SPRINGER BRIEFS IN
APPLIED SCIENCES AND TECHNOLOGY

Paulo Flores

Concepts and Formulations for Spatial Multibody Dynamics

Springer

SpringerBriefs in Applied Sciences and Technology

More information about this series at http://www.springer.com/series/8884

Paulo Flores

Concepts and Formulations for Spatial Multibody Dynamics

Springer

Paulo Flores
Department of Mechanical Engineering
University of Minho
Guimaraes
Portugal

ISSN 2191-530X ISSN 2191-5318 (electronic)
SpringerBriefs in Applied Sciences and Technology
ISBN 978-3-319-16189-1 ISBN 978-3-319-16190-7 (eBook)
DOI 10.1007/978-3-319-16190-7

Library of Congress Control Number: 2015933371

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Preface

As the name suggests, a multibody system is an assembly of several bodies connected to each other by joints and acted upon by forces. A body, which can be rigid or flexible, is composed of a collection of material points. A joint allows for certain degrees of freedom and constrains others. In practice, joints are connection devices such as bearings, rod guides, etc., which from a mathematical point of view are denominated as revolute joints, translational joints, etc., according to the relative degrees of motion permitted. The forces can have different sources and different levels of complexity. Dynamics of multibody systems is based on classical mechanics and has a long and prolific history. Multibody systems serve as a basis for many models of mechanical systems and have been applied in many areas of science and engineering. Multibody systems are often used to analyze biological and human locomotion. Controlled systems are frequently prototyped through computer simulation of multibody models. There are also applications in medical, robotics, space subsystems, and computer games. A free or unconstrained material point is the simplest multibody system that can be studied by applying the equations of motion established with genial acumen by Newton. In turn, D'Alembert considered a system of constrained rigid bodies, in which the distinction between the applied and reaction forces was established. A systematic analysis of constrained multibody systems was developed by Lagrange. However, it was only during the last half of the century that multibody dynamics received conspicuous attention thanks to the impressive computer progresses verified at both software and hardware levels. As a consequence, a great variety of methodologies have been proposed, despite the fact that all of them can be derived from a few fundamental principles of mechanics. The main purpose of this work is to present, in a review manner, the fundamental concepts and formulations for spatial multibody dynamics. The following material does not claim for completeness nor it is designed to substitute a textbook. It provides the reader with the basis background on the issue of spatial multibody dynamics and it might be helpful for understanding the methodologies and approaches offered in the present work.

Paulo Flores

Contents

1 Definition of Multibody System 1
References. 3

2 Fundamental Concepts in Multibody Dynamics 5
References. 9

3 Global and Local Coordinates 11
References. 14

4 Euler Angles, Bryant Angles and Euler Parameters 15
References. 22

5 Angular Velocity and Acceleration 23
References. 26

6 Vector of Coordinates, Velocities and Accelerations 27
References. 29

7 Kinematic Constraint Equations 31
References. 35

8 Basic Constraints Between Two Vectors 37
References. 41

9 Kinematic Joints Constraints 43
References. 48

10 Equations of Motion for Constrained Systems 49
References. 52

11 Force Elements and Reaction Forces 55
References. 59

12 Methods to Solve the Equations of Motion 61
References. 66

13 Integration Methods in Dynamic Analysis 67
References. 74

14 Correction of the Initial Conditions 75
References. 78

15 Demonstrative Example of Application 79
References. 82

Chapter 1
Definition of Multibody System

Abstract This chapter presents a general view of multibody system concept and definition by describing the main features associated with spatial systems. The mechanical components, which can be modeled as rigid or flexible, are constrained by kinematic pair of different types. Additionally, the bodies can be actuated upon by force elements and external forces due to interaction with environment. This chapter also presents some examples of application of multibody systems that can include automotive vehicles, mechanisms, robots and biomechanical systems.

Keywords Multibody systems · Definition · Spatial systems

In a simple manner, it can be said that a general multibody system (MBS) embraces two main characteristics, namely: (*i*) mechanical components that describe large translational and rotational displacements and (*ii*) kinematic joints that impose some constraints or restrictions on the relative motion of the bodies. In other words, a multibody system encompasses a collection of rigid and/or flexible bodies interconnected by kinematic joints and possibly some force elements (Nikravesh 2008). Driving elements and prescribed trajectories for given points of the system components, can also be represented under this general concept of multibody system. Figure 1.1 depicts an abstract representation of a multibody system (Flores et al. 2008).

The bodies that belong to a multibody system can be considered as rigid or flexible. A body is said to be rigid when its deformations are assumed to be small such that they do not affect the global motion produced by the body. Such a body can translate and rotate, but it can not change its shape. In contrast to this concept, a flexible body has an elastic structure. In the three-dimensional space, the motion of a free rigid body can be fully described by using six generalized coordinates associated with the six degrees of freedom. In turn, when a body includes some amount of flexibility, it has six rigid degrees of freedom plus the number of generalized coordinates necessary to describe the deformations (Shabana 1989). The expression flexible multibody system refers to a system holding deformable bodies with internal dynamics. In fact, rigid bodies are a representation of reality because bodies are not absolutely rigid in nature. However, in a good number of common

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_1

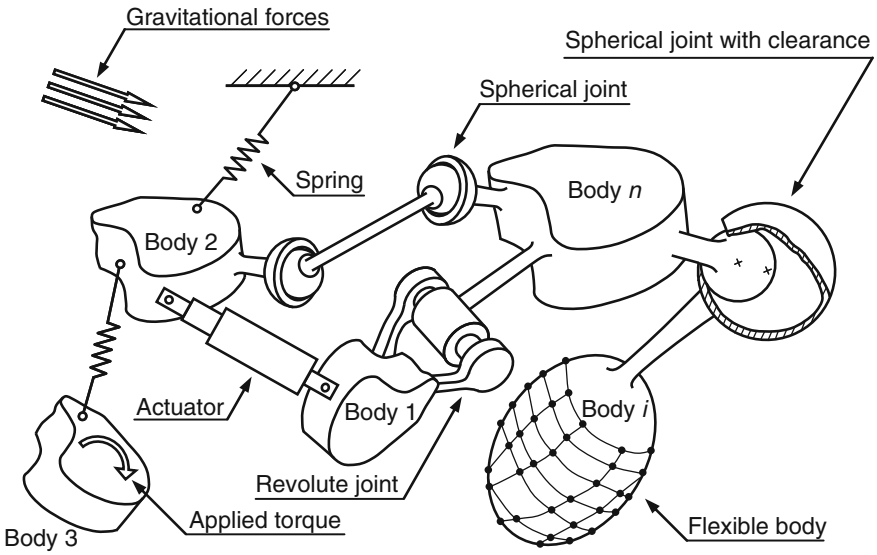

Fig. 1.1 Abstract representation of a multibody system with its most significant components: bodies, joints and forces elements

applications, the bodies are significantly stiff and, therefore, their flexibility can be disregarded and the bodies can be considered to be perfectly rigid. This assumption simplifies, in a significant manner, the process of modeling of multibody systems. Within the scope of the present work, only rigid bodies are considered.

By and large, the kinematic joints that can exist in multibody systems constrain the relative motion between the bodies connected by them. While the force elements represent the internal forces that are produced in the system and they are associated with the relative motion of the bodies. Two of the most typical kinematic joints employed in multibody systems are the revolute and the translational joints. The forces applied over the multibody system components can be the result of springs, dampers, actuators or external forces. External applied forces of different nature and different level of complexity can act on a multibody system with the purpose to simulate the interactions among the system components and between these and the surrounding environment (Nikravesh 1988; Schiehlen 1990).

A multibody system can be used to study the kinematic and dynamic motion characteristics of a wide variety of systems in a large number of engineering fields of application. Multibody systems can vary from very simple to highly complex. There is no doubt that multibody systems are ubiquitous in engineering and research activities, such as robotics (Zhu et al. 2006), automobile vehicles (Ambrósio and Veríssimo 2009), biomechanics (Silva and Ambrósio 2002), mechanisms (Flores 2011), railway vehicles (Pombo and Ambrósio 2008), just to mention a few. Figure 1.2 shows two multibody system examples of application,

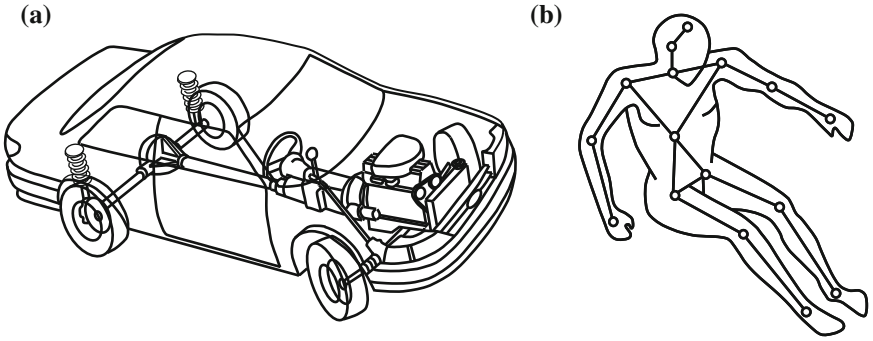

Fig. 1.2 Examples of application of multibody systems: **a** automobile vehicle model; **b** human biomechanical model

which result from the association of structural and mechanical subsystems with the purpose to transmit or transform a given motion.

References

Ambrósio J, Veríssimo P (2009) Sensitivity of a vehicle ride to the suspension bushing characteristics. J Mech Sci Technol 23:1075–1082

Flores P (2011) A methodology for quantifying the position errors due to manufacturing and assemble tolerances. J Mech Eng 57(6):457–467

Flores P, Ambrósio, J, Claro, JCP, Lankarani, HM (2008) Kinematics and dynamics of multibody systems with imperfect joints: models and case studies. In: Lecture notes in applied and computational mechanics, vol 34, Springer, Berlin

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Nikravesh PE (2008) Planar multibody dynamics: formulation, programming, and applications. CRC Press, London

Pombo J, Ambrósio J (2008) Application of a wheel-rail contact model to railway dynamics in small radius curved tracks. Multibody Sys Dyn 19:91–114

Schiehlen W (1990) Multibody systems handbook. Springer, Berlin

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Silva MPT, Ambrósio JAC (2002) Kinematic data consistency in the inverse dynamic analysis of biomechanical systems. Multibody Sys Dyn 8:219–239

Zhu W-H, Piedboeuf J-C, Gonthier Y (2006) A dynamics formulation of general constrained robots. Multibody Sys Dyn 16:37–54

Chapter 2
Fundamental Concepts in Multibody Dynamics

Abstract In this chapter, the fundamental ingredients related to formulation of the equations of motion for multibody systems are described. In particular, aspects such as degrees of freedom, types of coordinates, basic kinematics joints and types of analysis in multibody systems are briefly characterized. Illustrative examples of application are also presented to better clarify the fundamental issues for spatial rigid multibody systems, which are of crucial importance in the formulation development of mathematical models of mechanical systems, as well as its computational implementation.

Keywords Degrees of freedom · Types of coordinates · Kinematic joints

Prior to establish the equations of motion that govern the dynamic behavior of multibody systems, it is first necessary to select the manner how to describe them. The description variables must be able to characterize, at any instant of time, the configuration of the system, that is, the position of all the material points that compose the bodies. The description variables, also called generalized coordinates, must uniquely define the position of the system components at any instant of time during the multibody system analysis. The expression generalized coordinates is employed to include both linear and angular coordinates (Huston 1990).

The minimum number of variables necessary to fully describe the configuration of a system is denominated as degrees of freedom (DoF) of the system, or simply mobility (Müller 2009). When the configuration of a multibody system is completely defined by the orientation of one of its bodies, the system is said to have one degree of freedom. The number of degrees of freedom can also be defined as the number of independent generalized coordinates required to uniquely describe the configuration of a system. It is evident that the knowledge of the number of degrees of freedom is of prime importance in the processes of modeling and analysis of multibody systems. It is known that for the spatial case, each body has six degrees of freedom. Introducing a kinematic joint to a system, the total number of DoF will be reduced by the number of constraints imposed by the joint. It is clear that the number of constraints depends on the number and type of joint applied to the system, where the constraints must be independent from one another. The number

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_2

of degrees of freedom of a multibody system can be evaluated as the difference between the system coordinates and the number of independent constraints. The mathematical expression that summarizes this idea is known as the Grüebler-Kutzback criterion and is written as (Shigley and Uicker 1995)

$$n_{DoF} = 6n_b - m \tag{2.1}$$

where n_b represents the number of bodies that compose the multibody system and m is the number of independent constraints. For example, the spatial four-bar mechanism illustrated in Fig. 2.1 has six spherical joint constraints and ten revolute joint constraints, yielding two degrees of freedom.

Determining the number of degrees of freedom of a multibody system is mostly the first step in analyzing mechanical system, which typically consist of several bodies interconnected by different types of joints and force elements. When the number of degrees of freedom is negative, it denotes an over constrained non solvable mechanism. Zero or null degrees of freedom represents a structure, that is, a nonmovable system. Finally, when a multibody system has a positive number of degrees of freedom, it indicates a resolvable mechanism.

It is not unanimous and it is not a simple task either to define a criterion to classify the different types of coordinates that can be used to describe the configuration of multibody systems. A general and broad embracing rule to group the generalized coordinates is to divide them into independent and dependent coordinates (Wehage

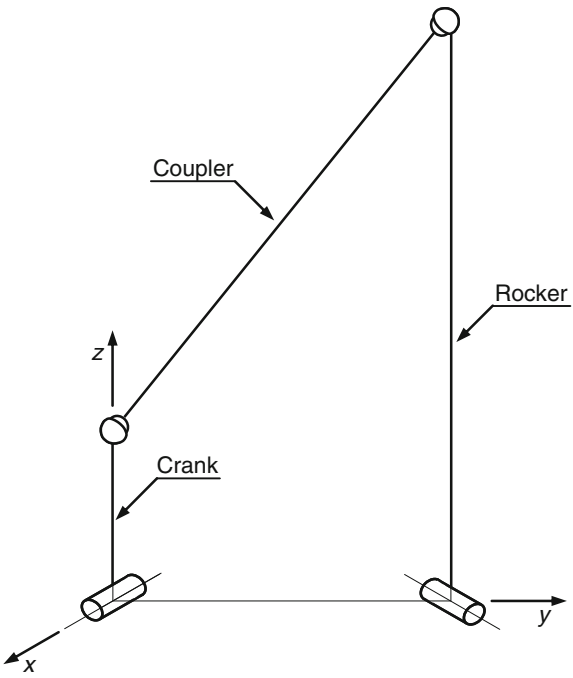

Fig. 2.1 Spatial four-bar linkage with two degrees of freedom

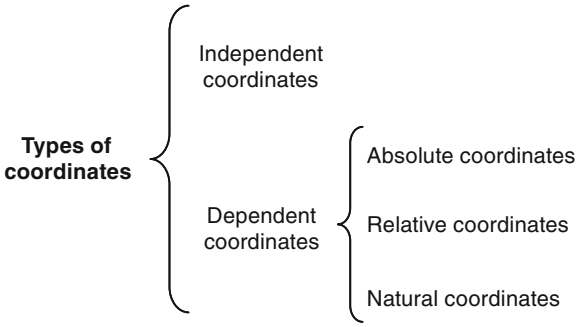

Fig. 2.2 Types of coordinates frequently used in multibody systems formulation

and Haug 1982). The independent coordinates are free to vary arbitrarily, while the dependent coordinates are required to satisfy the equations of constraints. Additionally, the dependent coordinates are classified as absolute coordinates (Orlandea et al. 1977), relative coordinates (Chace 1967) and natural coordinates (Jalón and Bayo 1994). Figure 2.2 summarizes the different types of coordinates most frequently used to describe the configuration of multibody systems.

In general, there are different manners of describing the configuration of a multibody system. In other words, there are many types of coordinates that can be helpful in the formulation of the equations of motion for multibody systems. The dilemma of selection of the type of coordinates to be used depends on the type of problem to be analyzed. In fact, the choice of the most appropriate set of coordinates is not indifferent, being a tradeoff between the advantages and drawbacks associated with each type of coordinates. A valuable comparison of the main types of coordinates are presented and discussed by Nikravesh (1988), Shabana (1989) and Jalón and Bayo (1994), where the pros and cons of each type of coordinates are highlighted. In particular, Shabana (1989) called attention for the selection of the most adequate type of coordinates to be used when modeling flexible multibody systems, which is a much more relevant task.

It is known that the degrees of freedom in a multibody system are directly related to the types of kinematic constraints considered, namely, those associated with kinematic joints. Furthermore, each type of joint allows for certain relative motions between adjacent bodies and constrains others. Figure 2.3 illustrates four of the most basic and frequently used kinematic joints when modeling multibody systems,

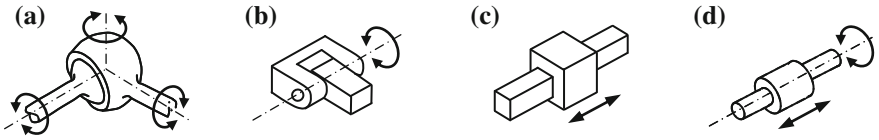

Fig. 2.3 Basic kinematic joints used in multibody systems: **a** spherical joint; **b** revolute joint; **c** translational joint; **d** cylindrical joint

in which their denomination, and relative degrees of freedom permitted are also represented (Reuleaux 1963).

Multibody systems methodologies include the main two phases: (*i*) development of mathematical models of multibody systems and (*ii*) implementation of computational procedures to perform the simulation, analysis and optimization of the global motion produced. Modeling or formulation is the process of generating and assembling the necessary equations of motion, when solved, would reveal the behavior of a multibody system. In a simple manner, there are two modeling approaches that can be considered, namely the point coordinates formulation and the body coordinates formulation. Broadly, it can be said that in the point coordinates formulation, the coordinates represent the joints and the constraints represent the bodies, whereas in the body coordinates formulation, the coordinates represent the bodies and the constraint represent the joints (Nikravesh 2008).

By describing the geometric configuration of a system with point coordinates formulation, the multibody system is represented as a multiple particle system. This collection of interconnected points usually stands for the joints in the system. Then, each point is assigned to a set of coordinates for which the kinematic constraints are constructed. Thus, the number of constraint equations only depends on the number and type of joint applied to the system. Using this formulation, the coordinates represent the joints and the applied constraints represent the bodies. Although this formulation could be realized in a computer program, the coordinates are not associated with the bodies. The analysis of a multibody system can be more convenient if the governing equations are solved for coordinates which correspond to the bodies directly.

The body coordinates formulation is a systematic approach to obtain the equations of motion for multibody systems based on the Newton-Euler equations. While other formulations describe the equations of motion in terms of generalized coordinates and generalized velocities, this formulation includes all coordinates and velocities of the involved bodies, which are expressed as the absolute coordinates and velocities. The resulting number of equations is large compared with other methods and, therefore, inappropriate for solving by hand. However, the equations are rather simple, although nonlinear, versatile and very suitable for the implementation in a computational program.

In a broad sense, the analysis of mechanical systems may be performed statically or dynamically. While statics denotes the study of stationary systems, i.e., time invariant systems, dynamics deals with the study of moving systems, i.e., systems whose behavior is time dependent. Furthermore, the branch of dynamics can be divided into two main disciplines, namely the kinematics and kinetics. In the kinematic analysis, the geometric aspects of motion are considered independently of forces that produce the motion. More precisely, kinematics deals with the study of the displacement, velocity and acceleration. In turn, kinetics is the study of the motion characteristics and the relation to the forces that produce the motion. Unlike the case of static and kinematic analysis, where only algebraic equations are utilized, in the kinetic analysis, the motion of a mechanical system is described by second-order differential equations (Nikravesh 1988).

It is very common to refer kinetic analysis as dynamic analysis because kinetic analysis must be based on the knowledge of the kinematic analysis of a system as well. Therefore, in this work, the term dynamic will be used instead of kinetic. In studying the dynamics of a mechanical system, there are two different types of analysis that can be performed, namely forward dynamics and inverse dynamics. In the forward dynamic analysis, the external forces acting on the bodies of a system are known and the resulting motion is obtained by solving the equations of motion. On the other hand, in the inverse dynamic analysis, a specific motion for a multibody system is sought and the objective is to determine the forces that are required to produce such a motion. In the context of the present work, methods of kinematics and forward dynamics are employed.

References

Chace MA (1967) Analysis of the time-dependence of multi-freedom mechanical systems in relative coordinates. J Eng Ind 89:119–125

Huston RL (1990) Multibody dynamics. Butterworth-Heinemann, Boston

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Müller A (2009) Generic mobility of rigid body mechanisms. Mech Mach Theor 44(6):1240–1255

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs, New Jersey

Nikravesh PE (2008) Planar multibody dynamics: formulation, programming, and applications. CRC Press, London

Orlandea N, Chace MA, Calahan DA (1977) A sparsity oriented approach to the dynamic analysis and design of mechanical systems—Part 1 and 2. J Eng Ind 99:773–784

Reuleaux F (1963) The kinematics of machinery. Dover, New York

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Shigley JE, Uicker JJ (1995) Theory of machines and mechanisms. McGraw Hill, New York

Wehage RA, Haug EJ (1982) Generalized coordinate partitioning for dimension reduction in analysis of constrained systems. J Mech Des 104:247–255

Chapter 3
Global and Local Coordinates

Abstract This chapter described the global and local coordinate systems utilized in the formulation of spatial multibody systems. Global coordinate system is considered in the present work to denote the inertia frame. Additionally, body-fixed coordinate systems, also called local coordinate systems, are utilized to describe local properties of points that belong to a particular body. Furthermore, the process of transforming local coordinates into global coordinates is characterized by considering a transformation matrix. In the present work, Cartesian coordinates are utilized to locate the center of mass of each rigid body, as well as the location of any point that belongs to a body.

Keywords Global coordinates · Local coordinates · Transformation matrix

Displacements, velocities and accelerations are quantities frequently used to characterize the configuration and motion properties of the multibody systems (Shabana 1989). For this purpose, a proper system of coordinates must be adopted, which includes the global and local systems of coordinates. The expression global coordinate system, which is represented by three orthogonal axes that are rigidly connected at a point called origin of this system, is utilized to represent the global, absolute or inertial frame of reference. In the present work, the global coordinate system is denoted by xyz. In addition, a body-fixed or local coordinate system is considered to define local properties of points that belong to a body. This local system of coordinates is, in general, attached to the center of mass of the bodies and is denoted by $\xi\eta\zeta$. This local coordinate system translates and rotates with the body motion and, consequently, its location and rotation vary with time (Nikravesh 1988).

