R}(\boldsymbol{l}, \phi)$ to a rotation $\boldsymbol{R}(\boldsymbol{z}, 90°)$ followed by a rotation $\boldsymbol{R}(\boldsymbol{y}, 90°)$.

Using quaternion notations, one has

$$\hat{e} = \; e_0 \;+\; \boldsymbol{e} \;=\; \hat{e}_2\,\hat{e}_1$$

with

$$\hat{e}_1 = \; (\cos 45° \;+\; \boldsymbol{k}\; \sin 45°)\hat{e}_2 \qquad\qquad = \; (\cos 45° \;+\; \boldsymbol{j}\; \sin 45°)$$

So one gets

$$\begin{aligned} \hat{e} = &\; \cos^2 45° \;+\; (\boldsymbol{j} + \boldsymbol{k})\; \cos 45°\; \sin 45° \;+\; (\boldsymbol{j} \times \boldsymbol{k})\; \sin^2 45° \\ = &\; \frac{1}{2} \;+\; \frac{1}{\sqrt{3}}\,(\boldsymbol{i} + \boldsymbol{j} + \boldsymbol{k})\,\frac{\sqrt{3}}{2} \\ = &\; \cos 60° \;+\; \frac{1}{\sqrt{3}}\,(\boldsymbol{i} + \boldsymbol{j} + \boldsymbol{k})\; \sin 60° \end{aligned}$$

Therefore the overall transformation is equivalent to a rotation

$$\boldsymbol{R}(\boldsymbol{n}, 120°) \qquad \text{with} \quad \boldsymbol{n} \;=\; \frac{1}{\sqrt{3}}\,(\boldsymbol{i} + \boldsymbol{j} + \boldsymbol{k})$$

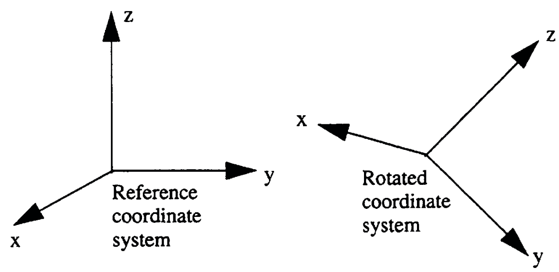


Figure B.7.1: The rotation of a coordinate system is described by a quaternion

B.7.2 Example 2: Using quaternions for determining robot orientation

Useful formula

The orientation of a coordinate system (such as that of a tool) can be described by a rotational matrix that gives the directions of the axes of the coordinate system in relation to a reference system (see Fig. B.7.1) The rotated system axes ($\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$) are vectors which can be expressed in the reference system as follows:

$$\begin{aligned}\boldsymbol{x} &= (x_1,\, x_2,\, x_3)\\ \boldsymbol{y} &= (y_1,\, y_2,\, y_3)\\ \boldsymbol{z} &= (z_1,\, z_2,\, z_3)\end{aligned}$$

This means that the x-component of the x-vector in the reference coordinate system will be x_1, the y-component will be x_2, etc.

These three vectors can be put together in a matrix, a rotational matrix, where each of the vectors form one of the columns:

$$\begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix}$$

A quaternion is a more concise way to describe this rotational matrix; one can calculate the quaternion from the rotational matrix:

$$\begin{aligned}
q_0 &= \frac{\sqrt{x_1 + y_2 + z_3 + 1}}{2} \\
q_1 &= \frac{\sqrt{x_1 - y_2 - z_3 + 1}}{2} \quad \text{with} \quad \text{sign}\, q_1 = \text{sign}\,(y_3 - z_2) \\
q_2 &= \frac{\sqrt{y_2 - x_1 - z_3 + 1}}{2} \quad \text{with} \quad \text{sign}\, q_2 = \text{sign}\,(z_1 - x_3) \\
q_3 &= \frac{\sqrt{z_3 - x_1 - y_2 + 1}}{2} \quad \text{with} \quad \text{sign}\, q_3 = \text{sign}\,(x_2 - y_1)
\end{aligned}$$

Moreover these four quantities are nor independent and linked by the relationship

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$$

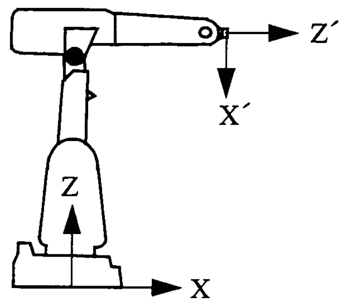


Figure B.7.2: The orientation of the wrist with respect to the base frame

Orientation of the wrist with respect to the base frame

A tool is oriented so that its $\boldsymbol{z}'$-axis points straight ahead (in the same direction as the $\boldsymbol{x}$-axis of the base coordinate system). The $\boldsymbol{y}'$-axis of the tool corresponds to the $\boldsymbol{y}$-axis of the base coordinate system (see Fig. B.7.2). How is the orientation of the tool defined in the position data?