A free particle i moving in three-dimensional space can be located by three independent variables. Since a particle does not have dimension, it can be seen as a point the three-dimensional Euclidean geometry that can be described by a position vector $\mathbf{r}$ that contains three scalar values as

$$\mathbf{r}_i = \{ x_i \quad y_i \quad z_i \}^T \tag{3.1}$$

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_3

Similarly, a free rigid body i in the same environment can be described by defining the position of one arbitrary point located on the body and the rotation of the body around that point. Therefore, it can be observed that there are infinite similar manners to define the spatial position of a rigid body, where some are more convenient than others (Schiehlen 1990; Jalón and Bayo 1994). The location and orientation of a body can be expressed with respect to any reference system. The generalized coordinates that describe the positions of the elements of a multibody system can be either with respect to each other or with respect to a common reference frame. The representation of a single body i with respect to a globally fixed xyz coordinate system, which is an absolute system, is shown in Fig. 3.1. The body possesses its own coordinate system, the body-fixed coordinate system denoted as $\xi_i\eta_i\zeta_i$. Vector $\mathbf{r}$, as expressed in Eq. (3.1), defines the location of the origin the local coordinate system. This vector represents the translational coordinates of the body. Another set of coordinates is needed to express the orientation of the body with respect to the global coordinate system.

The adequate chosen of the localization of the local coordinate system can simplify the definition of points and vectors on a body. For instance, point P_i located on body i in Fig. 3.1 can be defined from the origin of the local reference frame by vector $\mathbf{s}_i^P$, in such way that its direct location with respect to the global system can be expressed as (Flores et al. 2008)

$$\mathbf{r}_i^P = \mathbf{r}_i + \mathbf{s}_i^P \tag{3.2}$$

The same point can also be described with respect to the local coordinates system by $\mathbf{s}_i'^P$, that contains the local components of vector $\mathbf{s}_i^P$, and, therefore, $\mathbf{s}_i'^P$ is a constant vector for rigid bodies. In turn, the components of vector $\mathbf{s}_i^P$ vary if the body moves. The relation between $\mathbf{s}_i'^P$ and $\mathbf{s}_i^P$ is described by a transformation, which only depends on the relative orientation of the body with respect to the global frame in such way that

$$\mathbf{s}_i^P = \mathbf{A}_i\mathbf{s}_i'^P \tag{3.3}$$

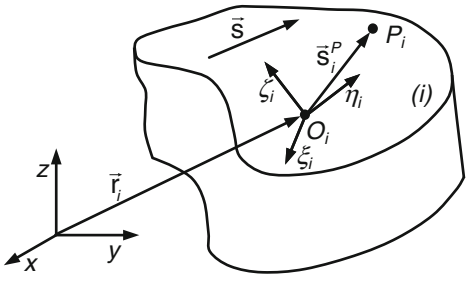

Fig. 3.1 Location of an unconstrained body i in the three-dimensional space

where $\mathbf{A}_i$ is a 3 × 3 rotation matrix that described the orientation of the local coordinate system with respect to the global frame. This rotation matrix introduces a general transformation from local coordinates to global coordinates. The global and local coordinates of the point P_i can be written as

$$\mathbf{s}_i^P = \begin{Bmatrix} x_i^P \\ y_i^P \\ z_i^P \end{Bmatrix} \quad \text{and} \quad \mathbf{s}_i'^P = \begin{Bmatrix} \xi_i^P \\ \eta_i^P \\ \zeta_i^P \end{Bmatrix} \tag{3.4}$$

Matrix $\mathbf{A}_i$ can also be utilized to transform components of vectors, such as vector $\mathbf{s}$ in Fig. 3.1, in a similar manner as

$$\mathbf{s}_i = \mathbf{A}_i \mathbf{s}_i' \tag{3.5}$$

in which $\mathbf{s}_i$ and $\mathbf{s}_i'$ have the following global and local components

$$\mathbf{s} = \begin{Bmatrix} s_x \\ s_y \\ s_z \end{Bmatrix} \quad \text{and} \quad \mathbf{s}_i' = \begin{Bmatrix} s_\xi \\ s_\eta \\ s_\zeta \end{Bmatrix} \tag{3.6}$$

Matrix $\mathbf{A}_i$ can be expressed in terms of direction cosines as

$$\mathbf{A}_i = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{3.7}$$

where the columns of $\mathbf{A}_i$ contain components of three unit vectors $\mathbf{u}_{\xi i}$, $\mathbf{u}_{\eta i}$ and $\mathbf{u}_{\zeta i}$ projected onto the *xyz* axes, such as

$$\mathbf{u}_{\xi_i} = \begin{Bmatrix} a_{11} \\ a_{21} \\ a_{31} \end{Bmatrix}, \quad \mathbf{u}_{\eta_i} = \begin{Bmatrix} a_{12} \\ a_{22} \\ a_{32} \end{Bmatrix}, \quad \mathbf{u}_{\zeta_i} = \begin{Bmatrix} a_{13} \\ a_{23} \\ a_{33} \end{Bmatrix} \tag{3.8}$$

It should be noted that

$$\mathbf{u}'_{\xi_i} = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}, \quad \mathbf{u}'_{\eta_i} = \begin{Bmatrix} 0 \\ 1 \\ 0 \end{Bmatrix}, \quad \mathbf{u}'_{\zeta_i} = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} \tag{3.9}$$

References

Flores P, Ambrósio, J, Claro, JCP, Lankarani, HM (2008) Kinematics and dynamics of multibody systems with imperfect joints: models and case studies. In: Lecture notes in applied and computational mechanics, vol 34. Springer, Berlin

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs, New Jersey

Schiehlen W (1990) Multibody systems handbook. Springer, Berlin

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 4
Euler Angles, Bryant Angles and Euler Parameters

Abstract This chapter deals with the different approaches for describing the rotational coordinates in spatial multibody systems. In this process, Euler angles and Bryant angles are briefly characterized. Particular emphasis is given to Euler parameters, which are utilized to describe the rotational coordinates in the present work. In addition, for all the types of coordinates considered in this chapter, a characterization of the transformation matrix is fully described.

Keywords Euler angles · Bryant angles · Euler parameters

The location of a rigid body i in the three-dimensional space can be defined by three translational coordinates and three rotational coordinates that describe the origin and orientation of the body-fixed coordinate system $\xi_i\eta_i\zeta_i$ attached to the body with respect to the global frame xyz, as it is illustrated in Fig. 4.1a. For the purpose of concentrating on the rotational coordinates of a body, let eliminate the translational coordinates by allowing the global and local coordinates systems to coincide at the origin, as Fig. 4.1b shows (Nikravesh 1988).

The orientation of a rigid body i can be specified by a transformation matrix, the elements of which may be expressed in terms of suitable sets of coordinates, such as Euler angles, Bryant angles or Euler parameters. Since the motion of the body is continuous, the transformation matrix must be a continuous function of time. It is known that the nine direction cosines present in the rotational transformation matrix $\mathbf{A}_i$ expressed in Eq. (3.7) define the orientation of the $\xi_i\eta_i\zeta_i$ axes. Since only three direction cosines are independent, six constraints must be considered to determine matrix $\mathbf{A}_i$. However, this manner to obtain the rotational coordinates is neither practical nor convenient. Thus, alternative orientation coordinates are required.

The Euler angles are three angles that can be utilized to describe the orientation of a rigid body. Euler angles represent three composed rotations that move a reference frame to a given referred frame. This is equivalent to saying that any orientation can be achieved by composing three elemental rotations, i.e., rotations about a single axis, and also equivalent to saying that any rotation can be decomposed as a product of three elemental rotation matrices. Thus, the angular orientation of a given body-fixed coordinate system can be envisioned to be the

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_4

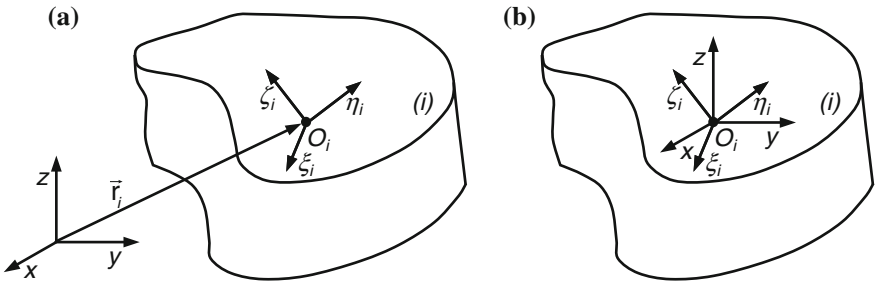

Fig. 4.1 **a** Translation and rotation; **b** pure rotation

result of three successive rotations. The three angles of rotation corresponding to these three successive rotations are defined as Euler angles. This is a minimal representation for the orientation of a rigid body since there is no need for constraint equations. Among the twelve possible conventions of the Euler angles, without loss of generality, the *zxz* convention is considered here. All the conventions are the result of three consecutive rotations about three different axes. Thus, depending on the choice of the rotational axis, different definitions can be found. However, the most common convention is the *zxz* convention. Another common convention is the *xyz* convention, being the parameters associated with this convention often known as Bryant angles.

The sequence of rotations employed in the *zxz* convention starts by rotating the initial *xyz* coordinate system counterclockwise about the *z*-axis by an angle ψ. The resulting coordinate system is denoted by $\xi''\eta''\zeta''$. In the second step, this intermediate coordinate system $\xi''\eta''\zeta''$ is rotated counterclockwise about ξ''-axis by an angle θ to produce another intermediate coordinate system labeled $\xi'\eta'\zeta'$. Finally, this last coordinate system is rotated counterclockwise about the ζ'-axis by an angle σ to produce the desired $\xi\eta\zeta$ system of axes. The various phases of this sequence are illustrated in Fig. 4.2. The angles ψ, θ and σ, which are the Euler angles, completely specify the orientation of the $\xi\eta\zeta$ coordinate system relative to the *xyz* frame and can, therefore, act as a set of three independent coordinates (Landau and Lifschitz 1976; Goldstein 1980).

When using Euler angles, the elements of the complete rotational transformation matrix **A** can be obtained as the triple product of the matrices that define the planar rotations, that is, the elemental planar matrices (Nikravesh 1988)

$$\mathbf{D} = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\theta & -s\theta \\ 0 & s\theta & c\theta \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} c\sigma & -s\sigma & 0 \\ s\sigma & c\sigma & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.1}$$

in which c ≡ cos and s ≡ sin. Hence, **A** = **DCB** can be written as

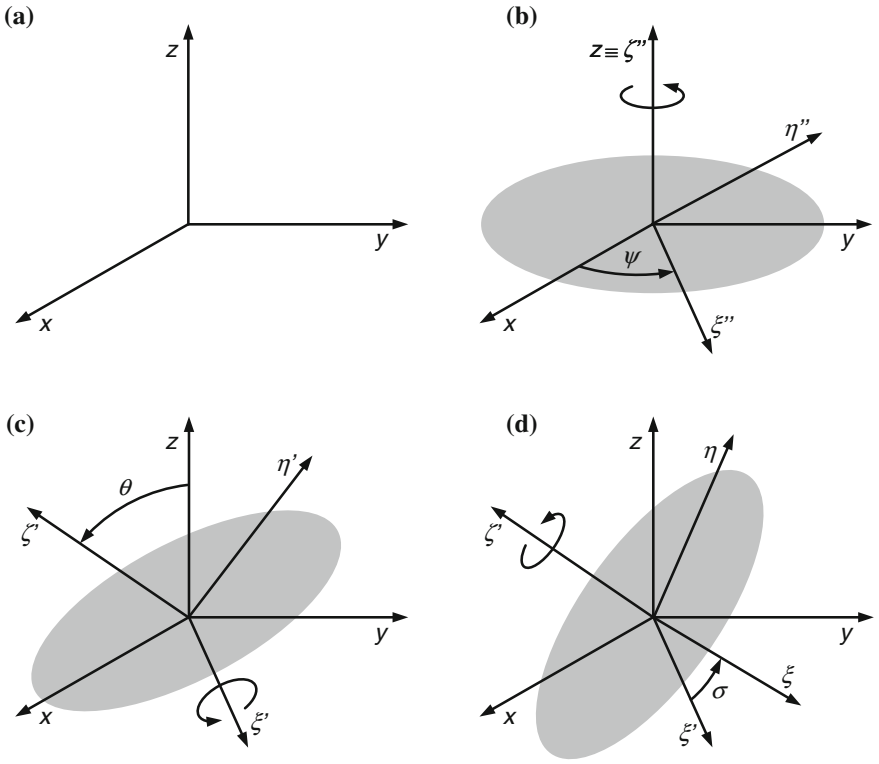

Fig. 4.2 Different stages of rotation for Euler angles. **a** Initial global system coordinate; **b** First rotation; **c** Second rotation; **d** Third rotation

$$\mathbf{A} = \begin{bmatrix} c\psi c\sigma - s\psi c\theta s\sigma & -c\psi s\sigma - s\psi c\theta c\sigma & s\psi s\theta \\ s\psi c\sigma + c\psi c\theta s\sigma & -s\psi s\sigma + c\psi c\theta c\sigma & -c\psi s\theta \\ s\theta s\sigma & s\theta c\sigma & c\theta \end{bmatrix} \tag{4.2}$$

At this stage, it should be noted that the resulting transformation matrix is highly nonlinear in terms of the three Euler angles. Since matrix multiplication does not commute, the order of the axes which one rotates about affect the result (Shabana 1989). Furthermore, if the three rotational angles are not chosen correctly, the rotated frame will not coincide with the $\xi\eta\zeta$ desired coordinate system. In spite of their popularity, Euler angles suffer from the drawback of representation of singularities. In fact, when $\sin\theta = 0$, the axes of the first and third rotations coincide, so that ψ and σ cannot be distinguished. This fact is visible by setting $\theta = 0$ in Eq. (4.2) to obtain

$$\mathbf{A} = \begin{bmatrix} c(\psi + \sigma) & -s(\psi + \sigma) & 0 \\ s(\psi + c) & -c(\psi + \sigma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

As it was mentioned above, the Bryant angles are the *xyz* convention of the Euler angles. In a similar manner, the *xyz* frame is rotated three times. The first rotation may be carried out counterclockwise about the *x*-axis through an angle ϕ_1. The resulting coordinate system is labeled $\xi''\eta''\zeta''$. Then, a second rotation, through an angle ϕ_2 counterclockwise about the η''-axis, produces the intermediate coordinate system $\xi'\eta'\zeta'$. Finally, the third rotation, counterclockwise bout ζ'-axis through an angle ϕ_3, results in the $\xi\eta\zeta$ coordinate system. The angles ϕ_1, ϕ_2 and ϕ_3, which are the Bryant angles, completely specify the orientation of the $\xi\eta\zeta$ frame relative to the *xyz* coordinate system. Figure 4.3 shows the various steps of the sequence of rotations associated with Bryant angles.

Similarly to the Euler angles, when using Bryant angles, the elements of the complete rotational transformation matrix **A** can be obtained as the triple product of the matrices that define the elemental planar rotations, i.e., the matrices (Nikravesh 1988)

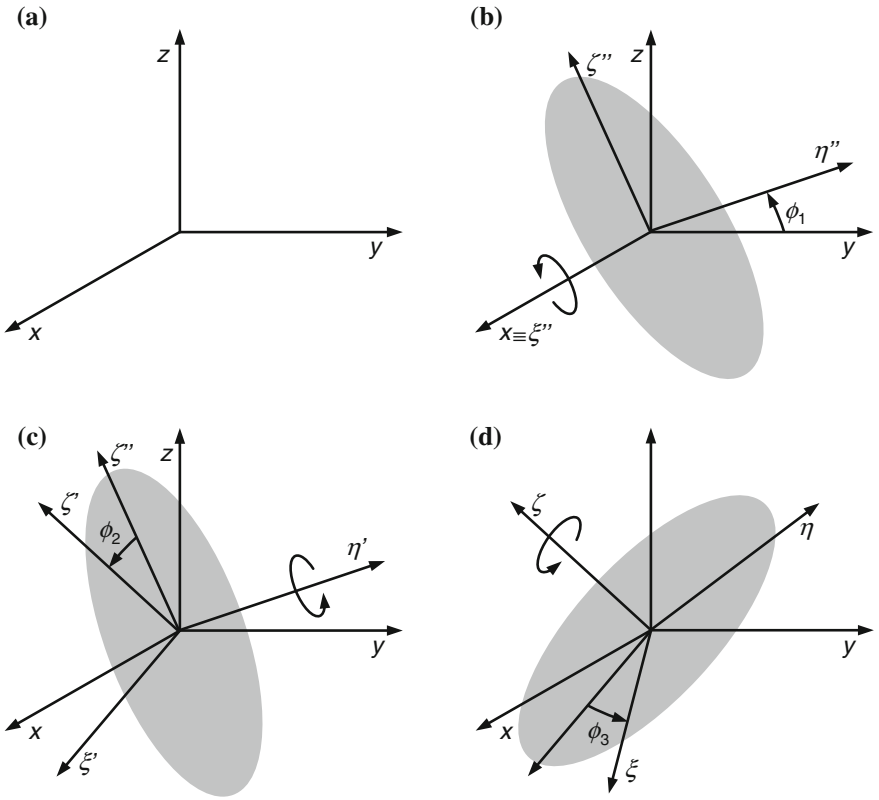

Fig. 4.3 Different stages of rotation for Bryant angles. **a** Initial global system coordinate; **b** First rotation; **c** Second rotation; **d** Third rotation

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}\phi_1 & -\mathrm{s}\phi_1 \\ 0 & \mathrm{s}\phi_1 & \mathrm{c}\phi_1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathrm{c}\phi_2 & 0 & \mathrm{s}\phi_2 \\ 0 & 1 & 0 \\ -\mathrm{s}\phi_2 & 0 & \mathrm{c}\phi_2 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathrm{c}\phi_3 & -\mathrm{s}\phi_3 & 0 \\ \mathrm{s}\phi_3 & \mathrm{c}\phi_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.4}$$

Hence, the complete transformation matrix, $\mathbf{A} = \mathbf{DCB}$, is given as

$$\mathbf{A} = \begin{bmatrix} \mathrm{c}\phi_2\mathrm{c}\phi_3 & -\mathrm{c}\phi_2\mathrm{s}\phi_3 & \mathrm{s}\phi_2 \\ \mathrm{c}\phi_1\mathrm{s}\phi_3 + \mathrm{s}\phi_1\mathrm{s}\phi_2\mathrm{c}\phi_3 & \mathrm{c}\phi_1\mathrm{c}\phi_3 - \mathrm{s}\phi_1\mathrm{s}\phi_2\mathrm{s}\phi_3 & -s\phi_1\mathrm{c}\phi_2 \\ \mathrm{s}\phi_1\mathrm{s}\phi_3 - \mathrm{c}\phi_1\mathrm{s}\phi_2\mathrm{c}\phi_3 & \mathrm{s}\phi_1\mathrm{c}\phi_3 + \mathrm{c}\phi_1\mathrm{s}\phi_2\mathrm{s}\phi_3 & \mathrm{c}\phi_1\mathrm{c}\phi_2 \end{bmatrix} \tag{4.5}$$

In a similar manner as in the *zxz* convention, the transformation matrix associated with the Bryant angles is highly nonlinear in terms of the three angles. In addition, if the three Bryant angles are not chosen correctly, the rotated frame will not coincide with the $\xi\eta\zeta$ desired frame. It can be observed that a singularity exists when $\cos\,\phi_2 = 0$, which means that the first and third rotations coincide and it is equivalent to rotate about the original *x*-axis twice. Therefore, the first and third angles can be combined as one single rotation. This fact can be illustrated by setting $\cos\,\phi_2 = 0$ in Eq. (4.5) to obtain (Nikravesh 1988)

$$\mathbf{A}_i = \begin{bmatrix} 0 & 0 & 1 \\ \mathrm{s}(\phi_1 + \phi_3) & \mathrm{c}(\phi_1 + \phi_3) & 0 \\ -\mathrm{c}(\phi_1 + \phi_3) & \mathrm{s}(\phi_1 + \phi_3) & 0 \end{bmatrix} \tag{4.6}$$

It can be drawn that the singularity phenomenon is an inherent problem associated with any three rotational coordinates regardless of their forms or convention. However, this problem can be avoided if four rotational coordinates are utilized, namely the Euler parameters (Schiehlen 1990).

Thus, according to Euler's theorem on finite rotation, a rotation in the three-dimensional space can always be described by a rotation along a certain axis over a certain angle. In other words, the Euler's theorem states that the general displacement of a body with a point fixed is a rotation about some axis. The theorem indicates that the orientation of the body-fixed axes at any instant can be obtained by an imaginary rotation of these axes from an orientation coincident with the global axes. Thus according to the Euler's theorem, there exists a unique axis that if the *xyz* frame is rotated about it by an angle ϕ it becomes parallel to the $\xi\eta\zeta$ frame, or vice versa. This imaginary axis is denoted by $\vec{u}$ and it is called the orientational axis of rotation, as Fig. 4.4 depicts (Nikravesh 1988). Thus, a set of rotational coordinates can be defined as

$$e_0 = \cos\frac{\phi}{2} \tag{4.7}$$

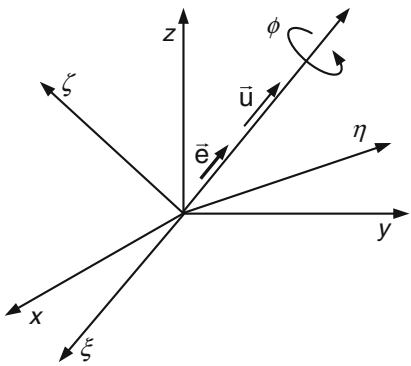

Fig. 4.4 Representation of the Euler parameters

$$\mathbf{e} = \{ e_1 \quad e_2 \quad e_3 \}^T = \mathbf{u} \sin \frac{\phi}{2} \tag{4.8}$$

in which vector $\vec{e}$ is defined along the orientational axis of rotation and has a magnitude of sin $\phi/2$. The four quantities e_0, e_1, e_2 and e_3 are the so-called Euler parameters. Analyzing Eqs. (4.7) and (4.8) it can be observed that the Euler parameters are not independent. Taking advantage of the fundamental equation of trigonometry, together with Eqs. (4.7) and (4.8) yields (Nikravesh 1988)

$$\cos^2 \frac{\phi}{2} + \mathbf{u}^T \mathbf{u} \sin^2 \frac{\phi}{2} = 1 \tag{4.9}$$

or, alternatively,

$$e_0^2 + \mathbf{e}^T \mathbf{e} = e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1 \tag{4.10}$$

Usually, the Euler parameters are expressed in the following form

$$\mathbf{p} = \begin{Bmatrix} e_0 \\ \mathbf{e} \end{Bmatrix} = \begin{Bmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{Bmatrix} \tag{4.11}$$

Then, Eq. (4.10) can be rewritten as

$$\mathbf{p}^T \mathbf{p} = 1 \tag{4.12}$$

According to Euler's theorem, any vector lying along the orientational axis of rotation must have the same components in both initial and final coordinate systems. The transformation matrix can be expressed in terms of the Euler parameters as (Nikravesh 1988)

$$\mathbf{A} = 2\begin{bmatrix} e_0^2 + e_1^2 - \frac{1}{2} & e_1e_2 - e_0e_3 & e_1e_3 + e_0e_2 \\ e_1e_2 + e_0e_3 & e_0^2 + e_2^2 - \frac{1}{2} & e_2e_3 - e_0e_1 \\ e_1e_3 - e_0e_2 & e_2e_3 + e_0e_1 & e_0^2 + e_3^2 - \frac{1}{2} \end{bmatrix} \tag{4.13}$$

In a compact form, Eq. (4.13) can be written as

$$\mathbf{A}^T = (2e_0^2 - 1)\mathbf{I} + 2(\mathbf{e}\mathbf{e}^T + e_0\tilde{\mathbf{e}}) \tag{4.14}$$

where $\mathbf{I}$ is the 3×3 identity matrix and $\tilde{\mathbf{e}}$ is the skew-symmetric matrix associated with the vector $\mathbf{e}$ and is given by

$$\tilde{\mathbf{e}} = \begin{bmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{bmatrix} \tag{4.15}$$

In most practical problems, the choice of how to embed a body-fixed coordinate system in a body is open. The $\xi\eta\zeta$ axes may be embedded in a body according to any of the configurations illustrated in Fig. 4.5. If the $\xi\eta\zeta$ axes are parallel to xyz axes, as shown in Fig. 4.5a, then

$$\mathbf{p} = \{1 \quad 0 \quad 0 \quad 0\}^T, \quad \xi\eta\zeta \,||\, xyz \tag{4.16}$$

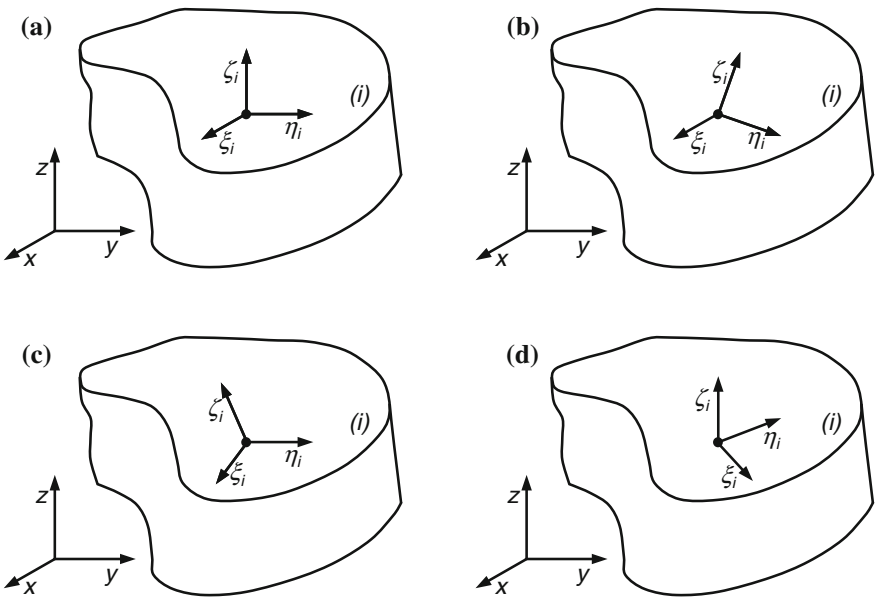

Fig. 4.5 Special cases of the orientation of body-fixed frame that simplify the determination of Euler parameters: **a** $\xi\eta\zeta \parallel xyz$; **b** $\xi \parallel x$; **c** $\eta \parallel y$; **d** $\zeta \parallel z$

In this case, any axis can be the orientational axis of rotation and the angle of rotation is zero. If the ξ-axis is parallel to the x-axis and the angle of rotation is ϕ, as shown in Fig. 4.5b, the Euler parameters are expressed as

$$\mathbf{p} = \left\{ \cos\frac{\phi}{2} \quad \sin\frac{\phi}{2} \quad 0 \quad 0 \right\}^T, \quad \xi \,||\, x \tag{4.17}$$

In this case, the x-axis is the orientational axis of rotation. In a similar manner, for orientations shown in Fig. 4.5c, d the Euler parameters can be written as follows, respectively

$$\mathbf{p} = \left\{ \cos\frac{\phi}{2} \quad 0 \quad \sin\frac{\phi}{2} \quad 0 \right\}^T, \quad \eta \,||\, y \tag{4.18}$$

$$\mathbf{p} = \left\{ \cos\frac{\phi}{2} \quad 0 \quad 0 \quad \sin\frac{\phi}{2} \right\}^T, \quad \zeta \,||\, z \tag{4.19}$$

In these special cases, it is relatively simple to determine the angle of rotation and then to calculate the Euler parameters (Nikravesh 1988).

References

Goldstein H (1980) Classical mechanics, 2nd edn. Addison-Wesley, Reading
Landau LD, Lifschitz EM (1976) Mechanics, 3rd edn. Pergamon Press, Oxford
Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs, New Jersey
Schiehlen W (1990) Multibody systems handbook. Springer, Berlin
Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 5
Angular Velocity and Acceleration

Abstract In this chapter, a complete characterization of the angular velocity and angular acceleration for rigid bodies in spatial multibody systems are presented. For both cases, local and global formulations are described taking into account the advantages of using Euler parameters. In this process, the transformation between global and local components of the angular velocity and time derivative of the Euler parameters are analyzed and discussed in this chapter.

Keywords Angular velocity · Angular acceleration · Spatial motion

In order to keep the present analysis simple, let concentrate on the rotation of a body and neglect its translational motion. For this desideratum, let consider that the $\xi\eta\zeta$ coordinate system is rotating and has its origin coincident with the origin of the nonrotating xyz coordinate system, as shown in Fig. 5.1. The angular velocity $\vec{\omega}$ describes the axis and the magnitude of the rotation of the $\xi\eta\zeta$ frame. This axis is called the instantaneous axis of rotation and should not be mistaken with the orientational axis of rotation. Thus, at this instant, if the rotation of the body is frozen, the axis around which the body must rotate in order for the two coordinate systems become parallel is the orientational axis of rotation (Shabana 1989; Schiehlen 1990).

The angular velocity vector can be projected onto either the $\xi\eta\zeta$ frame or xyz frame resulting into algebraic vectors expressed as

$$\boldsymbol{\omega}' = \{\, \omega_\xi \quad \omega_\eta \quad \omega_\zeta \,\}^T, \quad \boldsymbol{\omega} = \{\, \omega_x \quad \omega_y \quad \omega_z \,\}^T \tag{5.1}$$

Nikravesh (1988) demonstrated that the angular velocity and the time derivative of the transformation matrix $\mathbf{A}$ have the following relations

$$\dot{\mathbf{A}} = \mathbf{A}\tilde{\boldsymbol{\omega}}', \quad \dot{\mathbf{A}} = \tilde{\boldsymbol{\omega}}\mathbf{A} \tag{5.2}$$

or alternatively,

$$\mathbf{A}^T\dot{\mathbf{A}} = \tilde{\boldsymbol{\omega}}', \quad \dot{\mathbf{A}}\mathbf{A}^T = \tilde{\boldsymbol{\omega}} \tag{5.3}$$

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_5

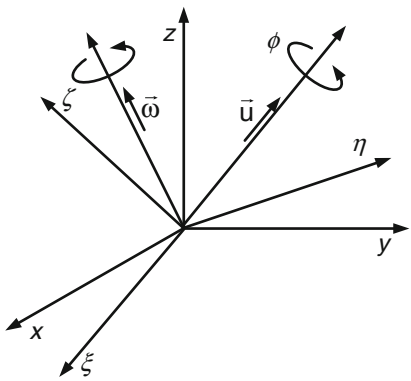

Fig. 5.1 Rotating $\xi\eta\zeta$ coordinates system

It should be noted that the angular velocity vector does not have an integral, i.e., there is no array of three rotational coordinates that its first time derivative is defined as the vector of angular velocity.

It is known that the global position of a point P that is fixed in the $\xi\eta\zeta$ coordinate system is given by the equation

$$\mathbf{s}^P = \mathbf{A}\mathbf{s}'^P \tag{5.4}$$

Differentiating this equation with respect to time yields

$$\dot{\mathbf{s}}^P = \dot{\mathbf{A}}\mathbf{s}'^P \tag{5.5}$$

Substituting Eq. (5.2) into Eq. (5.5) results in

$$\dot{\mathbf{s}}^P = \tilde{\boldsymbol{\omega}}\mathbf{A}\mathbf{s}'^P = \tilde{\boldsymbol{\omega}}\mathbf{s}^P \tag{5.6}$$

Thus, for any vector $\vec{s}$ attached to the $\xi\eta\zeta$ coordinate system, such as the one in Fig. 3.1, Eq. (5.6) can be written as (Nikravesh 1988)

$$\dot{\mathbf{s}} = \tilde{\boldsymbol{\omega}}\mathbf{s} = -\tilde{\mathbf{s}}\tilde{\boldsymbol{\omega}} \tag{5.7}$$

For a $\xi\eta\zeta$ frame that rotates and translates relative to the nonmoving xyz frame, the velocity of a point P_i that is fixed in the $\xi\eta\zeta$ frame can be determined. Thus, a point P_i can be located in the xyz frame by the relation

$$\mathbf{r}^P = \mathbf{r} + \mathbf{s}^P \tag{5.8}$$

The time derivative of this equation gives the velocity of point P as

$$\dot{\mathbf{r}}^P = \dot{\mathbf{r}} + \dot{\mathbf{s}}^P = \dot{\mathbf{r}} + \tilde{\boldsymbol{\omega}}\mathbf{s}^P \tag{5.9}$$

The transformation between the xyz components of the angular velocity vector and time derivative of Euler parameters is given by (Nikravesh 1988)

$$\boldsymbol{\omega} = 2\mathbf{G}\dot{\mathbf{p}} \tag{5.10}$$

In expanded form, Eq. (5.10) is

$$\begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix} = 2 \begin{bmatrix} -e_1 & e_0 & -e_3 & e_2 \\ -e_2 & e_3 & e_0 & -e_1 \\ -e_3 & -e_2 & e_1 & e_0 \end{bmatrix} \begin{Bmatrix} \dot{e}_0 \\ \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{Bmatrix} \tag{5.11}$$

The inverse transformation is found to be

$$\dot{\mathbf{p}} = \frac{1}{2}\mathbf{G}^T\boldsymbol{\omega} \tag{5.12}$$

In expanded form, Eq. (5.12) is

$$\begin{Bmatrix} \dot{e}_0 \\ \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{Bmatrix} = \frac{1}{2} \begin{bmatrix} -e_1 & -e_2 & -e_3 \\ e_0 & e_3 & -e_2 \\ -e_3 & e_0 & e_1 \\ e_2 & -e_1 & e_0 \end{bmatrix} \begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix} \tag{5.13}$$

The transformation between the $\xi\eta\zeta$ components of the angular velocity vector and the time derivative of Euler parameters is given by (Nikravesh 1988)

$$\boldsymbol{\omega}' = 2\mathbf{L}\dot{\mathbf{p}} \tag{5.14}$$

In expanded form, Eq. (5.14) is

$$\begin{Bmatrix} \omega_\xi \\ \omega_\eta \\ \omega_\zeta \end{Bmatrix} = 2 \begin{bmatrix} -e_1 & e_0 & e_3 & -e_2 \\ -e_2 & -e_3 & e_0 & e_1 \\ -e_3 & e_2 & -e_1 & e_0 \end{bmatrix} \begin{Bmatrix} \dot{e}_0 \\ \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{Bmatrix} \tag{5.15}$$

The inverse transformation is found to be

$$\dot{\mathbf{p}} = \frac{1}{2}\mathbf{L}^T\boldsymbol{\omega}' \tag{5.16}$$

In expanded form, Eq. (5.16) is

$$\begin{Bmatrix} \dot{e}_0 \\ \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{Bmatrix} = \frac{1}{2} \begin{bmatrix} -e_1 & -e_2 & -e_3 \\ e_0 & -e_3 & e_2 \\ e_3 & e_0 & -e_1 \\ -e_2 & e_1 & e_0 \end{bmatrix} \begin{Bmatrix} \omega_\xi \\ \omega_\eta \\ \omega_\zeta \end{Bmatrix} \tag{5.17}$$

Differentiating Eq. (5.10) with respect to time yields

$$\dot{\boldsymbol{\omega}} = 2\mathbf{G}\ddot{\mathbf{p}} + 2\dot{\mathbf{G}}\dot{\mathbf{p}} \tag{5.18}$$

Nikravesh (1988) showed that the product $\dot{\mathbf{G}}\dot{\mathbf{p}}$ is null, and hence, Eq. (5.17) can be simplified as

$$\dot{\boldsymbol{\omega}} = 2\mathbf{G}\ddot{\mathbf{p}} \tag{5.19}$$

In a similar manner, differentiating Eq. (5.14) with respect to time yields

$$\dot{\boldsymbol{\omega}}' = 2\mathbf{L}\ddot{\mathbf{p}} \tag{5.20}$$

Vectors $\dot{\boldsymbol{\omega}}$ and $\dot{\boldsymbol{\omega}}'$ are the global and local components of vector $\vec{\dot{\omega}}$ defined as the angular acceleration of the $\xi\eta\zeta$ frame. Finally, it can be shown that the inverses of Eqs. (5.19) and (5.20) are given by Nikravesh (1988)

$$\ddot{\mathbf{p}} = \frac{1}{2}\mathbf{G}^T\dot{\boldsymbol{\omega}} - \frac{1}{4}(\boldsymbol{\omega}^T\boldsymbol{\omega})\mathbf{p} \tag{5.21}$$

and

$$\ddot{\mathbf{p}} = \frac{1}{2}\mathbf{L}^T\dot{\boldsymbol{\omega}}' - \frac{1}{4}(\boldsymbol{\omega}'^T\boldsymbol{\omega}')\mathbf{p} \tag{5.22}$$

It is clear that $\boldsymbol{\omega}^T\boldsymbol{\omega} = \boldsymbol{\omega}'^T\boldsymbol{\omega}' = \omega^2$, where ω is the magnitude of $\vec{\omega}$.

References

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Schiehlen W (1990) Multibody systems handbook. Springer, Berlin

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 6
Vector of Coordinates, Velocities and Accelerations

Abstract This chapter describes the how the vector of coordinates are defined in the formulation of spatial multibody systems. For this purpose, the translational motion is described in terms of Cartesian coordinates, while rotational motion is specified using the technique of Euler parameters. This approach avoids the computational difficulties associated with the singularities in the case of using Euler angles or Bryant angles. Moreover, the formulation of the velocities vector and accelerations vector is presented and analyzed here. These two sets of vectors are defined in terms of translational and rotational coordinates.

Keywords Positions · Velocities · Accelerations

The configuration of a body free in the space is uniquely defined by six coordinates, whereas three coordinates are needed to specify the position of the body and additional three coordinates are required to describe its angular orientation. Throughout this work, translation is described in terms of Cartesian coordinates, while rotation is specified using the technique of Euler parameters. In general, the angular orientation of a body can be described with an arbitrary set of rotational coordinates, such as Euler angles or Bryant angles. However, since these alternative formulations may yield to singularities for critical configurations, Euler parameters are selected for formulating the constraint equations and the equations of motion presented later on in this work (Shabana 1989; Schiehlen 1990).

Thus, the position vector of a body i can be defined as

$$\mathbf{q}_i = \begin{Bmatrix} \mathbf{r}_i \\ \mathbf{p}_i \end{Bmatrix} \tag{6.1}$$

in which vector $\mathbf{r}_i$ contains the three translational coordinates of body i defined with respect to the global coordinates system as

$$\mathbf{r}_i = \begin{Bmatrix} x_i \\ y_i \\ z_i \end{Bmatrix} \tag{6.2}$$

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_6

In turn, the vector $\mathbf{p}_i$ that includes the four Euler parameters can be written in the following form

$$\mathbf{p}_i = \begin{Bmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{Bmatrix} \tag{6.3}$$

For simplifying the formulation at the velocity and acceleration levels, the concept of angular velocity is employed, i.e., velocity vector specifying the translational and rotational velocity of a body i does not correspond to the time derivative of Eq. (6.1), but it is defined as (Jalón and Bayo 1994)

$$\mathbf{v}_i = \begin{Bmatrix} \dot{\mathbf{r}}_i \\ \boldsymbol{\omega}_i \end{Bmatrix} \tag{6.4}$$

where the vector of angular velocities is written as

$$\boldsymbol{\omega}_i = \begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix}_i \tag{6.5}$$

Accordingly, acceleration vector specifying the translational and rotational accelerations of a body i is defined as

$$\dot{\mathbf{v}}_i = \begin{Bmatrix} \ddot{\mathbf{r}}_i \\ \dot{\boldsymbol{\omega}}_i \end{Bmatrix} \tag{6.6}$$

which corresponds to the time derivative of Eq. (6.4).

It must that be highlighted that alternative arrays for velocity and acceleration equations can be defined as (Nikravesh 1988)

$$\dot{\mathbf{q}}_i = \begin{Bmatrix} \dot{\mathbf{r}}_i \\ \dot{\mathbf{p}}_i \end{Bmatrix}, \quad \mathbf{h}_i = \begin{Bmatrix} \dot{\mathbf{r}}_i \\ \boldsymbol{\omega}'_i \end{Bmatrix} \tag{6.7}$$

and

$$\ddot{\mathbf{q}}_i = \begin{Bmatrix} \ddot{\mathbf{r}}_i \\ \ddot{\mathbf{p}}_i \end{Bmatrix}, \quad \dot{\mathbf{h}}_i = \begin{Bmatrix} \ddot{\mathbf{r}}_i \\ \dot{\boldsymbol{\omega}}'_i \end{Bmatrix} \tag{6.8}$$

References

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Schiehlen W (1990) Multibody systems handbook. Springer, Berlin

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 7
Kinematic Constraint Equations

Abstract This chapter presents a general methodology for the formulation of the kinematic constraint equations at position, velocity and acceleration levels. Also a brief characterization of the different type of constraints is offered, namely the holonomic and nonholonomic constraints. The kinematic constraints described here are formulated using generalized coordinates. The chapter ends with a general approach to deal with the kinematic analysis of multibody systems.

Keywords Kinematic constraints · Positions · Velocities · Accelerations

A constraint condition implies a restriction in the kinematical degrees of freedom of one or more bodies. The classical constraint is usually an algebraic equation that defines the relative translation or rotation between two bodies. There are furthermore possibilities to constrain the relative velocity between two bodies or a body and the ground. This is for example the case of a rolling disc, where the point of the disc that contacts the ground has always zero relative velocity with respect to the ground. In the case that the velocity constraint condition cannot be integrated in time in order to form a position constraint, it is called nonholonomic. This is the case for the general rolling constraint. In addition to that there are non-classical constraints that might even introduce a new unknown coordinate, such as a sliding joint, where a point of a body is allowed to move along the surface of another body. In the case of contact, the constraint condition is based on inequalities and, therefore, such a constraint does not permanently restrict the degrees of freedom of bodies (Huston 1990).

As it was presented previously, the configuration of a multibody system is described by a set of variables called generalized coordinates that completely define the location and orientation of each body in the system. Hereafter, the set of generalized coordinates of a multibody system will be denoted by vector $\mathbf{q} = \{\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \ldots, \mathbf{q}_n\}^T$, where n is the number of coordinates. Mostly, in multibody systems formulation the generalized coordinates can be divided into independent and dependent variables, consequently, several algebraic equations are needed to be introduced to relate them. In other words, the constraint equations represent the kinematic relation between independent and dependent coordinates. In a simple

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_7

manner, the constraint equations can arise from the description of the system topology and from the characterization of the driving and guiding constraints that are used to guide the system through the analysis. In this work, the set of constraint equations is denoted by symbol $\mathbf{\Phi}$. In order to distinguish among the different constraint equations, each elementary set of constraints is identified by a superscript containing two parameters. The first parameter denotes the type of constraint, while the second one defines the number of independent equations that it involves. For example, $\mathbf{\Phi}^{(s,3)}$ refers to a spherical (s) joint constraint, which contains three (3) equations.

Kinematic constraints can be classified as holonomic or nonholonomic. Holonomic constraints arise from geometric constraints and are integrable into a form involving only coordinates (*holo* comes from Greek that means whole, integer). Nonholonomic constraints are not integrable. The relation specified by a constraint can be an explicit function of time designated as rheonomic constraints (*rheo* comes from Greek that means hard, inflexible, independent) or not, being designated by scleronomic constraints (*scleros* comes from Greek that means flexible, changing). Figure 7.1 shows a typical spherical joint and a simple human body model placed on a spherical surface, which represents a holonomic and a nonholonomic constraint, respectively. Thus, for instance, in the motion of the human model on the spherical surface, the following mathematical relation has to be satisfied during the analysis (Flores 2006)

$$\mathbf{r}^T\mathbf{r} - R^2 \geq 0 \tag{7.1}$$

where R is the radius of the spherical surface and vector $\mathbf{r}$ represents the position of the model measured from the center of the spherical surface.