The axes will then be related as followed:

$$\begin{aligned} \boldsymbol{x}' &= -\boldsymbol{z} &&= (0,\,0,\,-1) \\ \boldsymbol{y}' &= \boldsymbol{y} &&= (0,\,1,\,0) \\ \boldsymbol{z}' &= \boldsymbol{x} &&= (1,\,0,\,0) \end{aligned}$$

Which corresponds to the following rotational matrix:

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

The rotation matrix provides the corresponding quaternion:

$$\begin{aligned} p_0 &= \frac{\sqrt{0+1+0+1}}{2} = \frac{\sqrt{2}}{2} = 0,707 \\ p_1 &= \frac{\sqrt{0-1-0+1}}{2} = 0 \\ p_2 &= \frac{\sqrt{1-0-0+1}}{2} = \frac{\sqrt{2}}{2} = 0,707 \qquad \text{with} \quad \operatorname{sign} q_2 = \operatorname{sign}(1-(-1)) = + \\ p_3 &= \frac{\sqrt{0-0-1+1}}{2} = 0 \end{aligned}$$

Orientation of the tool with respect to the wrist frame

The direction of the tool is rotated about 30° about the $\boldsymbol{x}'$ and the $\boldsymbol{z}'$-axes in relation to the wrist coordinate system (see Fig. B.7.3). How is the orientation of the tool defined with respect to the wrist system ?

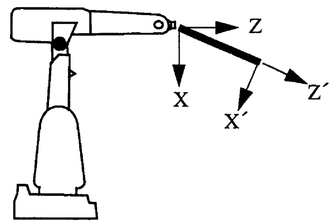


Figure B.7.3: The orientation of the tool with respect to the wrist frame

The axes will then be related as followed:

$$\boldsymbol{x}'' = (\cos 30°,\ 0,\ -\sin 30°)$$
$$\boldsymbol{y}'' = (0,\ 1,\ 0)$$
$$\boldsymbol{z}'' = (\sin 30°,\ 0,\ \cos 30°)$$

Which corresponds to the following rotational matrix:

$$\begin{bmatrix} \cos 30° & 0 & \sin 30° \\ 0 & 1 & 0 \\ -\sin 30° & 0 & \cos 30° \end{bmatrix}$$

The rotation matrix provides the corresponding quaternion:

$$q_0 = \frac{\sqrt{\cos 30° + 1 + \cos 30° + 1}}{2} = 0,9659$$
$$q_1 = \frac{\sqrt{\cos 30° - 1 - \cos 30° + 1}}{2} = 0$$
$$q_2 = \frac{\sqrt{1 - \cos 30° - \cos 30° + 1}}{2} = 0,2588 \qquad \text{with} \quad \text{sign}\, q_2 = \text{sign}\,(\sin 30° - (-\sin 30°)) = +$$
$$q_3 = \frac{\sqrt{\cos 30° - \cos 30° - 1 + 1}}{2} = 0$$

Orientation of the tool with respect to the base frame

Orientation of the tool system with respect to base frame system is given by undergoing the two elementary frame transformations. In quaternion notations the global transformation is described by the following quaternion $\hat{r}$:

$$\hat{r} = \hat{q}\,\hat{p}$$

Bibliography

[1] R.A. WEHAGE "Quaternions and Euler parameters. A brief exposition", in: *Computer Aided Analysis and Optimization of Mechanical System Dynamics*, E.J. HAUG, ed., Springer Verlag, 1984.

[2] K.W. Spring, "Euler Paramters and the Use of Quaternion Algebra in Manipulation of Finite Rotations: a review", *Mechanism and Machine Theory*, Vol. 21, 365-373, 1986.

[3] W.R. HAMILTON, *Elements of Quaternions*, Cambridge University Press, 1899.

[4] M. GERADIN and A. CARDONNA, "Kinematics and Dynamics of Rigid and Flexible Mechanisms Using Finite Elements and Quaternion Algebra", *Computational Mechanics*, 1987.