The kinematic constraints considered here are assumed to be holonomic, arising from geometrical constraints on the generalized coordinates. Holonomic constraints, also called geometric restrictions, are algebraic equations imposed to the system that are expressed as functions of the displacement and, possibly, time. If the time t does not appear explicitly in the constraint equation, then the system is said to be scleronomic. A simple example of scleronomic constraint equation is the revolute joint between two bodies. Otherwise, when the constraint is holonomic and t appears explicitly, the system is said to be rheonomic (Shabana 1989).

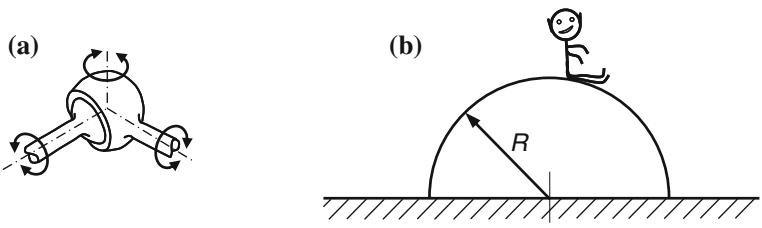

Fig. 7.1 **a** Holonomic constraint; **b** nonholonomic constraint

In general manner, the relations that describe the constraints imposed by kinematic pairs, such as mechanical joints, are formulated using algebraic equations. The kinematic constraint equations can be written as

$$\mathbf{\Phi} \equiv \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \tag{7.2}$$

where $\mathbf{q}$ denotes the vector of body-coordinates defined by Eq. (6.1) and $\mathbf{\Phi}$ represents a function describing the kinematic constraints. In general, constraint equations may also be functions of time, however, algebraic equality constraints as expressed by Eq. (7.2) are referred to as holonomic constraints.

The first time derivative of Eq. (7.2) yields the velocity constraints that provide relations between the velocity variables of a system. The velocity constraints can be expressed as

$$\dot{\mathbf{\Phi}} \equiv \mathbf{D}\mathbf{v} = \mathbf{0} \tag{7.3}$$

where $\mathbf{D}$ denotes the Jacobian matrix and $\mathbf{v}$ contains the velocity terms defined by Eq. (6.4). For driving elements, the corresponding velocity constraint equations can be written in the form

$$\dot{\mathbf{\Phi}} \equiv \mathbf{D}\mathbf{v} = \upsilon \tag{7.4}$$

in which the right-hand side contains the partial derivates of $\mathbf{\Phi}$ with respect to time, $\partial\mathbf{\Phi}/\partial t$. The constraints at the velocity level are represented by linear algebraic equations.

The second time derivative of Eq. (7.2) results in

$$\ddot{\mathbf{\Phi}} \equiv \mathbf{D}\dot{\mathbf{v}} + \dot{\mathbf{D}}\mathbf{v} = \mathbf{0} \tag{7.5}$$

where $\dot{\mathbf{v}}$ denotes the acceleration terms defined by Eq. (6.6) and the term $-\dot{\mathbf{D}}\mathbf{v}$ is referred to as the right-hand side of the kinematic acceleration equations. By introducing $\boldsymbol{\gamma} = -\dot{\mathbf{D}}\mathbf{v}$, Eq. (7.4) can be rewritten as

$$\mathbf{D}\dot{\mathbf{v}} = \boldsymbol{\gamma} \tag{7.6}$$

In should be highlighted that the terms involved in Eqs. (7.2) through (7.6) appear in a general form, that is, they do not reflect the type of coordinates considered. In addition, the constraint equations represented by Eq. (7.2) are non–linear in terms of $\mathbf{q}$ and are, usually, solved by employing the Newton-Raphson method. Equations (7.3) and (7.6) are linear in terms of $\mathbf{v}$ and $\dot{\mathbf{v}}$, respectively, and can be solved by any usual method adopted for the solution of systems of linear equations. It should be noted that the issues related to the treatment of redundant constraints are not presented in this work. The interested reader in the details on this particular topic is referred to the work by Wehage and Haug (1982).

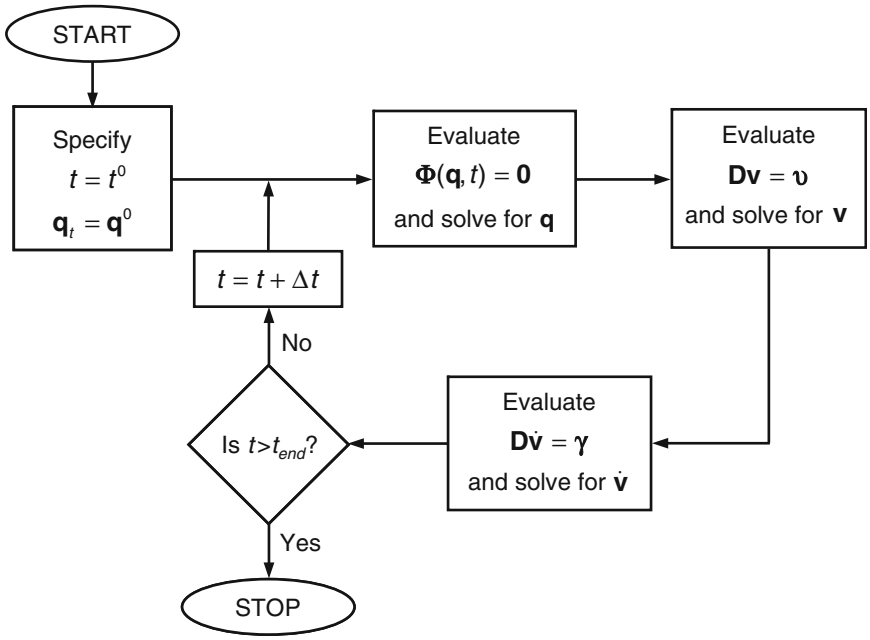

Fig. 7.2 Flowchart of computational procedure for kinematic analysis of a MBS

The kinematic analysis is the study of the motion of a multibody system, independently of the causes that produce it. Since in the kinematic analysis the forces are not considered, the motion of the system is specified by driving or guiding elements that govern the motion of specific degrees of freedom of the system during the analysis. The position, velocity and acceleration of the remaining elements of the system are defined by kinematic constraint equations that describe the system topology. It is clear that in the kinematic analysis, the number of driving and guiding constraints must be equal to the number of degrees of freedom of the multibody system. In short, the kinematic analysis is performed by solving a set of equations that result from the kinematic, driving and guiding constraints (Jalón and Bayo 1994).

The kinematic analysis of a multibody system can be carried out by solving the set of Eqs. (7.2)–(7.6) together with the necessary driver constraints corresponding to the free degrees of freedom. Therefore, the necessary steps to perform this type of analysis, sketched in Fig. 7.2, are summarized as:

1. Specify initial conditions for positions $\mathbf{q}^0$ and initialize the time t^0.
2. Evaluate the position constraint Eq. (7.2) and solve them for positions, $\mathbf{q}$.
3. Evaluate the velocity constraint Eq. (7.4) and solve them for velocities, $\mathbf{v}$.
4. Evaluate the acceleration constraint Eq. (7.6) and solve them for accelerations, $\dot{\mathbf{v}}$.
5. Increment the time. If the time is smaller than final time, go to step (2), otherwise stop the kinematic analysis.

A close observation of the Eqs. (7.4) and (7.6) shows that both expressions represent systems of linear equations, with the same leading matrix and different right-hand side vectors. Moreover, since both expressions share the same leading matrix, Jacobian matrix of the constraints, evaluated with the latest calculated configuration of the system, then this matrix only needs to be factorized once during each step (Nikravesh 1988).

References

Flores P (2006) Modelação de restrições cinemáticas em sistemas de corpos múltiplos. School of Engineering, University of Minho, Guimarães

Huston RL (1990) Multibody dynamics. Butterworth-Heinemann, Boston

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Wehage RA, Haug EJ (1982) Generalized coordinate partitioning for dimension reduction in analysis of constrained systems. J Mech Des 104:247–255

Chapter 8
Basic Constraints Between Two Vectors

Abstract This chapter deals with the characterization of the basic constraints between two vectors. This issue plays a crucial role in the formulation of constraint equations for mechanical joints. In particular, relations between two parallel and two perpendicular vectors are derived. Moreover, formulation for a vector that connects two generic points is presented. The material described here is developed under the framework of multibody systems formulation for spatial systems.

Keywords Constraints · Parallel vectors · Perpendicular vectors

The main purpose of this section is to provide the fundamental concepts that are necessary to formulate the constraint equations associated with kinematic pairs. For instance, some kinematic constraints require that two vectors remain parallel or perpendicular. On the other hand, it is frequently necessary to define one or more points on bodies. Taking into consideration that points P_i and Q_i represented in Fig. 8.1 are located on body i by their local coordinates, then the corresponding global coordinates can be determined at any given instant by the following expressions (Nikravesh 1988; Schiehlen 1990; Flores et al. 2008)

$$\mathbf{r}_i^P = \mathbf{r}_i + \mathbf{A}_i \mathbf{s}_i'^P \tag{8.1}$$

$$\mathbf{r}_i^Q = \mathbf{r}_i + \mathbf{A}_i \mathbf{s}_i'^Q \tag{8.2}$$

Considering now that points P_i and Q_i are connected by a vector $\mathbf{s}_i$, as Fig. 8.1 shows, then this vector attached on body i can be computed as

$$\mathbf{s}_i = \mathbf{s}_i^P - \mathbf{s}_i^Q = \mathbf{A}_i \mathbf{s}_i'^P - \mathbf{A}_i \mathbf{s}_i'^Q = \mathbf{A}_i (\mathbf{s}_i'^P - \mathbf{s}_i'^Q) \tag{8.3}$$

It should be noted that vector $\mathbf{s}_i$ is only function of the rotational matrix of the body i. This vector has a constant magnitude in any system.

Figure 8.2 depicts a vector that links two points P_i and P_j located on different bodies. Knowing the local components of points P_i and P_j, then the corresponding global coordinates of vector $\mathbf{d}$ can be evaluated as

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_8

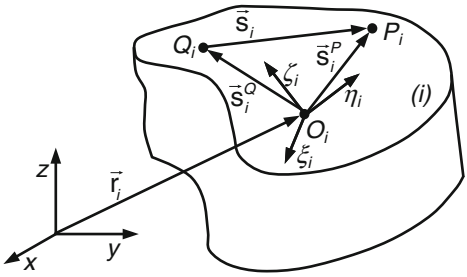

Fig. 8.1 Vector connecting two points P_i and Q_i located on body i

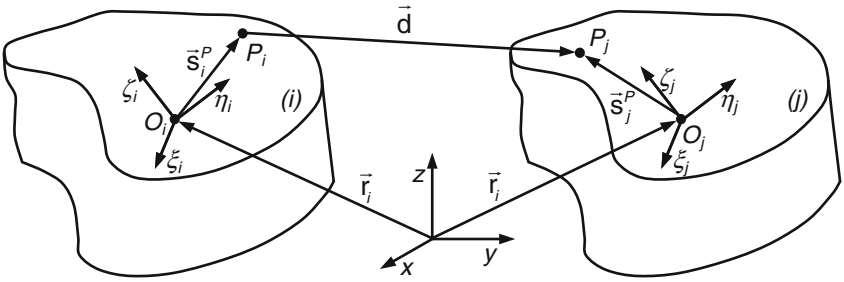

Fig. 8.2 Vector connecting two points P_i and P_j located on different bodies

$$\mathbf{d} = \mathbf{r}_j^P - \mathbf{r}_i^P = \mathbf{r}_j + \mathbf{A}_j \mathbf{s}_j'^P - \mathbf{r}_i - \mathbf{A}_i \mathbf{s}_i'^P \tag{8.4}$$

which, in the most general case, can have a variable length.

Figure 8.3 shows two vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ attached to bodies i and j, respectively. In order to specify that these two vectors remain perpendicular at all times, it is necessary to define a constraint equation in the form

$$\Phi^{(n1,1)} \equiv \mathbf{s}_i^T \mathbf{s}_j = 0 \tag{8.5}$$

where the superscripts indicate that this is a normal type 1 constraint having 1 equation (Nikravesh 1988). If vector $\mathbf{d}$ in Fig. 8.4, which connects points P_i and P_j, is to remain perpendicular to $\mathbf{s}_i$, then it can be written that

$$\Phi^{(n2,1)} \equiv \mathbf{s}_i^T \mathbf{d} = 0 \tag{8.6}$$

which is a normal type 2 constraint having 1 equation (Nikravesh 1988).

For two vectors attached to different bodies to remain parallel, two constraint equations are required. The two constraint equations are derived by setting the cross product between them to zero. Yet, the cross product yields three algebraic equations, of which only two are independent. Thus, two of the equations can serve as

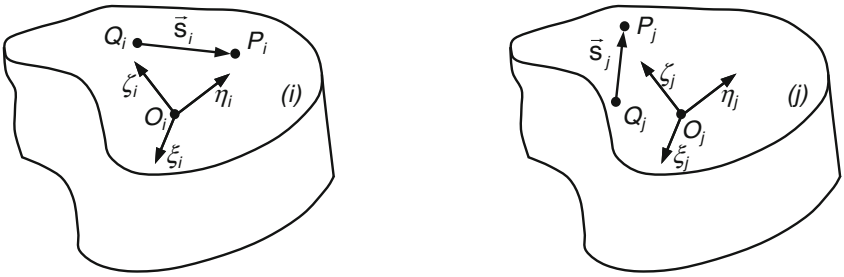

Fig. 8.3 Vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ located on bodies i and j that remain perpendicular at all instants of time

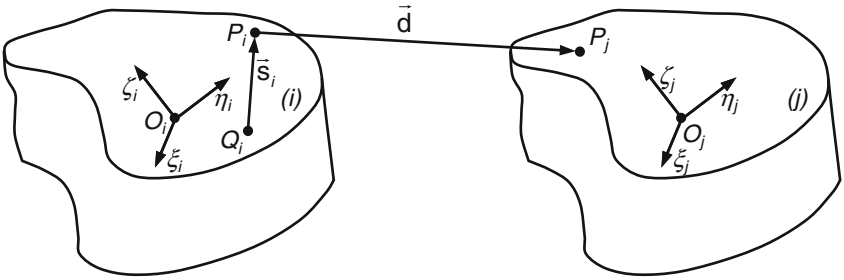

Fig. 8.4 Vector $\mathbf{s}_i$ located on body i that remains perpendicular to vector $\mathbf{d}$ connecting points P_i and P_j

the constraint equations. For two vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ that are embedded in corresponding bodies, as Fig. 8.5 illustrates, the constraint equations imposing parallelism (parallel type 1) are

$$\mathbf{\Phi}^{(p1,2)} \equiv \tilde{\mathbf{s}}_i \mathbf{s}_j = \mathbf{0} \tag{8.7}$$

In turn, the condition for two parallel vectors, one fixed to one body and the other connecting two bodies, as it is represented in Fig. 8.6, is written in the following form

$$\mathbf{\Phi}^{(p2,2)} \equiv \tilde{\mathbf{s}}_i \mathbf{d} = \mathbf{0} \tag{8.8}$$

It must be highlighted that Eqs. (8.7) and (8.8) provide three equations each. However, the sets of equations each have only two independent equations. There exists a critical case that is associated with selection of two equations from (8.7) and (8.8). This critical case occurs when two vectors become parallel to one of the global axes. To circumvent this issue, the dot product can be used twice. In first place, two perpendicular vectors to $\mathbf{s}_j$, and also to each other, are defined, as it is illustrated in the representation of Fig. 8.7. Then, the two constraint equations can be written in the following form (Nikravesh 1988)

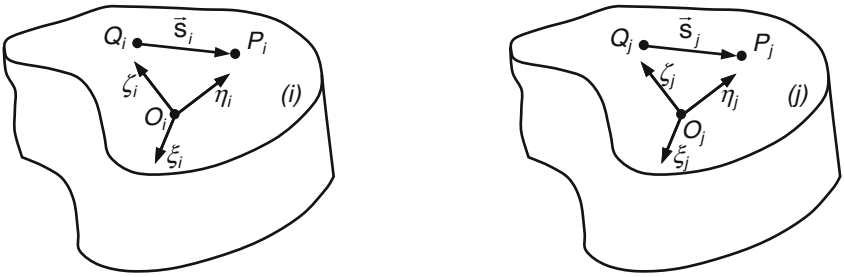

Fig. 8.5 Vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ located on bodies i and j that remain parallel

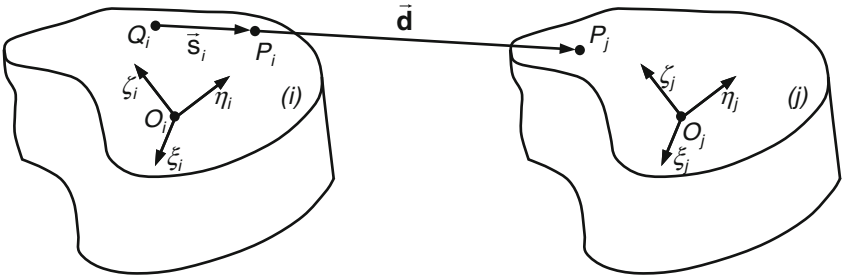

Fig. 8.6 Vector $\mathbf{s}_i$ located on body i that remains parallel to vector $\mathbf{d}$ connecting points P_i and P_j

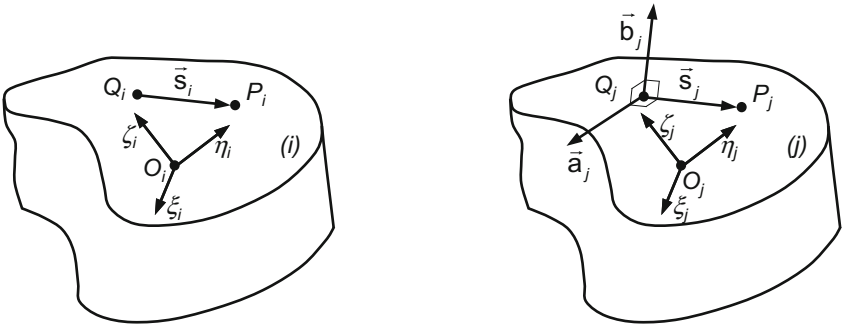

Fig. 8.7 Vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ located on bodies i and j that remain parallel

$$\mathbf{\Phi}^{(p1,2)} \equiv \begin{cases} \mathbf{s}_i{}^T \mathbf{a}_j = 0 \\ \mathbf{s}_i{}^T \mathbf{b}_j = 0 \end{cases} \tag{8.9}$$

Finally, it must stated that the constraints that keep two vectors perpendicular or parallel are the basic constraints between two vectors. These constraints remove relative degrees of freedom between adjacent bodies. For instance, the constraints

$n1$ and $n2$ each remove one DoF from a multibody system. In general, there is no need to use $p1$ and $p2$ constraints, therefore, $n1$ and $n2$ are utilized instead by defining some other appropriate vectors. These basic constraints will be used to construct the constraint equations for a variety of kinematic joints, as it will be seen in the next section. The form of the basic constraints remains the same regardless of the choice of rotational coordinates, in the measure that they only affect the process of evaluating the rotational transformation matrix **A** for each body (Shabana 1989; Schiehlen 1990; Jalón and Bayo 1994).

References

Flores P, Ambrósio J, Claro JCP, Lankarani HM (2008) Kinematics and dynamics of multibody systems with imperfect joints: models and case studies. Lecture notes in applied and computational mechanics, vol 34. Springer, Berlin

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Schiehlen W (1990) Multibody systems handbook. Springer, Berlin

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 9
Kinematic Joints Constraints

Abstract The kinematic joints constraints for several types of mechanical joints are derived here. Special attention is given to the spherical joint, revolute joint and spherical-spherical joint. In this process, the fundamental issues associated with kinematic constraints are developed, namely the right-hand side of the acceleration constraint equations and the contributions to the Jacobin matrix. The material presented in this chapter is developed under the framework of multibody systems formulation for spatial systems.

Keywords Kinematic constraints · Spherical joint · Revolute joint

The goal of this paragraph is to present the formulation of the most common types of kinematic constraint equations used to model spatial multibody systems. It is also objective to introduce their contributions to the Jacobian matrix of the constraints and to the right-hand side of the acceleration constraint equations (Schiehlen 1990; Jalón and Bayo 1994). The given presentation is restricted to those constraint formulations that are required to describe the kinematic joints utilized in the context of the present work, namely the ideal spherical joint, the revolute or rotational joint and the composite spherical-spherical joint. With only these three basic kinematic joints, a large class of spatial multibody systems can be studied. For details on the formulation of other types of kinematic joints the interested reader is referred to Nikravesh (1988)

An ideal or perfect spherical joint, also known as ball and socket joint, illustrated in Fig. 9.1, constrains the relative translations between two adjacent bodies i and j, allowing only three relative rotations. Therefore, the center of the spherical joint has constant coordinates with respect to any of the local coordinates systems of the connected bodies, i.e., a spherical joint is defined by the condition that the point P_i on body i coincides with the point P_j on body j. This condition is simply the spherical constraint, which can be written in a scalar form as (Flores et al. 2008)

$$\mathbf{\Phi}^{(s,3)} \equiv \mathbf{r}_j{}^P - \mathbf{r}_i{}^P = \mathbf{r}_j + \mathbf{s}_j{}^P - \mathbf{r}_i - \mathbf{s}_i{}^P = \mathbf{0} \tag{9.1}$$

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_9

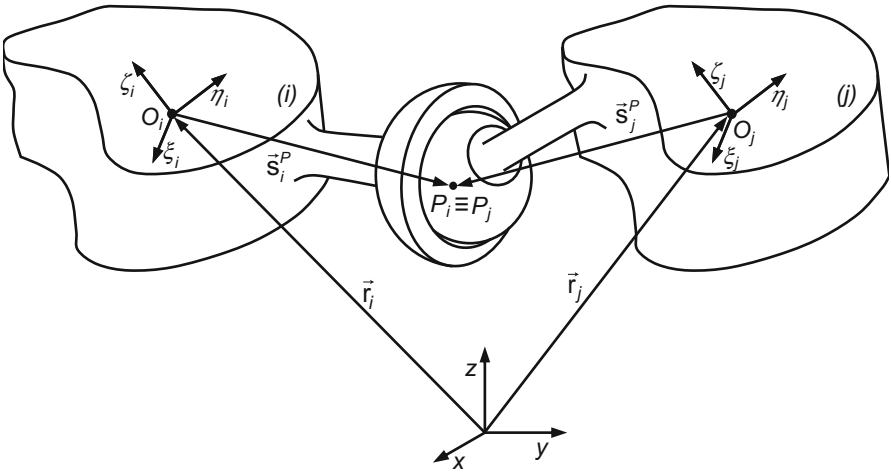

Fig. 9.1 Spherical joint connecting bodies i and j

The three scalar constraint equations implied by Eq. (9.1) restrict the relative position of points P_i and P_j. Therefore, three relative degrees of freedom are maintained between two bodies that are connected by a perfect spherical joint.

The first time derivative of Eq. (9.1) results in the velocity constraint equations for a spherical joint, which can be expressed as (Shabana 1989)

$$\dot{\mathbf{\Phi}}^{(s,3)} = \dot{\mathbf{r}}_j + \dot{\mathbf{s}}_j^P - \dot{\mathbf{r}}_i - \dot{\mathbf{s}}_i^P = \mathbf{0} \tag{9.2}$$

Taking advantage of Eq. (5.7), this equation can be rewritten as

$$\dot{\mathbf{\Phi}}^{(s,3)} = \dot{\mathbf{r}}_j - \tilde{\mathbf{s}}_j^P \boldsymbol{\omega}_j - \dot{\mathbf{r}}_i + \tilde{\mathbf{s}}_i^P \boldsymbol{\omega}_i = \mathbf{0} \tag{9.3}$$

or, alternatively, in the matrix form

$$\dot{\mathbf{\Phi}}^{(s,3)} = \begin{bmatrix} -\mathbf{I} & \tilde{\mathbf{s}}_i^P & \mathbf{I} & -\tilde{\mathbf{s}}_j^P \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{r}}_i \\ \boldsymbol{\omega}_i \\ \dot{\mathbf{r}}_j \\ \boldsymbol{\omega}_j \end{Bmatrix} = \mathbf{0} \tag{9.4}$$

Thus, by observing Eq. (9.4), it can be concluded that the contribution to the Jacobian matrix of the spherical joint constraints is given by

$$\mathbf{D}_{(3\times 12)}^{(s,3)} = \begin{bmatrix} -\mathbf{I} & \tilde{\mathbf{s}}_i^P & \mathbf{I} & -\tilde{\mathbf{s}}_j^P \end{bmatrix} \tag{9.5}$$

In a similar manner, the acceleration constraint equations of the spherical joint can be obtained by taking the time derivative of Eq. (9.3), which results in

$$\ddot{\vec{\Phi}}^{(s,3)} = \ddot{\mathbf{r}}_j - \overset{\sim}{\dot{\mathbf{s}}}_j^{\,P}\boldsymbol{\omega}_j - \tilde{\mathbf{s}}_j^{\,P}\dot{\boldsymbol{\omega}}_j - \ddot{\mathbf{r}}_i + \overset{\sim}{\dot{\mathbf{s}}}_i^{\,P}\boldsymbol{\omega}_i + \tilde{\mathbf{s}}_i^{\,P}\dot{\boldsymbol{\omega}}_i = \mathbf{0} \tag{9.6}$$

In the matrix form, Eq. (9.6) can be expressed as

$$\ddot{\boldsymbol{\Phi}}^{(s,3)} = \begin{bmatrix} -\mathbf{I} & \tilde{\mathbf{s}}_i^{\,P} & \mathbf{I} & -\tilde{\mathbf{s}}_j^{\,P} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{r}}_i \\ \dot{\boldsymbol{\omega}}_i \\ \ddot{\mathbf{r}}_j \\ \dot{\boldsymbol{\omega}}_j \end{Bmatrix} = -\overset{\sim}{\dot{\mathbf{s}}}_i^{\,P}\boldsymbol{\omega}_i + \overset{\sim}{\dot{\mathbf{s}}}_j^{\,P}\boldsymbol{\omega}_j \tag{9.7}$$

Thus, the contribution to the right-hand side of the acceleration of the spherical joint constraints is given by

$$\boldsymbol{\gamma}^{(s,3)} = -\overset{\sim}{\dot{\mathbf{s}}}_i^{\,P}\boldsymbol{\omega}_i + \overset{\sim}{\dot{\mathbf{s}}}_j^{\,P}\boldsymbol{\omega}_j \tag{9.8}$$

A revolute joint between bodies i and j, shown in Fig. 9.2, is built with a journal-bearing that allows a relative rotation about a common axis, but precludes relative translation along this axis. Equation (9.1) is imposed on an arbitrary point P located on the joint axis. Let now consider two vectors $\mathbf{a}_j$ and $\mathbf{b}_j$ on body j perpendicular to each other and perpendicular to the joint axis, as Fig. 9.2 shows. It is clear that these two vectors must remain perpendicular to vector $\mathbf{s}_i$ defined along the joint axis. Thus, there are five constraint equations for a revolute joint that can be written as

$$\boldsymbol{\Phi}^{(r,5)} \equiv \begin{cases} \boldsymbol{\Phi}^{(s,3)} \equiv \mathbf{r}_j + \mathbf{s}_j^{\,P} - \mathbf{r}_i - \mathbf{s}_i^{\,P} = \mathbf{0} \\ \Phi^{(n1,1)} \equiv \mathbf{s}_i^{\,T}\mathbf{a}_j = 0 \\ \Phi^{(n1,1)} \equiv \mathbf{s}_i^{\,T}\mathbf{b}_j = 0 \end{cases} \tag{9.9}$$

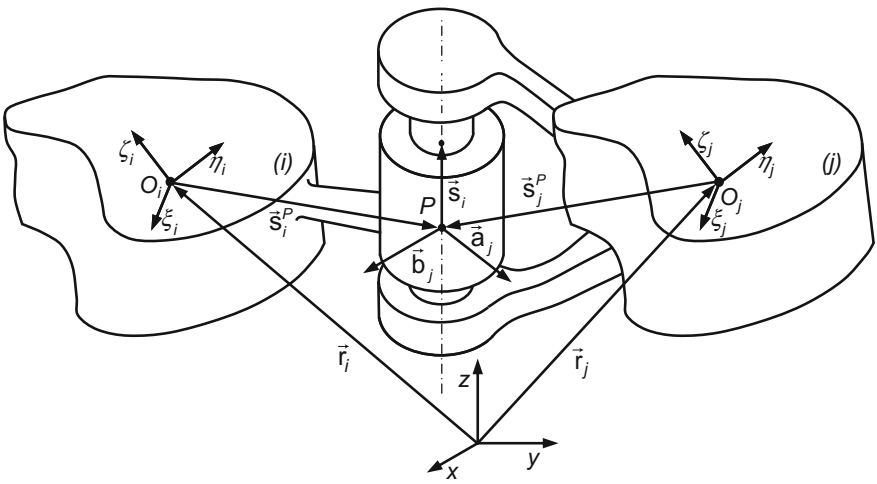

Fig. 9.2 Revolute joint connecting bodies i and j

It should be highlighted that these five scalar constraint equations yield only one relative degree of freedom for this joint, that is, rotation about the common axis of the revolute joint.

Derivate the last equation of Eq. (9.9) with respect to time yields

$$\dot{\Phi}^{(n1,1)} = \mathbf{b}_j{}^T\dot{\mathbf{s}}_i + \mathbf{s}_i^T\dot{\mathbf{b}}_j = 0 \tag{9.10}$$

Considering now Eq. (5.7), this equation can be written as

$$\dot{\Phi}^{(n1,1)} = -\mathbf{b}_j{}^T\tilde{\mathbf{s}}_i\boldsymbol{\omega}_i - \mathbf{s}_i{}^T\tilde{\mathbf{b}}_j\boldsymbol{\omega}_j = 0 \tag{9.11}$$

or in the matrix form

$$\dot{\Phi}^{(n1,1)} = \begin{bmatrix}\mathbf{0} & -\mathbf{b}_j{}^T\tilde{\mathbf{s}}_i & \mathbf{0} & -\mathbf{s}_i{}^T\tilde{\mathbf{b}}_j\end{bmatrix}\begin{Bmatrix}\dot{\mathbf{r}}_i\\ \boldsymbol{\omega}_i\\ \dot{\mathbf{r}}_j\\ \boldsymbol{\omega}_j\end{Bmatrix} = 0 \tag{9.12}$$

Thus, the Jacobian matrix terms of the revolute joint can be written as

$$\mathbf{D}^{(r,5)}_{(5\times12)} = \begin{bmatrix}-\mathbf{I} & \tilde{\mathbf{s}}_i{}^P & \mathbf{I} & -\tilde{\mathbf{s}}_j{}^P\\ \mathbf{0} & -\mathbf{a}_j{}^T\tilde{\mathbf{s}}_i & \mathbf{0} & -\mathbf{s}_i{}^T\tilde{\mathbf{a}}_j\\ \mathbf{0} & -\mathbf{b}_j{}^T\tilde{\mathbf{s}}_i & \mathbf{0} & -\mathbf{s}_i{}^T\tilde{\mathbf{b}}_j\end{bmatrix} \tag{9.13}$$

Differentiating now Eq. (9.10) with respect to time yields

$$\ddot{\Phi}^{(n1,1)} = \mathbf{s}_i{}^T\ddot{\mathbf{b}}_j + \mathbf{b}_j{}^T\ddot{\mathbf{s}}_i + 2\dot{\mathbf{b}}_j{}^T\dot{\mathbf{s}}_i = 0 \tag{9.14}$$

Substituting $\ddot{\mathbf{s}} = -\tilde{\mathbf{s}}\dot{\boldsymbol{\omega}} + \tilde{\boldsymbol{\omega}}\dot{s}$ in the above acceleration yields

$$\ddot{\Phi}^{(n1,1)} = -\mathbf{b}_j{}^T\tilde{\mathbf{s}}_i\dot{\boldsymbol{\omega}}_i - \mathbf{s}_i{}^T\tilde{\mathbf{b}}_j\boldsymbol{\omega}_j = \gamma \tag{9.15}$$

where the right-hand side term is expressed as

$$\gamma = -\mathbf{s}_i{}^T\tilde{\boldsymbol{\omega}}_j\dot{\mathbf{b}}_j - \mathbf{b}_j{}^T\tilde{\boldsymbol{\omega}}_i\dot{\mathbf{s}}_i - 2\dot{\mathbf{b}}_j{}^T\dot{\mathbf{s}}_i \tag{9.16}$$

Thus, the full right-hand side of the acceleration constraints of the revolute joint is given by

$$\boldsymbol{\gamma}^{(r,5)} = \begin{Bmatrix}-\overset{\sim}{\dot{\mathbf{s}}}_i{}^P\boldsymbol{\omega}_i + \overset{\sim}{\dot{\mathbf{s}}}_j{}^P\boldsymbol{\omega}_j\\ -\mathbf{s}_i{}^T\tilde{\boldsymbol{\omega}}_j\dot{\mathbf{a}}_j - \mathbf{a}_j{}^T\tilde{\boldsymbol{\omega}}_i\dot{\mathbf{s}}_i - 2\dot{\mathbf{a}}_j{}^T\dot{\mathbf{s}}_i\\ -\mathbf{s}_i{}^T\tilde{\boldsymbol{\omega}}_j\dot{\mathbf{b}}_j - \mathbf{b}_j{}^T\tilde{\boldsymbol{\omega}}_i\dot{\mathbf{s}}_i - 2\dot{\mathbf{b}}_j{}^T\dot{\mathbf{s}}_i\end{Bmatrix} \tag{9.17}$$

Figure 9.3 depicts two bodies connected by a rigid link containing two spherical joints, which makes a composite joint named spherical-spherical joint. In this particular joint the link between the bodies is modeled as a constraint and not as a third body. For a spherical-spherical joint only one constraint equation is required, which can be written in the form

$$\Phi^{(s-s,1)} \equiv \mathbf{d}^T\mathbf{d} - l^2 = 0 \tag{9.18}$$

where

$$\mathbf{d} = \mathbf{r}_j + \mathbf{s}_j^P - \mathbf{r}_i - \mathbf{s}_i^P \tag{9.19}$$

and l denotes the length of the link (Nikravesh 1988).

Differentiating Eq. (9.18) with respect to time yields the velocity constraint equation of a spherical-spherical joint that can be expressed as

$$\dot{\Phi}^{(s-s,1)} = 2\mathbf{d}^T(\dot{\mathbf{r}}_j + \dot{\mathbf{s}}_j^P - \dot{\mathbf{r}}_i - \dot{\mathbf{s}}_i^P) = 0 \tag{9.20}$$

Taking advantage of Eq. (5.7), this equation can be rewritten as

$$\dot{\Phi}^{(s-s,1)} = 2\mathbf{d}^T(\dot{\mathbf{r}}_j - \tilde{\mathbf{s}}_j^P\boldsymbol{\omega}_j - \dot{\mathbf{r}}_i + \tilde{\mathbf{s}}_i^P\boldsymbol{\omega}_i) = 0 \tag{9.21}$$

or in a compact form as

$$\dot{\Phi}^{(s-s,1)} = \begin{bmatrix} -2\mathbf{d}^T & 2\mathbf{d}^T\tilde{\mathbf{s}}_i^P & 2\mathbf{d}^T & -2\mathbf{d}^T\tilde{\mathbf{s}}_j^P \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{r}}_i \\ \boldsymbol{\omega}_i \\ \dot{\mathbf{r}}_j \\ \boldsymbol{\omega}_j \end{Bmatrix} = 0 \tag{9.22}$$

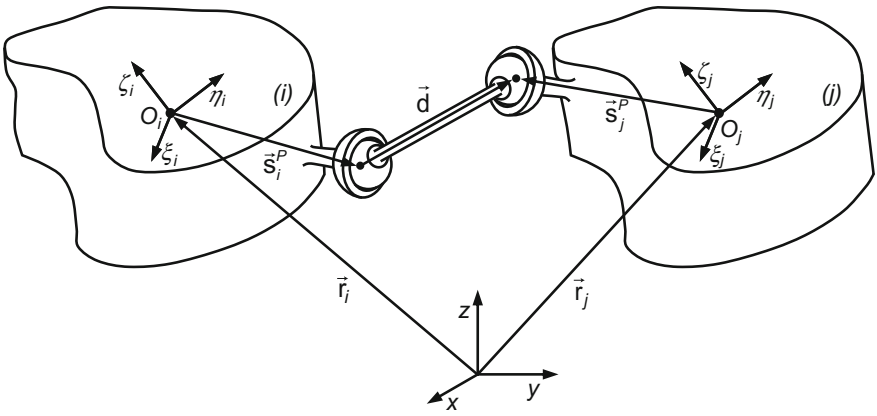

Fig. 9.3 Spherical-spherical joint connecting bodies i and j

Thus, the Jacobian matrix associated with a spherical-spherical joint is expressed in the form

$$\mathbf{D}_{(1\times 12)}^{(s-s,1)} = \begin{bmatrix} -2\mathbf{d}^T & 2\mathbf{d}^T\tilde{\mathbf{s}}_i^P & 2\mathbf{d}^T & -2\mathbf{d}^T\tilde{\mathbf{s}}_j^P \end{bmatrix} \tag{9.23}$$

Differentiating now Eq. (9.21) with respect to time yields the acceleration constraint equation as

$$\ddot{\Phi}^{(s-s,1)} = 2\mathbf{d}^T(\ddot{\mathbf{r}}_j - \tilde{\mathbf{s}}_j^P\dot{\omega}_j - \ddot{\mathbf{r}}_i + \tilde{\mathbf{s}}_i^P\dot{\boldsymbol{\omega}}_i) - 2\{\mathbf{d}^T(\tilde{\omega}_i\dot{\mathbf{s}}_i^P - \tilde{\boldsymbol{\omega}}_j\dot{\mathbf{s}}_j^P) - \dot{\mathbf{d}}^T\dot{\mathbf{d}}\} = 0 \tag{9.24}$$

or in a compact form as

$$\ddot{\boldsymbol{\Phi}}^{(s-s,1)} = \begin{bmatrix} 2\mathbf{d}^T & -2\mathbf{d}^T\tilde{\mathbf{s}}_i^P & 2\mathbf{d}^T & 2\mathbf{d}^T\tilde{\mathbf{s}}_j^P \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{r}}_i \\ \dot{\boldsymbol{\omega}}_i \\ \ddot{\mathbf{r}}_j \\ \dot{\boldsymbol{\omega}}_j \end{Bmatrix} = \gamma \tag{9.25}$$

where the right-hand side of the accelerations is expressed as

$$\gamma^{(s-s,1)} = 2\mathbf{d}^T(\{\tilde{\boldsymbol{\omega}}_i\dot{\mathbf{s}}_i^P - \{\tilde{\boldsymbol{\omega}}_j\dot{\mathbf{s}}_j^P) - 2\dot{\mathbf{d}}^T\dot{\mathbf{d}} \tag{9.26}$$

References

Flores P, Ambrósio J, Claro JCP, Lankarani HM (2008) Kinematics and dynamics of multibody systems with imperfect joints: models and case studies. Lecture notes in applied and computational mechanics, vol 34. Springer, Berlin

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Schiehlen W (1990) Multibody systems handbook. Springer, Berlin

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 10
Equations of Motion for Constrained Systems

Abstract In this chapter, the formulation of motion's equations of multi-rigid body systems is described. The generalized coordinates are the centroidal Cartesian coordinates, being the system configuration restrained by constraint equations. The present formulation uses the Newton-Euler's equations of motion, which are augmented with the constraint equations that lead to a system of differential algebraic equations. This formulation is straightforward in terms of assembling the equations of motion and providing all reaction forces.

Keywords Equations of motion · Newton-Euler formulation · Spatial systems

The translational equations of motion for an unconstrained rigid body can be expressed as (Shabana 1989; Schiehlen 1990)

$$m\ddot{\mathbf{r}} = \mathbf{f} \tag{10.1}$$

in which m represents the mass of the body, $\ddot{\mathbf{r}}$ denotes the acceleration of the center of mass and $\mathbf{f}$ represents the sum of all forces acting on the body (Jalón and Bayo 1994; Nikravesh 2008).

Nikravesh (1988) demonstrated that the rotational equations of motion for a rigid body can be written in the form

$$J\dot{\boldsymbol{\omega}} + \tilde{\boldsymbol{\omega}}J\omega = \mathbf{n} \tag{10.2}$$

where $\mathbf{J}$ is the global inertia tensor, $\dot{\boldsymbol{\omega}}$ denotes the global angular accelerations, $\boldsymbol{\omega}$ is global angular velocities and $\mathbf{n}$ represents the sum of all moments acting on the body. Thus, the translational and rotational equations of motion, also known as the Newton-Euler equations of motion, for an unconstrained rigid body can be obtained by combining Eqs. (10.1) and (10.2), which in the matrix form are written as

$$\begin{bmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{J} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{r}} \\ \dot{\boldsymbol{\omega}} \end{Bmatrix} + \begin{Bmatrix} \mathbf{0} \\ \tilde{\boldsymbol{\omega}}J\omega \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{n} \end{Bmatrix} \tag{10.3}$$

or, alternatively,

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_10

$$\begin{bmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{J} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{r}} \\ \dot{\boldsymbol{\omega}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{n} - \tilde{\boldsymbol{\omega}} J \omega \end{Bmatrix} \tag{10.4}$$

The equations of motion can also be derived and expressed in terms of local components, namely the rotational equations of motion. However, the form how the equations of motion are presented here is consistent with the kinematic constraints offered in the previous sections. Thus, in a compact form, Eq. (10.4) can be expressed as

$$\mathbf{M}_i \dot{\mathbf{v}}_i = \mathbf{g}_i \tag{10.5}$$

where

$$\mathbf{M}_i = \begin{bmatrix} m_i\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_i \end{bmatrix}, \quad \dot{\mathbf{v}}_i = \begin{Bmatrix} \ddot{\mathbf{r}}_i \\ \dot{\boldsymbol{\omega}}_i \end{Bmatrix}, \quad \mathbf{g}_i = \begin{Bmatrix} \mathbf{f}_i \\ \mathbf{n}_i - \tilde{\boldsymbol{\omega}}_i \mathbf{J}_i \boldsymbol{\omega}_i \end{Bmatrix} \tag{10.6}$$

Hence, the Newton-Euler equations of motion of a multibody system composed by n_b unconstrained bodies are written as

$$\mathbf{M}\dot{\mathbf{v}} = \mathbf{g} \tag{10.7}$$

in which

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & & & \\ & \mathbf{M}_1 & & \\ & & \ddots & \\ & & & \mathbf{M}_{n_b} \end{bmatrix}, \quad \dot{\mathbf{v}} = \begin{Bmatrix} \dot{\mathbf{v}}_1 \\ \dot{\mathbf{v}}_2 \\ \vdots \\ \dot{\mathbf{v}}_{n_b} \end{Bmatrix}, \quad \mathbf{g} = \begin{Bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \vdots \\ \mathbf{g}_{n_b} \end{Bmatrix} \tag{10.8}$$

In turn, for a multibody system of constrained bodies, the Newton-Euler equations of motion are written as (Nikravesh 1988)

$$\mathbf{M}\dot{\mathbf{v}} = \mathbf{g} + \mathbf{g}^{(c)} \tag{10.9}$$

where $\mathbf{g}^{(c)}$ denotes the vector of reaction forces that can be expressed in terms of the Jacobian matrix and Lagrange multipliers as (Nikravesh 1988; Jalón and Bayo 1994)

$$\mathbf{g}^{(c)} = \mathbf{D}^T \boldsymbol{\lambda} \tag{10.10}$$

Finally, the dynamic equations of motion for a constrained multibody system can be written in its general form as

$$\mathbf{M}\dot{\mathbf{v}} - \mathbf{D}^T \boldsymbol{\lambda} = \mathbf{g} \tag{10.11}$$

Different methods of resolution of the equations of motion will be presented and discussed in the next sections of the present document.

It is known that physically, the Lagrange multipliers are related to the joint reaction forces. In what follows, the relation between the constrained reaction forces and the constraint equations is revisited. For this purpose, let first consider that $\mathbf{g}^{(c)}$ can be transformed to a coordinate system consistent with $\mathbf{q}$ and denoted as $\mathbf{g}^{(*)}$. Furthermore, it is assumed that there are m independent constraint equations written as

$$\mathbf{\Phi} \equiv \mathbf{\Phi}(\mathbf{q}) = 0 \tag{10.12}$$

For frictionless kinematic joints, the work done by the constraint forces in a virtual displacement $\delta\mathbf{q}$ is zero, i.e.,

$$\mathbf{g}^{(*)T}\delta\mathbf{q} = 0 \tag{10.13}$$

Since the virtual displacement $\delta\mathbf{q}$ must be consistent with the constraint equations, then Eq. (10.12) yields

$$\mathbf{D}\delta\mathbf{q} = \mathbf{0} \tag{10.14}$$

The vector of n coordinates $\mathbf{q}$ may be partitioned into a set of m dependent coordinates $\mathbf{u}$, and a set of n-m independent coordinates $\mathbf{v}$, as

$$\mathbf{q} \equiv \begin{bmatrix} \mathbf{u}^T & \mathbf{v}^T \end{bmatrix}^T \tag{10.15}$$

This yields a partitioned vector of virtual displacements and a partitioned Jacobian matrix as

$$\delta\mathbf{q} \equiv \begin{bmatrix} \delta\mathbf{u}^T & \delta\mathbf{v}^T \end{bmatrix}^T, \quad \mathbf{D} = \begin{bmatrix} \mathbf{D}_\mathbf{u} & \mathbf{D}_\mathbf{v} \end{bmatrix} \tag{10.16}$$

Hence, Eq. (10.13) can be rewritten as

$$\mathbf{g}_\mathbf{u}^{(*)T}\delta\mathbf{u} + \mathbf{g}_\mathbf{v}^{(*)T}\delta\mathbf{v} = 0 \tag{10.17}$$

or

$$\mathbf{g}_\mathbf{u}^{(*)T}\delta\mathbf{u} = -\mathbf{g}_\mathbf{v}^{(*)T}\delta\mathbf{v} \tag{10.18}$$

In a similar way, from Eq. (10.14) yields

$$\mathbf{D}_\mathbf{u}\delta\mathbf{u} = -\mathbf{D}_\mathbf{v}\delta\mathbf{v} \tag{10.19}$$

Appending now Eqs. (10.18) and (10.19) results in

$$\begin{bmatrix} \mathbf{g}_{\mathbf{u}}^{(*)T} \\ \mathbf{D}_{\mathbf{u}} \end{bmatrix} \delta\mathbf{u} = -\begin{bmatrix} \mathbf{g}_{\mathbf{v}}^{(*)T} \\ \mathbf{D}_{\mathbf{v}} \end{bmatrix} \delta\mathbf{v} \tag{10.20}$$

The matrix to the left in Eq. (10.20) is an $(m + 1) \times m$ matrix. Since $\mathbf{D}_{\mathbf{u}}$ is an $m \times m$ nonsingular matrix, the first row of the $(m + 1) \times m$ matrix can be expressed as a linear combination of the other rows of the matrix as

$$\mathbf{g}_{\mathbf{u}}^{(*)} = \mathbf{D}_{\mathbf{u}}^{T} \boldsymbol{\lambda} \tag{10.21}$$

where $\boldsymbol{\lambda}$ is an m-vector of multipliers known as Lagrange multipliers. Substituting now Eq. (10.21) into (10.18) yields

$$\boldsymbol{\lambda}^{T} \mathbf{D}_{\mathbf{u}} \delta\mathbf{u} = -\mathbf{g}_{\mathbf{v}}^{(*)T} \delta\mathbf{v} \tag{10.22}$$

or

$$-\boldsymbol{\lambda}^{T} \mathbf{D}_{\mathbf{v}} \delta\mathbf{v} = -\mathbf{g}_{\mathbf{v}}^{(*)T} \delta\mathbf{v} \tag{10.23}$$

in which Eq. (10.19) has been employed. Vector $\delta\mathbf{v}$ is an arbitrary independent vector. The consistency of the constraints for virtual displacements $\delta\mathbf{q}$ is guaranteed by solving Eq. (10.19) for $\delta\mathbf{u}$. Since Eq. (10.23) must hold for any arbitrary $\delta\mathbf{v}$, then

$$\boldsymbol{\lambda}^{T} \mathbf{D}_{\mathbf{v}} = \mathbf{g}_{\mathbf{v}}^{(*)T} \tag{10.24}$$

or

$$\mathbf{g}_{\mathbf{v}}^{(*)} = \mathbf{D}_{\mathbf{v}}^{T} \boldsymbol{\lambda} \tag{10.25}$$

Appending Eq. (10.21)–(10.25) yields

$$\mathbf{g}^{(*)} = \mathbf{D}^{T} \boldsymbol{\lambda} \tag{10.26}$$

which expresses the constraint reaction forces.

References

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs, New Jersey

Nikravesh PE (2008) Planar multibody dynamics: formulation, programming, and applications. CRC Press, London
Schiehlen W (1990) Multibody systems handbook. Springer, Berlin
Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 11
Force Elements and Reaction Forces

Abstract In the present chapter some of the most relevant applied forces and joint reaction forces are introduced. There are many types of forces that can be present in multibody systems, such as gravitational forces, spring-damper-actuator forces, normal contact forces, tangential or frictional forces, external applied forces and moments, forces due to elasticity of bodies, and thermal, electrical and magnetic forces. However, only the first six types of forces are relevant in the multibody systems of common application.

Keywords Springer-damper forces · Contact forces · Reaction forces

Figure 11.1 illustrates a body i acted upon by a gravitational field in the negative z direction. The choice of the negative z direction as the direction of gravity is totally arbitrary. However, in the present work, the gravitational field will be considered to be acting in this direction unless indicated otherwise. If w_i is the weight of the body i, resulting from the product of mass of the body by the gravitational constant, then the contribution of this force to the generalized vector of forces of body i is given by (Nikravesh 1988; Shabana 1989)

$$\mathbf{g}_i^{(g)} = \{0 \quad 0 \quad -w_i \quad 0 \quad 0 \quad 0\}^T \tag{11.1}$$

Consider a single body force $\mathbf{f}_i$ acting on body i at point P_i, as shown in Fig. 11.2a. This force has three Cartesian components. In addition, a moment with respect to the body center of mass must be computed as (Jalón and Bayo 1994)

$$\mathbf{n}_i = \tilde{\mathbf{s}}_i^P \mathbf{f}_i \tag{11.2}$$

Thus, the contribution to generalized vector of forces of a single force is

$$\mathbf{g}_i^{(f)} \equiv \begin{Bmatrix} \mathbf{f}_i \\ \mathbf{n}_i \end{Bmatrix} \tag{11.3}$$

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_11

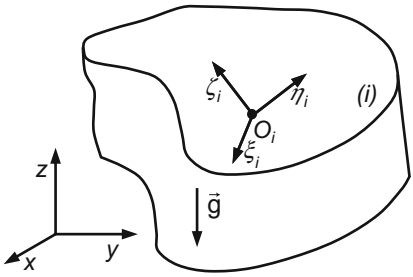

Fig. 11.1 Gravitational field acting on a body i

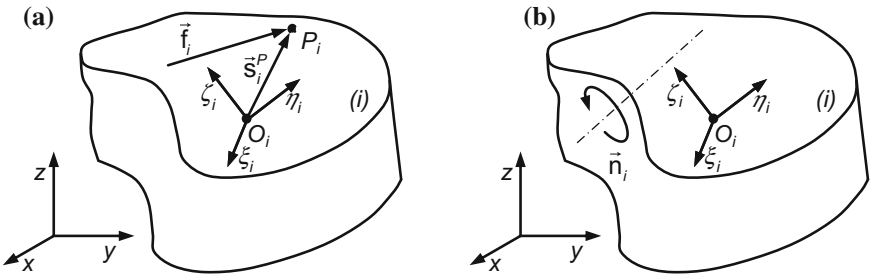

Fig. 11.2 A body i acted upon by **a** a single force **b** a pure moment

When a pure moment with magnitude n_i acts on a body i, as shown in Fig. 11.2b, its contribution to the vector of forces of body i is given by

$$\mathbf{g}_i^{(n)} \equiv \begin{Bmatrix} \mathbf{0} \\ \mathbf{n}_i \end{Bmatrix} \tag{11.4}$$

Figure 11.3 shows a spring-damper-actuator element connecting bodies i and j through two points of connectivity P_i and P_j. The vector $\mathbf{l}$ that connects the points P_i and P_j can be evaluated by

$$\mathbf{l} = \mathbf{r}_j^P - \mathbf{r}_i^P = \mathbf{r}_j + \mathbf{A}_j \mathbf{s}_j'^P - \mathbf{r}_i - \mathbf{A}_i \mathbf{s}_i'^P \tag{11.5}$$

The magnitude of this vector is

$$l = \sqrt{\mathbf{l}^T \mathbf{l}} \tag{11.6}$$

The unit vector along the spring-damper-actuator element is defined as

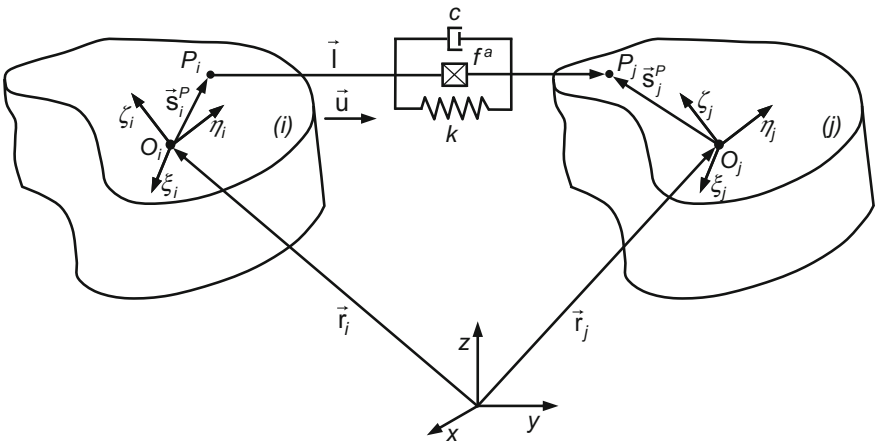

Fig. 11.3 Spring-damper-actuator element connecting bodies i and j

$$\mathbf{u} = \frac{1}{l}\mathbf{l} \tag{11.7}$$

The time rate of change of the damper length can be obtained by differentiating Eq. (11.6), yielding

$$\dot{l} = \frac{1}{l}\mathbf{l}^T\dot{\mathbf{l}} \tag{11.8}$$

where $\dot{\mathbf{l}}$, in turn, is found from Eq. (11.5)

$$\dot{\mathbf{l}} = \dot{\mathbf{r}}_j^P - \dot{\mathbf{r}}_i^P \tag{11.9}$$

Then, the resulting spring-damper-actuator force is evaluated as

$$f^{sda} = k(l - l^0) + c\dot{l} + f^a \tag{11.10}$$

where the first term on the right-hand side is the spring force, the second term represents the damper force and the third term denotes the actuator force. The spring stiffness is represented by k, l is the deformed length, l^0 is the undeformed or natural length of the spring, c is the damping coefficient of the damper and $\dot{l}$ is the time rate of change of the damper length.

The forces that act on the bodies i and j can be evaluated as

$$\mathbf{f}_i^{sda} = f^{sda}\mathbf{u} \quad \text{and} \quad \mathbf{f}_j^{sda} = -f^{sda}\mathbf{u} \tag{11.11}$$

Finally, the contribution to generalized vector of forces is given by

$$\mathbf{g}_i^{(sda)} \equiv \begin{Bmatrix} \mathbf{f}_i^{sda} \\ \tilde{\mathbf{s}}_i^P \mathbf{f}_i^{sda} \end{Bmatrix} \quad \text{and} \quad \mathbf{g}_j^{(sda)} \equiv \begin{Bmatrix} \mathbf{f}_j^{sda} \\ \tilde{\mathbf{s}}_j^P \mathbf{f}_j^{sda} \end{Bmatrix} \tag{11.12}$$

As it was presented previously, the joint reaction forces and moments are expressed in terms of the Jacobian matrix of the constraint equations and a vector of Lagrange multipliers and expressed by Eq. (10.10). Thus, for instance, for a spherical joint between bodies i and j, the vector of reaction forces is expressed as

$$\mathbf{D}^T \boldsymbol{\lambda} = \begin{bmatrix} -\mathbf{I} \\ \tilde{\mathbf{s}}_i^P \\ \mathbf{I} \\ -\tilde{\mathbf{s}}_j^P \end{bmatrix} \boldsymbol{\lambda} \tag{11.13}$$

Equation (9.5) has been considered in Eq. (11.13). For a spherical joint, $\boldsymbol{\lambda}$ is a 3-vector representing exactly the reaction force acting at point P_i. The same force but in the opposite direction acts at point P_j, as it is shown in Fig. 11.4. It must be noted that a spherical joint does not produce a reaction moment. However, when a reaction force is moved to the corresponding mass center, the moment associated with that force must be included in the rotational equations of motion. These reaction moments are automatically taken care of by the Jacobian matrix (Nikravesh 1988).

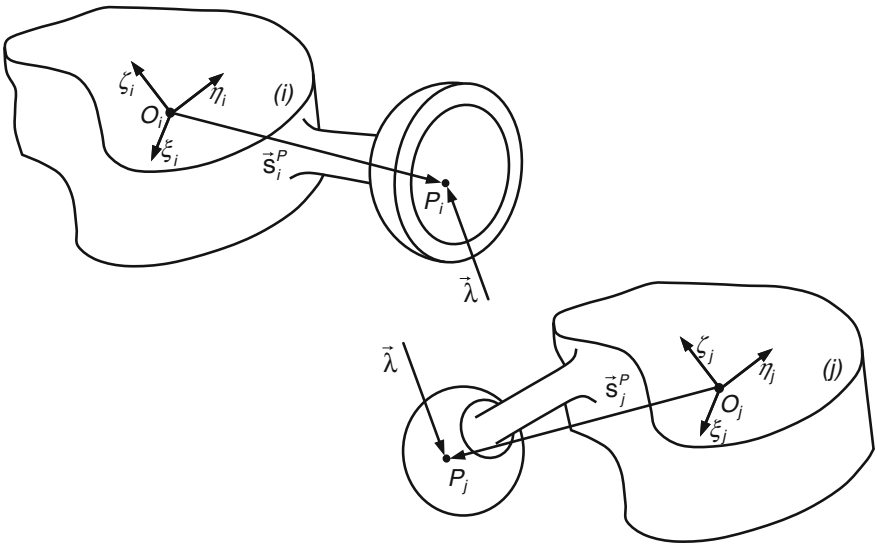

Fig. 11.4 Reaction forces associated with a spherical joint

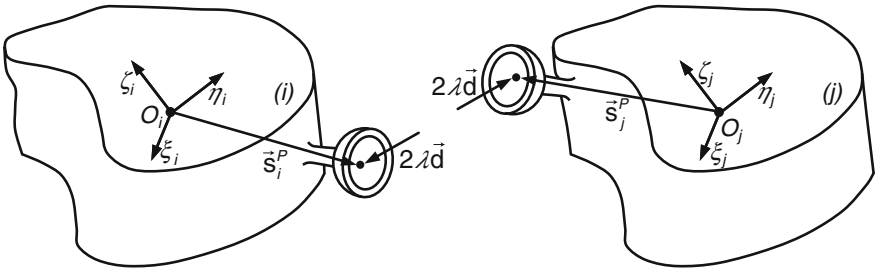

Fig. 11.5 Reaction forces associated with a spherical-spherical joint

For the case of a spherical-spherical joint, the reaction force can be expressed in the form

$$\mathbf{D}^T\boldsymbol{\lambda} = \begin{bmatrix} -2\mathbf{d}^T \\ 2\mathbf{d}^T\tilde{\mathbf{s}}_i^P \\ 2\mathbf{d}^T \\ -2\mathbf{d}^T\tilde{\mathbf{s}}_j^P \end{bmatrix}\boldsymbol{\lambda} = \begin{bmatrix} -\mathbf{I} \\ \tilde{\mathbf{s}}_i^P \\ \mathbf{I} \\ -\tilde{\mathbf{s}}_j^P \end{bmatrix} 2\mathbf{d}\lambda \tag{11.14}$$

in which Eq. (10.23) has been employed. This composite joint contains a single Lagrange multiplier which its value is proportional to the magnitude of the reaction force. This reaction force acts exactly along the axis of the link that defines the joint, as it is illustrated in Fig. 11.5. The reaction moments are the result of the reaction forces having arms with respect to their corresponding center of mass (Schiehlen 1990).

References

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Schiehlen W (1990) Multibody systems handbook. Springer, Berlin

Shabana AA (1989) Dynamics of multibody systems. Wiley, New York

Chapter 12
Methods to Solve the Equations of Motion

Abstract This chapter presents several methods to solve the equations of motion of spatial multibody systems. In particular, the standard approach, the Baumgarte method, the penalty method and the augmented Lagrangian formulation are revised here. In this process, a general procedure for dynamic analysis of multibody systems based on the standard Lagrange multipliers method is described. Moreover, the implications in terms of the resolution of the equations of motion, accuracy and efficiency are also discussed in this chapter.

Keywords Dynamic analysis · Baumgarte method · Penalty method

As it was presented previously, the Newton-Euler equations of motion for a constrained multibody system of rigid bodies are written as

$$\mathbf{M}\dot{\mathbf{v}} - \mathbf{D}^T\boldsymbol{\lambda} = \mathbf{g} \tag{12.1}$$

In dynamic analysis, a unique solution is obtained when the algebraic constraint equations at the acceleration level are considered simultaneously with the differential equations of motion. Therefore, the second time derivative of the constraint equations are considered here and written as

$$\mathbf{D}\dot{\mathbf{v}} = \boldsymbol{\gamma} \tag{12.2}$$

Equation (12.2) can be appended to Eq. (12.1), yielding a system of differential algebraic equations (DAE). This system of equations is solved for accelerations vector, $\dot{\mathbf{v}}$, and Lagrange multipliers, $\boldsymbol{\lambda}$. Then, in each integration time step, the accelerations vector, $\dot{\mathbf{v}}$, together with velocities vector, $\mathbf{v}$, is integrated in order to obtain the system velocities and positions for the next time step. This procedure is repeated until the final analysis time is reached. A set of initial conditions, positions and velocities, is required to start the dynamic simulation. In the present work, the initial conditions are based on the results of kinematic simulation of the mechanical systems. The subsequent initial conditions for each time step in the simulation are obtained in the usual manner from the final conditions of the previous time step (Nikravesh 2007).

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_12

Equations (12.1) and (12.2) can be rewritten in the matrix form as

$$\begin{bmatrix} \mathbf{M} & \mathbf{D}^T \\ \mathbf{D} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{v}} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{g} \\ \boldsymbol{\gamma} \end{Bmatrix} \tag{12.3}$$

The linear system of Eq. (12.3) can be solved by applying any method suitable for the solution of linear algebraic equations. The existence of null elements in the main diagonal of the leading matrix and the possibility of ill-conditioned matrices suggest that methods using partial or full pivoting are preferred. However, none of these formulations help in the presence of redundant constraints. Alternatively, the equations of motion can be solved analytically. For this purpose, Eq. (12.1) is rearranged to put the accelerations vector in evidence, yielding

$$\dot{\mathbf{v}} = \mathbf{M}^{-1}(\mathbf{g} + \mathbf{D}^T\boldsymbol{\lambda}) \tag{12.4}$$

In this process, it is assumed that the multibody system under analysis does not include any body with null mass or inertia so that the inverse of the mass matrix **M** exists. Thus, introducing Eq. (12.4) into Eq. (12.2) and after basic mathematical manipulation results in

$$\boldsymbol{\lambda} = \left[\mathbf{D}\mathbf{M}^{-1}\mathbf{D}^T\right]^{-1}(\boldsymbol{\gamma} - \mathbf{D}\mathbf{M}^{-1}\mathbf{g}) \tag{12.5}$$

Substituting now Eq. (12.5) into Eq. (12.4) yields

$$\dot{\mathbf{v}} = \mathbf{M}^{-1}\mathbf{g} + \mathbf{M}^{-1}\mathbf{D}^T\left\{\left[\mathbf{D}\mathbf{M}^{-1}\mathbf{D}^T\right]^{-1}(\boldsymbol{\gamma} - \mathbf{D}\mathbf{M}^{-1}\mathbf{g})\right\} \tag{12.6}$$

Thus, Eq. (12.6) can be solved for $\dot{\mathbf{v}}$ then, the velocities and positions can be obtained by integration process in a similar manner as it was described above. This manner to solve the dynamic equations of motion is commonly referred to as the standard Lagrange multipliers method (Nikravesh 1988). Figure 12.1 presents a flowchart that shows the algorithm of the standard solution of the equations of motion. At $t = t^0$, the initial conditions on $\mathbf{q}^0$ and $\mathbf{v}^0$ are required to start the integration process. These values cannot be specified arbitrarily, but must satisfy the constraint equations defined by Eqs. (7.2) and (7.3). The algorithm presented in Fig. 12.1 can be summarized by the following steps:

1. Start at instant of time t^0 with given initial conditions for positions $\mathbf{q}^O$ and velocities $\mathbf{v}^0$.
2. Assemble the global mass matrix **M**, evaluate the Jacobian matrix **D**, construct the constraint equations **Φ**, determine the right-hand side of the accelerations $\boldsymbol{\gamma}$, and calculate the force vector **g**.
3. Solve the linear set of the equations of motion (12.3) for a constrained multibody system in order to obtain the accelerations $\dot{\mathbf{v}}$ at instant t and the Lagrange multipliers $\boldsymbol{\lambda}$.

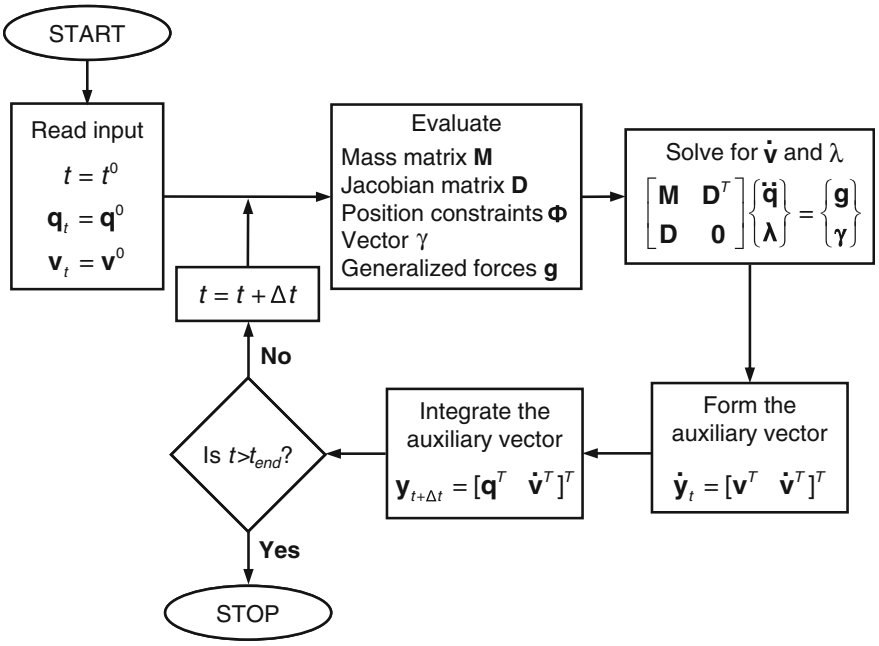

Fig. 12.1 Flowchart of computational procedure for dynamic analysis of multibody systems based on the standard Lagrange multipliers method

4. Assemble the vector $\dot{\mathbf{y}}_t$ containing the generalized velocities $\mathbf{v}$ and accelerations $\dot{\mathbf{v}}$ for instant of time t.
5. Integrate numerically the $\mathbf{v}$ and $\dot{\mathbf{v}}$ vectors for time step $t + \Delta t$ and obtain the new positions and velocities.
6. Update the time variable, go to step (2) and proceed with the process for a new time step, until the final time of analysis is reached.

The system of the motion Eq. (12.3) does not use explicitly the position and velocity equations associated with the kinematic constraints, that is, Eqs. (7.2) and (7.3). Consequently, for moderate or long simulations, the original constraint equations start to be violated due to the integration process and/or to inaccurate initial conditions. Therefore, methods able to eliminate errors in the position or velocity equations or, at least, to keep such errors under control, must be implemented. In order to keep the constraint violations under control, the Baumgarte stabilization method is considered here (Baumgarte 1972). This method allows constraints to be slightly violated before corrective actions can take place, in order to force the violation to vanish. The objective of Baumgarte method is to replace the differential Eq. (7.5) by the following equation

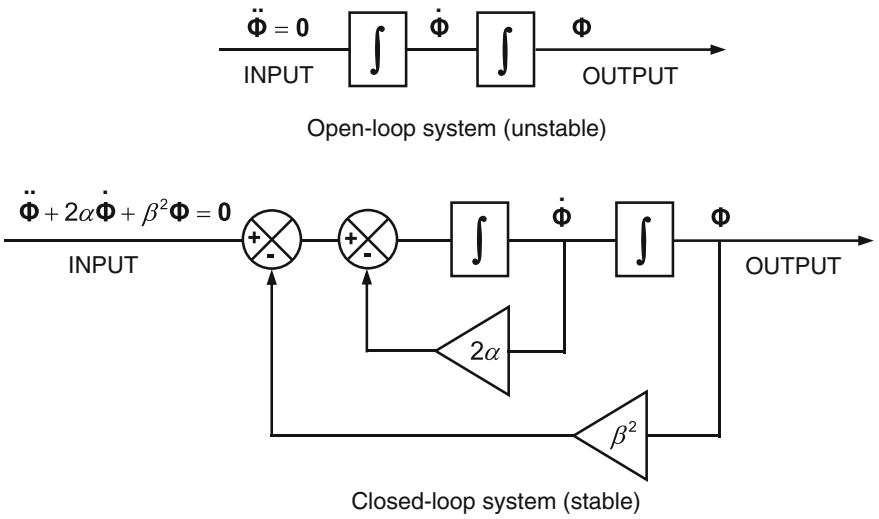

Fig. 12.2 Open loop and closed loop control systems

$$\ddot{\Phi} + 2\alpha\dot{\Phi} + \beta^2\Phi = \mathbf{0} \tag{12.7}$$

Equation (12.7) is a differential equation for a closed-loop system in terms of kinematic constraint equations, in which the terms $2\alpha\dot{\Phi}$ and $\beta^2\Phi$ play the role of control terms. The principle of the method is based on the damping of acceleration of constraint violation by feeding back the position and velocity of constraint violations, as illustrated in Fig. 12.2, which shows open-loop and closed-loop control systems. In the open-loop systems Φ and $\dot{\Phi}$ do not converge to zero if any perturbation occurs and, therefore, the system is unstable. Thus, using the Baumgarte approach, the equations of motion for a system subjected to constraints are stated in the following form

$$\begin{bmatrix} \mathbf{M} & \mathbf{D}^T \\ \mathbf{D} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{v}} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{g} \\ \boldsymbol{\gamma} - 2\alpha\dot{\Phi} - \beta^2\Phi \end{Bmatrix} \tag{12.8}$$

If α and β are chosen as positive constants, the stability of the general solution of Eq. (12.8) is guaranteed. Baumgarte (1972) highlighted that the suitable choice of the parameters α and β is performed by numerical experiments. Hence, the Baumgarte method has some ambiguity in determining optimal feedback gains. Indeed, it seems that the value of the parameters is purely empiric, and there is no reliable method for selecting the coefficients α and β. The improper choice of these coefficients can lead to unacceptable results in the dynamic analysis of the multibody systems (Nikravesh 1984; Flores et al. 2011).

The penalty method presented by Jalón and Bayo (1994) constitutes an alternative way to solve the equations of motion. In this method, the equations of motion are modeled as a linear second-order differential equation that can be written in the form

$$m_c\ddot{\Phi} + d_c\dot{\Phi} + k_c\Phi = \mathbf{0} \tag{12.9}$$

Introducing Eq. (7.5) into Eq. (12.9) yields

$$m_c(\mathbf{D}\dot{\mathbf{v}} + \dot{\mathbf{D}}\mathbf{v}) + d_c\dot{\Phi} + k_c\mathbf{\Phi} = \mathbf{0} \tag{12.10}$$

Pre-multiplying Eq. (12.10) by the transpose of Jacobian matrix, $\mathbf{D}^T$, and after mathematical treatment, results in

$$m_c\mathbf{D}^T\mathbf{D}\dot{\mathbf{v}} = -\mathbf{D}^T(m_c\dot{\mathbf{D}}\mathbf{v} + d_c\dot{\Phi} + k_c\mathbf{\Phi}) \tag{12.11}$$

Let consider now the Newton-Euler equations of motion for a system of unconstrained system and written here as

$$\mathbf{M}\dot{\mathbf{v}} = \mathbf{g} \tag{12.12}$$

Adding Eqs. (12.12) and (12.11) yields

$$\mathbf{M}\dot{\mathbf{v}} + m_c\mathbf{D}^T\mathbf{D}\dot{\mathbf{v}} = \mathbf{g} - \mathbf{D}^T(-m_c\boldsymbol{\gamma} + d_c\dot{\Phi} + k_c\mathbf{\Phi}) \tag{12.13}$$

in which Eq. (7.6) has been employed. Finally, Eq. (12.13) can be written in the following form

$$(\mathbf{M} + \alpha\mathbf{D}^T\mathbf{D})\dot{\mathbf{v}} = \mathbf{g} - \alpha\mathbf{D}^T(-\boldsymbol{\gamma} + 2\mu\omega\dot{\Phi} + \omega^2\mathbf{\Phi}) \tag{12.14}$$

where

$$\alpha = m_c, \quad \frac{d_c}{m_c} = 2\mu\omega \quad \text{and} \quad \frac{k_c}{m_c} = \omega^2 \tag{12.15}$$

Equation (12.14) can be solved for $\dot{\mathbf{v}}$. This method gives good results if α tends to infinity. Typical values of α, ω and μ are 1×10^7, 10 and 1, respectively (Jalón and Bayo 1994). It should be noted that with this penalty method, multibody systems with redundant constraints or kinematic singular configurations can be easily solved.

The augmented Lagrangian formulation is a methodology that penalizes the constraint violations, much in the same form as the Baumgarte stabilization method (Baumgarte 1972). This is an iterative procedure that presents a number of advantages relative to other methods because it involves the solution of a smaller set of equations, handles redundant constraints and still delivers accurate results in

the vicinity of singular configurations. The augmented Lagrangian formulation consists of solving the system equations of motion by an iterative process. Let index i denote the i-th iteration. The evaluation of the system accelerations in a given time step starts as (Jalón and Bayo 1994)

$$\mathbf{M}\dot{\mathbf{v}}_i = \mathbf{g}, \quad (i = 0) \tag{12.16}$$

The iterative process to evaluate the system accelerations proceeds with the evaluation of

$$(\mathbf{M} + \alpha\mathbf{D}^T\mathbf{D})\dot{\mathbf{v}}_{i+1} = \mathbf{M}\dot{\mathbf{v}}_i - \alpha\mathbf{D}^T(-\gamma + 2\mu\omega\dot{\mathbf{\Phi}} + \omega^2\mathbf{\Phi}) \tag{12.17}$$

The iterative process continues until

$$\|\dot{\mathbf{v}}_{i+1} - \dot{\mathbf{v}}_i\| = \varepsilon \tag{12.18}$$

where ε is a specified tolerance. The augmented Lagrangian formulation involves the solution of a system of equations with a dimension equal to the number of coordinates of the multibody system. Though mass matrix $\mathbf{M}$ is generally positive semi-definite the leading matrix of Eq. (12.17) $\mathbf{M} + \alpha\mathbf{D}^T\mathbf{D}$ is always positive definite (Jalón and Bayo 1994). Even when the system is close to a singular position or when in presence of redundant constraints the system of equations can still be solved.

References

Baumgarte J (1972) Stabilization of constraints and integrals of motion in dynamical systems. Comput Method Appl Mech Eng 1:1–16

Flores P, Machado M, Seabra E, Silva MT (2011) A parametric study on the Baumgarte stabilization method for forward dynamics of constrained multi-body systems. J Comput Nonlinear Dyn 6(1):011019, 9p

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1984) Some methods for dynamic analysis of constrained mechanical systems: a survey. In: Haug EJ (ed) Computer-Aided analysis and Optimization of Mechanical System Dynamics, Springer Verlag, Berlin, Germany, pp 351–368

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Nikravesh PE (2007) Initial condition correction in multibody dynamics. Multibody Syst Dyn 18:107–115

Chapter 13
Integration Methods in Dynamic Analysis

Abstract This chapter describes the main integration algorithms utilized in the resolution of the dynamics equations of motion. Particular emphasis is paid to the Euler method, Runge-Kutta approach and Adams predictor-corrector method that allows for the use of variable time steps during the integration process. The material presented here, relative to numerical integration of ordinary differential equations, follows that of any undergraduate text on numerical analysis.

Keywords Euler method · Runge-Kutta method · Adams predictor-corrector method

In the previous paragraph, the equations of motion for multibody systems were derived from the Newton-Euler formulation together with the augmentation method. The Newton-Euler equations represent the translational and rotational motions of bodies, while the augmentation method is used to adjoin the constraint equations of the multibody systems. In other words, the augmentation formulation denotes the process where the algebraic kinematic constraint equations are augmented to the differential equations of motion, in order that the number of unknowns for which the system is being solved corresponds to the number of system equations (Nikravesh 1988). As a consequence, the equations of motion of multibody systems (12.1) are differential and algebraic equations (DAE) rather than ordinary differential equations (ODE) (Blajer 1999). Prior to integrate the system state variables, Eq. (12.1) is solved for $\dot{\mathbf{v}}$ and $\boldsymbol{\lambda}$.

In the present work, the DAE are converted to ODE because the most frequently used numerical integration algorithms are useful in solving ODE (Shampine and Gordon 1975). However, for a detailed discussion on DAE, the interested reader may consult the works by Petzold (1983) and Brenan et al. (1989). The material presented below, relative to numerical integration of ODE, follows that of any undergraduate text on numerical analysis such as those by Conte and Boor (1981) and Atkinson (1989).

The process of converting n second-order differential equations to $2n$ first-order equations can be expressed by (Shampine and Gordon 1975; Conte and Boor 1981; Atkinson 1989)

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_13

$$\ddot{y}_1 = f(y_1, \dot{y}_1, t) \tag{13.1}$$

such that it can be written as the following system

$$\dot{y}_1 = y_2 \tag{13.2}$$

$$\dot{y}_2 = f(y_1,\ y_2,\ t) \tag{13.3}$$

The most popular and used numerical integration methods introduced in the vast thematic literature are Euler method, Rung-Kutta methods and Adams predictor-corrector methods. These methods have been known for many years, for instance, the Runge-Kutta methods have been known for more than an 100 years, but their potential was not fully realized until computers became available. These methods involve a step-by-step process in which a sequence of discrete points t^0, t^1, t^2, …, t^n is generated. The discrete points may have either constant or variable spacing defined as $h^i = t^{i+1} - t^i$, where h^i is the step size for any discrete point t^i. At each point t^i, the solution $y(t^i)$ is approximated by a number y^i. Since no numerical method is capable of finding $y(t^i)$ exactly, the quantity

$$\varepsilon_g^i = \left| y(t^i) - y^i \right| \tag{13.4}$$

represents the global or total error at $t = t^i$. The total error consists of two components, the truncation error and the round-off error. The truncation error depends on the nature of the numerical algorithm used in computing y^i. The round-off error is due to the finite word length in a computer.

The integration methods are called single step methods when they only require information on the current time step to advance to the next time step. Euler and Runge-Kutta methods are single step methods. When information of the previous steps is used, the algorithm methods are called multistep methods, as it is the case of Adams predictor-corrector schemes. The single step methods are self starting and they need a minimum amount of storage requirements. However, these methods require a larger number of function evaluations, for instance, four for the fourth-order Runge-Kutta method. Function evaluation is the name of the process by which, given t and y, the value of $\dot{y}$ is computed. The multistep methods require a small amount of function evaluations, particularly if the time step is chosen so that the number of predictor-corrector iterations per step is kept below two or three. Moreover, error estimates are easily provided and step size adjustments can be performed with no difficulties. The multistep methods are not self starting and require the help of a single step scheme to start the integration process (Atkinson 1989).

Regardless of the numerical method used, the numerical task deals with the integration of an initial-value problem that can be written as

$$\dot{y}_1 = f(y, t) \tag{13.5}$$

With the initial condition $y(t^0) = y^0$ and where y is the variable to be integrated and function $f(t, y)$ is defined by the computational sequence of the algorithm selected. Equation (13.5) has a solution $y(t)$. The initial value y^0 can be defined for any value of t^0, although it is often assumed that a transformation has been made so that $t^0 = 0$. This does not affect the solution or method used to approximate the solution.

It is known that the Euler integration method is one of the simplest integrators available. This approach may be sufficient in giving a very rough idea of the motion of multibody systems. This method solves differential equations in a single step as

$$y^{i+1} = y^i + hf(y^i, t) \tag{13.6}$$

where h is the integration step size $h = t^{i+1} - t^i$, for i a non-negative integer. This method implies that the next step of the state variable can be evaluated by using the current state variable.

The intuitive basis of the Euler method is illustrated in Fig. 13.1, in which the curve labeled $y = y(t)$ is the solution of the differential Eq. (13.5), which passes through point $P(t^0, y^0)$. It is desired to find the value of $y^1 = y^0 + \Delta y$ corresponding to $t = t^1$. In other words, the height RQ needs to be determined. Although the position of the curve at every point is not known, its slope is equal to $f(t, y)$, which is simply the geometric interpretation of the differential equation. Thus, the slope of the tangent at point P is $\dot{y}^0 = f(t^0, y^0)$, which can be computed since y^0 and t^0 are both known. If h is reasonable small, the tangent line PS should not deviate too much from the curve PQ, hence, the height RS (which by simple geometry is equal to $h\dot{y}^0$) should be an approximation to the required height RQ. Thus, a first

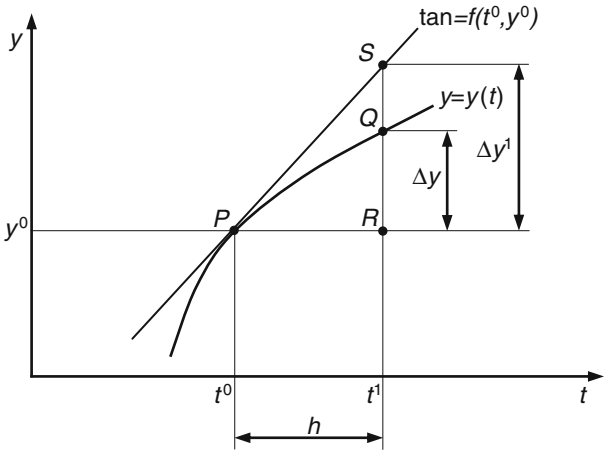

Fig. 13.1 Geometric interpretation of the Euler integration method

approximation to Δy is given by $\Delta y^1 = RS = hf(t^0, y^0)$. Assuming that the appropriate derivatives exist, then $y(t)$ can be expanded in a Taylor series about $t = t^i$ and the expression is evaluated at $t = t^{i+1}$, yielding

$$y(t^{i+1}) = y(t^i) + hf(t^i, y^i) + O(h^2) \tag{13.7}$$

From the analysis of Eq. (13.7), neglecting the higher-order terms, the discretization or local truncation error is given by

$$\varepsilon_l = O(h^2) \tag{13.8}$$

The order of a numerical integration method can be used to specify its accuracy and can be expressed using the local truncation error. Knowing that for a scalar equation of type

$$\varepsilon_l = O(h^{p+1}) \tag{13.9}$$

is said to be of pth order, then it is clear that the Euler integration method is of first order. Thus, for highly oscillatory motion there are rapid changes in the derivatives of the function and if h is too large, then inaccuracies in the computation of the state variables are made (Nikravesh 1988).

In turn, the global truncation error at t^i can be evaluated as the difference between the actual and computed solution, in the absence of round-off error by the end of the simulation, that is

$$\varepsilon_g^i = \left| y(t^i) - y^i \right| \tag{13.10}$$

A more accurate integration method is the second-order Runge-Kutta algorithm, which can be expressed as

$$y^{i+1} = y^i + \frac{h}{2}(f_1 + f_2) \tag{13.11}$$

where

$$f_1 = f(t^i, y^i) \tag{13.12}$$

$$f_2 = f(t^i + h, y^i + hf_1) \tag{13.13}$$

This approach is also known as the improved Euler method, modified trapezoidal method or the Heun method. It should be noted that two function evaluations are required per time step, which in the case of multibody systems implies the solution of the equations of motion to obtain the accelerations twice at the given time step. Figure 13.2 shows the geometric interpretation of the second-order

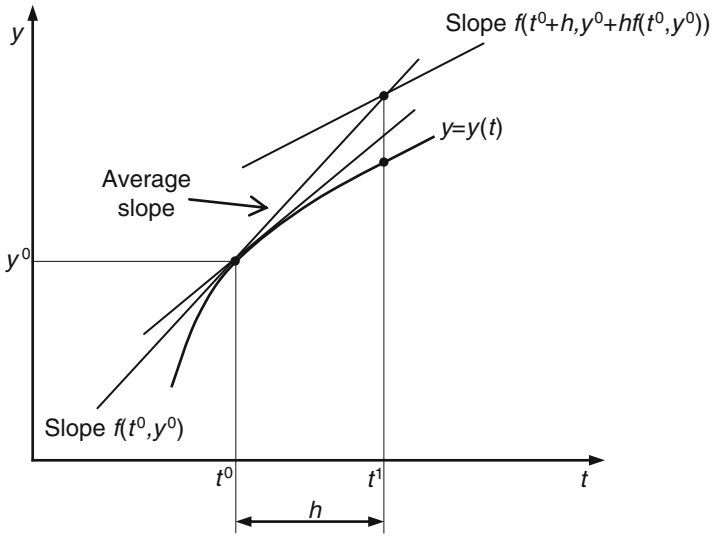

Fig. 13.2 Geometric interpretation of the second-order Runge-Kutta method

Runge-Kutta method. This method is explicit in the measure that f_1 does not depend on f_2 and neither one depends on y^{i+1} (Jalón and Bayo 1994).

The local error of the second-order Runge-Kutta method is of order h^3, whereas that of Euler method is h^2. Thus, it is expected to be able to use a larger time step with the second-order Runge-Kutta method. The price to pay for this is that it requires to evaluate the function $f(t, y)$ twice for each time step of the integration process.

For larger time steps and for greater accuracy, the fourth-order Runge-Kutta integration method is most popular and widely used. This method is stable and, as a computer program, occupy relatively small amount of core storage. The fourth-order Runge-Kutta integration algorithm can be expressed by Pina (1995)

$$y^{i+1} = y^i + hg \tag{13.14}$$

where

$$g = \frac{1}{6}(f_1 + 2f_2 + 2f_3 + f_4) \tag{13.15}$$

$$f_1 = f(t^i, y^i) \tag{13.16}$$

$$f_2 = f(t^i + \frac{h}{2},\ y^i + \frac{h}{2}f_1) \tag{13.17}$$

$$f_3 = f\left(t^i + \frac{h}{2},\ y^i + \frac{h}{2} f_2\right) \tag{13.18}$$

$$f_4 = f\left(t^i + h,\ y^i + h f_3\right) \tag{13.19}$$

This method is explicit because all f_i depend only on previous values already calculated. This algorithm is easy to implement in the measure that it only requires function evaluations, and it is self starting integrator scheme, which means that there is no need for any other algorithm or technique to start the integration process.

Figure 13.3 illustrates the geometric interpretation of the fourth-order Runge-Kutta integration method. In this method four tangents are determined, being their average weighted according to Eqs. 13.14–13.19.

The standard fourth-order Runge-Kutta method does not provide an estimate of the local error, so that the user does not have way of knowing whether the time step being used is adequate. The local error of this method is of order h^5, which is relatively small even for larger time steps. The major disadvantage of this method is that the function $f(t, y)$ needs to be evaluated four time at each time step.

For the Euler and Runge-Kutta methods the next step value y^{i+1} is computed by using solely the current value y^i and time t^i, over a time range of $h = t^{i+1} - t^i$. Multistep methods utilize information about the solution at more than one point. The objective of the multistep methods is to automatically select the proper order and the proper time step size, which will minimize the amount of work required to achieve the specified accuracy for a given problem. The multistep algorithms require only two function evaluation per step compared with four function evaluations per step with the fourth-order Runge-Kutta method, being, therefore,

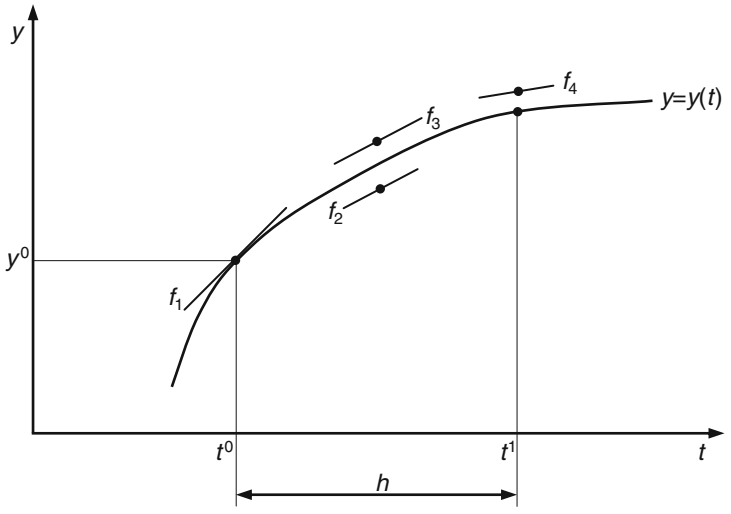

Fig. 13.3 Geometric interpretation of the fourth-order Runge-Kutta method

considerably faster and require less computation work. Predictor-corrector methods provide an automatic error estimate at each time step, thus allowing the algorithm to select an optimum value of h for a required accuracy. This type of approach is also better with respect to the propagation of error that it can use time steps more than twice as large.

In Adams predictor-corrector methods an explicit method is used to predict a value of y^{i+1}, while an implicit method corrects that value. The implicit corrects appear to be more stable and accurate than the explicit predictors and are both chosen to be of equal order. The Adams-Bashforth predictor algorithm of fourth-order can be written as

$$y^{i+1} = y^i + \frac{h}{24}(55f^i - 59f^{i-1} + 37f^{i-2} - 9f^{i-3}) \tag{13.20}$$

where

$$f^i = f(t^i, y^i) \tag{13.21}$$

$$f^{i-j} = f(t^{i-j}, y^{i-j}), \qquad (j = 1, 2, 3) \tag{13.22}$$

The corresponding Adams-Moulton corrector algorithm can be expressed by

$$y^{i+1} = y^i + \frac{h}{24}(9f^{i+1} + 19f^i - 5f^{i-1} + f^{i-2}) \tag{13.23}$$

where

$$f^i = f(t^i, y^i) \tag{13.24}$$

$$f^{i-j} = f(t^{i-j}, y^{i-j}), \qquad (j = 1, 2) \tag{13.25}$$

The major disadvantage of multistep methods is that they are not self starting. Thus, in the fourth-order Adams predictor-corrector method four successive values of function evaluation at equally spaced points before instant of time t^i must be known. These starting values must be obtained by some independent method, such as the Runge-Kutta method. On the other hand, Adams predictor-corrector algorithms are more complicated to program in the measure that they require special techniques for starting and for doubling and halving the time step, and they be subject to numerical instability (Conte and Boor 1981). In short, the Adams methods, when being carefully use, are more efficient than any other method. To achieve this efficiency it is necessary to vary the time step and the order that are used. Thus, it is necessary to estimate the errors that are incurred for various time steps and orders so as to make these decisions. Advanced codes also attempt to detect abnormal situations such as discontinuities or certain types of instabilities and to deal with them in a reasonable way. A detailed discussion on the Adams

predictor-corrector implementation can be found in the book by Shampine and Gordon (1975).

Gear (1971) developed a family of variable order stiffly-stable algorithms for the solution of stiff problems. A stiff system is referred to as any initial-value problem in which the complete solution consists of fast and slow components. The stiffness can be produced by physical characteristics of the multibody systems, such as components with large differences in their masses, stiffness and damping. However, in many other instances, stiffness is numerically induced due to either the discretization process, the large number of components and equations of motion, or sudden or accumulated violations in the constraint conditions.

The Gear algorithm of fourth-order can be expressed as

$$y^{i+1} = \frac{1}{25}(48y^{i} - 36y^{i-1} + 16y^{i-2} - 3y^{i-3} + 12hf^{i+1}) \tag{13.26}$$

where

$$f^{i+1} = f(t^{i+1}, y^{i+1}) \tag{13.27}$$

Since the Gear algorithm is an implicit multistep scheme, it is necessary to solve an implicit equation in each time step (Nikravesh 1988).

References

Atkinson KA (1989) An introduction to numerical analysis, 2nd edn. Wiley, New York

Blajer W (1999) Elimination of constraint violation and accuracy improvement in numerical simulation of multibody systems. In: Ambrósio J, Schiehlen W (eds) Proceedings of EUROMECH colloquium 404, advances in computational multibody dynamics IDMEC/IST. Lisbon, 20–23 Sept, pp 769–787

Brenan KE, Campbell SL, Petzold LR (1989) Numerical solution of initial-value problems in differential-algebraic equations. Elsevier, New York

Conte SD, Boor C (1981) Elementary numerical analysis: an algorithmic approach, 3rd edn. McGraw-Hill, Singapore

Gear CW (1971) Numerical initial value problems in ordinary differential equations. Prentice-Hall, Englewood Cliffs

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York

Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs

Petzold LR (1983) A description of DASSL: a differential/algebraic system solver. In: Stepleman R et al (ed) Scientific computing North-Holland Pub. Co. pp 65–68

Pina H (1995) Métodos numéricos. McGraw-Hill, Lisboa

Shampine L, Gordon M (1975) Computer solution of ordinary differential equations: the initial value problem. Freeman, San Francisco

Chapter 14
Correction of the Initial Conditions

Abstract This chapter presents a general approach to deal with the correction of the initial conditions at the position and velocity levels. This procedure is of paramount importance to avoid constraints violation during the numerical resolution of the equations of motion. The material presented here closely follows the standard methodologies available in the literature. Thus, in this chapter, a simple and efficient approach to correct the initial conditions at the position and velocity levels is revised.

Keywords Initial conditions · Positions · Velocities

It was verified in the paragraph sections that the numerical solution of the dynamic equations of motion (12.3) requires a set of initial conditions on the positions and velocities. Moreover, this system of equations of motion does not use explicitly the position and velocity equations associated with the kinematic constraints. Consequently, for moderate and long time simulations, the original constraint equations start to be violated due to the integration process and inaccurate initial conditions. Therefore, special procedures must be followed to avoid or minimize this phenomenon. Several methods to deal with this problem have been suggested and tested over the last years (Blajer 2002; Zahariev and McPhee 2003; Nikravesh 2007; Masarati 2011).

It is well known that for a constrained multibody system, the kinematic constraints can be described by a set of linear and/or nonlinear algebraic equations as (Nikravesh 2008)

$$\boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \tag{14.1}$$

The time derivative of these constraints provide the velocity constraints that can be written as

$$\dot{\Phi}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{D}\mathbf{v} = \mathbf{0} \tag{14.2}$$

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_14

Let consider that the initial conditions for positions ($\mathbf{q}^0$) do not satisfy Eq. (14.1), then the corrected positions can be expressed as

$$\mathbf{q}^c = \mathbf{q}^0 + \delta\mathbf{q} \tag{14.3}$$

where $\delta\mathbf{q}$ represents the set of corrections that adjusts the initial conditions, such that

$$\mathbf{\Phi}(\mathbf{q}^c) = \mathbf{\Phi}(\mathbf{q}^0) + \delta\mathbf{\Phi} = \mathbf{0} \tag{14.4}$$

and, hence

$$\delta\mathbf{\Phi} = -\mathbf{\Phi}(\mathbf{q}^0) \tag{14.5}$$

For small $\delta\mathbf{\Phi}$, Eq. (14.1) can be expanded and the higher order terms can be neglected, yielding the variation of the constraint equations as (Flores et al. 2011)

$$\delta\mathbf{\Phi} = \frac{\partial\mathbf{\Phi}}{\partial\mathbf{q}}\delta\mathbf{q} = \mathbf{D}\delta\mathbf{q} \tag{14.6}$$

Substituting now Eq. (14.6) in Eq. (14.5) results in

$$\delta\mathbf{q} = -\mathbf{D}^{-1}\mathbf{\Phi}(\mathbf{q}^0) \tag{14.7}$$

It must be noted that, in general, the Jacobian matrix, $\mathbf{D}$, is not square, therefore, $\mathbf{D}^{-1}$ does not exist. However, the concept of the Moore-Penrose generalized inverse matrix can be employed as (Rao and Mitra 1971; Neto and Ambrósio 2003)

$$\mathbf{D}^{+} = \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1} \tag{14.8}$$

such that

$$\mathbf{D}\mathbf{D}^{+}\mathbf{D} = \mathbf{D} \tag{14.9}$$

$$\mathbf{D}^{+}\mathbf{D}\mathbf{D}^{+} = \mathbf{D}^{+} \tag{14.10}$$

and both $\mathbf{D}^{+}\mathbf{D}$ and $\mathbf{D}\mathbf{D}^{+}$ are symmetric matrices.

Consequently,

$$\mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1} = \mathbf{D}^T(\mathbf{D}^{+})^T\mathbf{D}^{+} = (\mathbf{D}^{+}\mathbf{D})^T\mathbf{D}^{+} = \mathbf{D}^{+}\mathbf{D}\mathbf{D}^{+} = \mathbf{D}^{+} \tag{14.11}$$

Thus, Eq. (14.7) can be rewritten in the following form

$$\delta\mathbf{q} = -\mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{\Phi}(\mathbf{q}^0) \tag{14.12}$$

Since the kinematic constraint equations at the position level are, in general, nonlinear, Eq. (14.12) must be solved iteratively by employing a numerical algorithm. A Newton-Raphson algorithm to solve the nonlinear position constraints can be stated, for i-th iteration, as (Nikravesh 2007)

```
For q=q^i
    Evaluate Φ(q^i)=0
    Compute δq^i=-D^T(DD^T)^-1 Φ(q^i)
    Correct  q^(i+1)=q^i+δq^i
    Repeat if necessary
```

This approach is effective in provide the correct initial positions, and, in general, only a few number of iterations is necessary to converge with this Newton-Raphson scheme.

In a similar manner, the corrected initial velocities can be written as

$$\mathbf{v}^c = \mathbf{v}^0 + \delta\mathbf{v} \tag{14.13}$$

It must be stated that at this stage, the initial conditions for positions are already corrected. Thus, from Eq. (14.13) it can be verified that

$$\dot{\Phi}(\mathbf{q}^c, \mathbf{v}^c) = \dot{\Phi}(\mathbf{q}^c, \mathbf{v}^0) + \delta\dot{\Phi} = \mathbf{0} \tag{14.14}$$

and, hence

$$\delta\dot{\Phi} = -\dot{\Phi}(\mathbf{q}^c, \mathbf{v}^0) \tag{14.15}$$

Again, for small $\delta\dot{\Phi}$, the variation of the constraint velocity equations can be obtained from Eq. (14.2) as

$$\delta\dot{\Phi} = -\mathbf{D}\delta\mathbf{v} \tag{14.16}$$

Combining Eqs. (14.15) and (14.16) yields

$$\delta\mathbf{v} = -\mathbf{D}^{-1}\dot{\Phi}(\mathbf{q}^c, \mathbf{v}^0) \tag{14.17}$$

Introducing now Eq. (14.8) in Eq. (14.17) results in

$$\delta\mathbf{v} = -\mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}\dot{\Phi}(\mathbf{q}^c, \mathbf{v}^0) \tag{14.18}$$

Finally, the correction of the initial velocities can be expressed as

$$\mathbf{v}^c = \mathbf{v}^0 - \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}\dot{\Phi}(\mathbf{q}^c, \mathbf{v}^0) \tag{14.19}$$

References

Blajer W (2002) Elimination of constraint violation and accuracy aspects in numerical simulation of multibody systems. Multibody Syst Dyn 7:265–284

Flores P, Machado M, Seabra E, Silva MT (2011) A parametric study on the Baumgarte stabilization method for forward dynamics of constrained multibody systems. J Comput Nonlinear Dyn 6(1):011019, 9p

Masarati P (2011) Constraint stabilization of mechanical systems in ordinary differential equations form. Proc Inst Mech Eng Part K J Multi-body Dyn 225:12–33

Neto MA, Ambrósio J (2003) Stabilization methods for the integration of DAE in the presence of redundant constraints. Multibody Syst Dyn 10:81–105

Nikravesh PE (2007) Initial condition correction in multibody dynamics. Multibody Syst Dyn 18:107–115

Nikravesh PE (2008) Planar multibody dynamics: formulation, programming, and applications. CRC Press, London

Rao CR, Mitra SK (1971) Generalized inverse of matrices and its applications. Wiley, New York

Zahariev E, McPhee J (2003) Stabilization of constraints of multibody system dynamics. Mech Based Des Struct Mach 31:25–55

Chapter 15
Demonstrative Example of Application

Abstract In this chapter a simple pendulum is considered as a demonstrative example of application of the methodologies described in the previous paragraphs. This example allows for the comparison of the different methods to solve the equations of motion in terms of accuracy and efficiency. Finally, the main concluding remarks of the material presented here are summarized and analyzed.

Keywords Application example · Constraints violation · Spatial system

The simple pendulum is made up of one rigid body, which is connected by revolute joint to the ground. This system is modeled with six coordinates and five kinematic constraints, which results in a system with one degree of freedom. Initially, the pendulum is resting in the *xy* plane position, as Fig. 15.1 shows. The system is then released from this initial configuration under the gravity action only, which is taken as acting in the negative *z* direction. The geometric dimension and inertia properties of the simple pendulum are listed in Table 15.1 (Flores et al. 2008).

Long time computational simulations are performed in order to test and compare the accuracy and efficiency of use different methods to solve the dynamic equations of motion. For this purpose, the four approaches are considered, namely the standard method based on the technique of Lagrange multipliers, the Baumgarte method, the penalty method and the augmented Lagrangian formulation. The quantitative measure of the efficiency of these approaches is drawn from the constraint violation as $\mathbf{\Phi}^T\mathbf{\Phi}$, as well as the number of function evaluations and the time consumed during the dynamic simulations. Table 15.2 gives the parameters used for the different models, required to characterize the problem, and for the numerical methods, required to solve the system dynamics. In the present case, there is no need for the initial conditions correction in the measure that the correct initial data can be easily determined.

Figure 15.2 shows that when the standard method is utilized the violation of the constraint equations grows indefinitely with time. In fact, this approach produces unacceptable results because the constraint equations are rapidly violated due to the inherent instability of the equations used and to the numerical errors that develop during computation. In sharp contrast, with other methods the behavior of the simple

P. Flores, *Concepts and Formulations for Spatial Multibody Dynamics*,
SpringerBriefs in Applied Sciences and Technology,
DOI 10.1007/978-3-319-16190-7_15

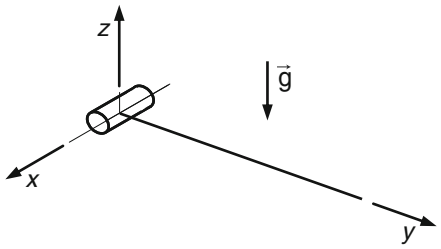

Fig. 15.1 Simple pendulum modeled by one rigid body and one revolute joint

Table 15.1 Governing properties for the simple pendulum

Length (m)	Mass (kg)	Moment of inertia (kgm^2)		
		$I_{\xi\xi}$	$I_{\eta\eta}$	$I_{\zeta\zeta}$
1.0	7.02	0.5855265	0.0010530	0.5855265

Table 15.2 Parameters used for the dynamic simulations

Final time of simulation	10.0 s	Baumgarte-α	5
Integrator algorithm	ode45	Baumgarte-β	5
Reporting time step	0.02 s	Penalty-α	1×10^7
Relative tolerance	1×10^{-6}	Penalty-ω	10
Absolute tolerance	1×10^{-9}	Penalty-μ	1

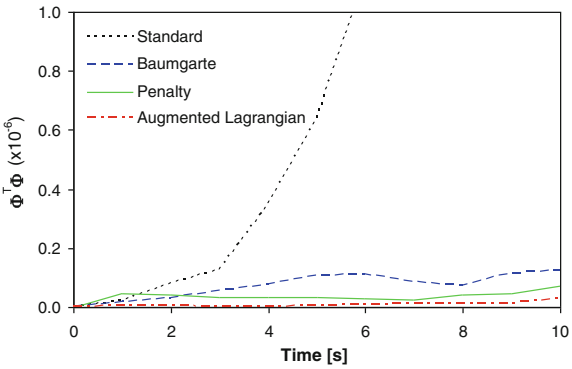

Fig. 15.2 Constraints violation at the position level

pendulum is different in the measure that the level of the constraints violation is kept under control during the dynamic simulations. Indeed, Baumgarte approach, penalty method and augmented Lagrangian formulation, experience tells that the numerical result does not diverge from the exact solution, but oscillates around it. Magnitude and frequency of the oscillation depend on the values of penalty parameters. Table 15.3 lists the number of function evaluations and the time consumed during the

Table 15.3 Function evaluations and time consumed

Method	Function evaluations	Time consumed (s)
Standard	1639	6.3
Baumgarte	2401	10.7
Penalty	3769	17.2
Augmented lagrangian	3769	17.6

dynamic simulations for the cases described above, which can be used to have a rough idea about the computational efficiency of the different methods.

The fundamentals of the formulation for the dynamics of spatial multibody systems have been presented throughout this work. In first place, the definition of multibody system, made of interconnected bodies that undergo large displacements and rotations, was presented. In addition, basic concepts in multibody dynamics were also introduced. The main types of coordinates that can be used in the formulation of the equations of motion of constrained multibody systems were analyzed, in which their relative advantages and drawbacks were also discussed.

Displacements, velocities and accelerations are quantities frequently used to characterize the motion properties of the multibody systems. For this purpose, a proper system of coordinates must be adopted. In addition, special attention must be given to the selection of the angular coordinates used to describe the orientation of the bodies. In the study, the Euler angles, Bryant angles or Euler parameters were presented. However, due to the singularity phenomenon associated with the Euler angles and Bryant angles, the Euler parameters has been selected as the set of rotational coordinates utilized to define the orientation of the bodies (Nikravesh 1988).

From the mathematical point of view, Cartesian coordinates and Euler parameters are the supporting structure for all methodologies and dynamic analysis developed within the multibody systems methodologies. In the sequel of this concept, the constraint equations associated with the basic kinematic joints, as well as those related to the constraints between two vectors, were presented. In addition, their contributions to the Jacobian matrix of the constraints and to the right-hand side of acceleration constraint equations were studied (Shabana 1989; Schiehlen 1990; Jalón and Bayo 1994).

The formulation of multibody systems adopted in this work uses the generalized coordinates and the Newton-Euler approach to derive the equations of motion. This formulation results in the establishment of a mixed set of differential and algebraic equations, which are solved in order to predict the dynamic behavior of multibody systems. This approach is very straightforward in terms of assembling the equations of motion and providing all joint reaction forces. Additionally, to the standard method based on the Lagrange multipliers technique, three different approaches were presented and utilized to solve the equations of motion, namely the Baumgarte stabilization scheme, the penalty method and the augmented Lagrangian formulation (Baumgarte 1972; Jalón and Bayo 1994).

Under the framework of the multibody systems formulation, some of the fundamental forces that can act upon the bodies were also presented. These forces include the action of gravitational field and forces due to springs, dampers and actuators. Applied forces can be represented by force elements that act on the system components, modifying their dynamic response and the overall motion of the multibody systems. Moreover, the joint reaction forces, expressed in terms of the Jacobian matrix of the constraint equations and a vector of Lagrange multipliers, were analyzed.

In a simple way, the equations of motion for constrained multibody systems are expressed in the Hessenberg form. A set of initial conditions imposed on the positions and velocities is required to start the dynamic simulation. The selection of the appropriate initial conditions plays a crucial role in the prediction of the dynamic response of multibody systems. The subsequent initial conditions, for each time step in the simulation, are obtained from the final conditions of the previous time step. Then, from the initial values for positions and velocities, the equations of motion are solved for accelerations. The positions and velocities at the next time step are then obtained by integration of the velocity and acceleration vectors. This procedure is repeated until the final time of simulation is reached. The integration process can be performed by using a constant step size scheme, such as the fourth-order Runge-Kutta method, or a predictor-corrector algorithm with both variable step and order, such as the Gear method. The fundamental issues related to the numerical integrators used in dynamic analysis of multibody systems were briefly analyzed.

It was demonstrated that the numerical solution of the dynamic equations of motion requires a set of initial conditions on the positions and velocities. Moreover, this system of equations of motion does not use explicitly the position and velocity equations associated with the kinematic constraints. Hence, for moderate and long time simulations, the original constraint equations start to be violated due to the integration process and inaccurate initial conditions. Therefore, a special procedure to avoid this phenomenon was presented, which allows for the correction of the set of initial conditions. The described method is a simple, general and effective to correct the initial conditions at the position and velocity levels is presented. Finally, a simple pendulum was considered as a demonstrative example of application of the methodologies described in the present work. This example was utilized to quantify the accuracy and efficiency of the different methods presented to solve the dynamic equations of motion.

References

Baumgarte J (1972) Stabilization of constraints and integrals of motion in dynamical systems. Comput Methods Appl Mech Eng 1:1–16

Flores P, Ambrósio J, Claro JCP, Lankarani HM (2008) Kinematics and dynamics of multibody systems with imperfect joints: models and case studies. Lecture notes in applied and computational mechanics, vol 34. Springer, Berlin

Jalón JG, Bayo E (1994) Kinematic and dynamic simulations of multibody systems: the real-time challenge. Springer, New York
Nikravesh PE (1988) Computer-aided analysis of mechanical systems. Prentice Hall, Englewood Cliffs, New Jersey
Schiehlen W (1990) Multibody systems handbook. Springer, Berlin
Shabana AA (1989) Dynamics of multibody systems. Wiley, New York